

FAKULTETA ZA KEMIJO IN KEMIJSKO TEHNOLOGIJO Smetanova ulica 17 2000 Maribor, Slovenija www.fkkt.um.si

Majda Krajnc

Računalništvo v kemiji

zapiski predavanj

2. popravljena in dopolnjena izdaja

Maribor, 2014

Copyright 2014.	1. izdaja 2009
	2. (popravljena in dopolnjena) izdaja 2014

Majda Krajnc, Računalništvo v kemiji

Avtorica:	doc. dr. Majda Krajnc	
Recenzentka:	dr. Severina Oreški, FKKT Univerze v Mariboru	
Vrsta publikacije:	zapiski predavanj	
Založnik:	FKKT Univerze v Mariboru	
Naklada:	On-line	
Dostopno na naslovu: <u>http://atom.uni-mb.si/stud/egradiva.php</u> in na univerzitetnem e-učnem okolju Moodle pri predmetih Računalništvo v kemiji in Procesno računanje l		

Gradiva iz publikacije, brez dovoljenja avtorice, ni dovoljeno kopirati, reproducirati, objavljati ali prevajati v druge jezike.

ISBN 978-961-248-416-3

CIP - Kataložni zapis o publikaciji Univerzitetna knjižnica Maribor

004:54(075.8)

KRAJNC, Majda

Računalništvo v kemiji [Elektronski vir] : zapiski predavanj / Majda Krajnc. - 2. Popravljena in dopolnjena izd. - Maribor : Fakulteta za kemijo in kemijsko tehnologijo, 2014 Način dostopa (URL): <u>http://atom.uni-mb.si/stud/egradiva.php</u>

ISBN 978-961-248-416-3 COBISS.SI-ID 76551169

Predgovor k drugi izdaji

Gradivo je koristno dopolnilo k predavanjem pri predmetih Računalništvo v kemiji in Procesno računanje I v 1. letniku UNI oziroma VS programa na naši fakulteti.

V prvem delu zapiskov so predstavljene teoretične osnove programskega jezika fortran obogatene s primeri, ki jih, če je le mogoče, izvajamo sproti. Veliko primerov ima »kemijsko« vsebino, nekaj pa je splošnih iz vsakdanjega življenja. S tem je poudarjena kompleksnost uporabe fortrana na različnih področjih. Ko študentje spoznajo osnove programiranja, jim je v nadaljevanju olajšano delo in razumevanje uporabnosti programa Excel.

Za posameznimi poglavji so predstavljene praktične vaje, ki jih lahko študenti rešujejo sami, ter vprašanja za preverjanje znanja. Seveda samo primeri, ki so zbrani v tem gradivu, po navadi niso dovolj za kakovostno znanje programiranja v fortranu. Koristno je obiskovati predavanja in si beležiti manjkajoče podatke ter napotke, ki jih daje predavatelj. Zato so primeri napisani tako, da lahko zapisujete še ostale teoretične zakonitosti, ki veljajo pri programiranju. Moram poudariti, da je dobro obvladati fortran (tako kot vsak jezik) mogoče le z veliko truda in vaje ter nenehnega izpopolnjevanja. Prav zaradi tega študentje usvojeno znanje poglobijo na vajah v računalniški učilnici, ki sledijo predavanjem.

V drugem delu predavanj je predstavljena uporaba programskega paketa MS EXCEL 2007. Poudarek je na predstavitvi različnih kemijskih problemov, ki jih lahko rešujemo in oblikujemo s tem programom. Študentje usvojijo vnašanje podatkov v preglednice, vnašanje enačb, uporabo obstoječih funkcij, risanje grafikonov, osnove regresije, reševanje nelinearnih enačb z eno neznanko, reševanje sistemov linearnih enačb ter reševanje integralnih in diferencialnih enačb. Navodila so napisana tako, da lahko študent samostojno usvaja teorijo.

Dopolnila v drugi izdaji so bila izdelana na osnovi naslednje literature:

- R. M. Felder, R. W. Rousseau, Elementary Principles of Chemical Processes. Third Edition, John Wiley & Sons, Inc., New York, 2000;
- D. M. Bourg, Excel Scientific and Engineering Cookbook, O'Reilly Media, Inc., Sebastopol, CA, 2006;
- B. Orel, Osnove numerične matematike, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Ljubljana, 2004;
- L. Townsend, Using Excel to solve differential equations, 2003.

Gradivo je napisano zgoščeno in pregledno, a kljub temu se je verjetno pojavila kakšna neljuba napaka oziroma nejasnost v tekstu. Če ste jih našli, jih sporočite predavatelju ali asistentu, da jih bomo popravili in za naslednje generacije pripravili še kvalitetnejše gradivo.

Na koncu bi se zahvalila svoji dolgoletni asistentki in kolegici dr. Severini Oreški za pregled in strokovno mnenje vsebine, saj je tako dodatno pripomogla k ustvarjanju kakovostnega gradiva. Prav tako bi se zahvalila podiplomski študentki Vanji Forjan, ki mi je s svojim znanjem in veščinami uporabe MS Excela 2007 pripravila del navodil, ki sem jih lahko vključila v vsebino drugega dela. Zahvala tudi vsem ostalim, sodelavcem, kolegom, študentom, za koristne predloge saj sem z njihovo pomočjo izboljševala vsebino gradiva.

Študentom pa še nasvet: naj vam gradivo služi kot pripomoček pri delu v času študija kakor tudi kasneje v praksi na delovnem mestu. Pri programiranju in delu z MS Excelom vam želim veliko uspeha.

Majda Krajnc

KAZALO

1. DEL: PROGRAMIRANJE V FORTRANU

1.	UVOD	3
	1.1 Spoznavanje računalnika	3
	1.1.1 Predstava spomina oziroma pomnilnika	4
	1.1.1.1 Binarno kodiranje	5
	1.2 Spoznavanje strojne opreme (Hardware)	7
2.	OSNOVE	11
	2.1 Predstavitev fortrana	11
	2.2 Konstante in spremenljivke	13
	2.3 Aritmetični izrazi	13
	2.3.1 Vrstni red izvajanja aritmetičnih operacij v aritmetičnem izrazu	13
	2.4 Cela in realna števila	15
	2.4.1 Stavka INTEGER in REAL	16
3.	POMEMBNE (matematične) FUNKCIJE	19
4.	IZPISOVANJE IN BRANJE PODATKOV	23
	4.1 Stavek WRITE	23
	4.2 Stavek READ	23
	4.3 Izpis na datoteko	24
	4.4 Branje z datoteke	25
	4.5 Komentarji v programu	26
5.	SISTEMATIČNI PRISTOP K UČINKOVITEMU PROGRAMIRANJU	29
	5.1 Problem	29
	5.2 Analiza problema	29
	5.3 Algoritem in diagram poteka	30
	5.4 Računalniški program	31
6 .	ZANKE	33
	6.1 Zanka DO s števcem	33
	6.2 Zanka DO z inkrementom ali prirastkom	35
	6.3 Stavek GO TO	35
7.	STAVEK FORMAT IN FORMATNA DOLOČILA	39
	7.1 Stavek FORMAT	39
	7.2 Različna formatna določila	39
8 .	POGOJNI STAVKI	45
	8.1 Enostavni ali logični stavek IF	45
	8.2 Blokovna IF struktura	46
	8.3 Struktura IF THEN ELSE	47
9.	ENODIMENZIONALNA POLJA	51
	9.1 Uporaba elementov polja v zanki DO	51
	9.2 Vhod (READ) in izhod (WRITE) z implicitno zanko DO	51
10	. PODPROGRAMI	57
	10.1 Podprogram FUNCTION (funkcija)	57
	10.1.1 Uporaba podprograma FUNCTION in presnos podatkov	59
	10.2 Podprogram SUBROUTINE	59
11	. OSTALO	63
	11.1 Stavek COMMON	63
	11.2 Stavek DATA	64

2.	DEL	: MS EXCEL 2007 – Kratka navodila za uporabo	
12.	OSN	OVE DELA V PROGRAMU EXCEL 2007	69
	12.1	Zagon programa Microsoft Excel	69
	12.2	Zapiranje programa Microsoft Excel	70
	12.3	Excelovo delovno okno	71
13.	OSN	OVNE OPERACIJE	73
	13.1	Vnos podatkov	73
		13.1.1 Vnos besedil	73
		13.1.2 Vnos števil	74
		13.1.3 Vnos datuma in časa	74
	13.2	Premikanje po listih	75
	13.3	Premikanje in kopiranje podatkov	75
	13.4	Spreminjanje ali brisanje podatkov	76
	13.5	Vstavljanje in brisanje celic, vrstic ali stolpcev	76
	13.6	Operacije z delovnimi listi	78
		13.6.1 Pregledovanje listov	78
		13.6.2 Preimenovanje delovnih listov	78
		13.6.3 Vstavljanje in brisanje listov	79
		13.6.4 Premikanje in kopiranje listov	79
14.	OBL	IKOVANJE PODATKOV	81
	14.1	Uporaba traku Osnovno	81
	14.2	Uporaba okna Oblikovanje celice	81
		14.2.1 Številka	81
		14.2.2 Poravnava	82
		14.2.3 Pisava	83
		14.2.4 Obroba	83
		14.2.5 Polnilo	03 04
	14.3	Urejanje podatkov v tabele	84 95
	14.4	Samooblikovanje tabele	96
	14.5	Določanje širine stolpcev in vrstic	20
15.	FOR		09
	15.1	Formule	80
		15.1.1 Vrste formul	89
		15.1.2 Uporaba racunskih operatorjev v formulan	Q1
		15.1.3 Vrstni red izvajanja operacij v formulan	92
	4 - 0	15.1.4 Uporaba skilcev v formulan	92
	15.2		93
	15 2	15.2. I IZDOF IUNKCIJE	97
16	ID.J		101
10.		Flomenti grafikana	102
	10.1	Vieto grafikona	103
	10.2	16.2.1 Stolnění grofikon	103
		16.2.2 Polični grafikon	104
		16.2.2 Fallolli yrallikoli	104
		16.2.4 Tortni grafikon	105
		16.2.5 Ploščinski grafikon	105
		16.2.6 Polarni grafikon	106
		16.2.7 Tridimenzionalni grafikoni (3D)	106
	16.2	Izdelava grafikonov	107
	10.0	16.3.1 Hitro izdelovanje grafikonov	107
		16.3.2 Spreminianie videza grafikona	109
		16.3.3 Oblikovanie legend in naslovov grafikonov	110
		. e.e.e e.eoranje regena in naciorer graintener in ministri in ministri in ministri in ministri in ministri	

16.3.4 Spreminjanje telesa grafikona	111
16.3.5 Prilagajanje podatkov grafikona	114
16.3.6 Kreiranje logaritemskih in semilogaritemskih skal	118
16.3.7 Prileganje krivulj (Curve Fitting), regresija (Regression)	120
17. POGOJNI STAVKI V MS EXCELU	125
17.1 Uporaba pogojnega stavka IF v Excelu	125
17.1.1 Barvni efekti v pogojnih stavkih	127
17.1.2 Uporaba funkcije COUNTIF	128
17.1.3 Enačbe v pogojnem stavku	129
17.1.4 Uporaba logičnih operatorjev AND, OR in NOT	130
18. REŠEVANJE ALGEBRSKIH ENAČB Z ENO NEZNANKO	133
18.1 Graf funkcije	133
18.2 Orodje Goal Seek	134
18.3 Orodje Solver	135
19. REŠEVANJE SISTEMOV LINEARNIH ENAČB	141
19.1 Reševanje z inverzno matriko	141
19.2 Reševanje z orodjem Solver	142
19.2.1 Definiranje pogojev	142
19.2.2 Minimiranje odstopanj	144
20. NUMERICNA INTEGRACIJA	147
20.1 Trapezna formula ali trapezno pravilo	147
21. NAVADNE DIFERENCIALNE ENAČBE	151
21.1 Eulerjeva metoda	151
22. TISKANJE	159
22.1 Predogled tiskanja	159
22.2 Priprava strani	159
22.3 Tiskanje	160

PRILOGE

PRILOGA A: ASCII kode	163
PRILOGA B: VMS operacijski sistem	165
PRILOGA C: Funkcijske tipke	100
PRILOGA D. Rešitve problemov za 1. del	169
	173

1. DEL

Programiranje v fortranu

1. UVOD

Pri računalniškem programiranju moramo vedno imeti predstavo o dveh stvareh:

- o računalniškem programu in
- o računalniškem ali programskem jeziku.

Računalniški program predstavlja zaporedje stavkov oziroma informacij, ki podajajo določene operacije računalniku. Z računalniškim ali programskim jezikom zapišemo zaporedje stavkov oziroma informacij in jih posredujemo računalniku. Programski jeziki so npr. PASCAL, BASIC, fortran, C, C++, MATLAB itd. Mi bomo spoznali fortranski jezik.

1.1 Spoznavanje računalnika

Za lažje razumevanje poglejmo, kako sta povezana računalniški program in računalnik. Kaj je software?

Ko govorimo o softwaru mislimo na programsko opremo, ki nam je na voljo oziroma jo izdelamo sami. Gre za množico računalniških programov.

Kaj je hardware?

Hardware predstavlja fizični del računalnika, tj. strojno opremo.

Strokovnjaki pravijo: programska oprema je nadrejena (upravlja) strojni opremi. Zakaj? Poglejmo dva podobna primera iz življenja.

V prvem primeru si predstavljajmo simfonični orkester, ki ga sestavljajo razni instrumenti. V računalniškem besednjaku bi rekli, da je to hardware. Orkester lahko zaigra čudovite melodije šele, ko skladatelj (programer) zapiše melodijo v obliki notnega zapisa (software) in ga posreduje glasbeniku ali zaigra sam.

Drugi primer je prikazan na sliki 1.1. Kaj prikazuje? Kaj predstavlja programsko opremo in kaj strojno opremo?



Slika 1.1. Kaj je software in kaj hardware?¹

¹Middlebrooks C., VAX Fortran, Reston Publishing Company, Inc., Reston, 1984.

Kako sta povezana računalniški program in računalnik?

Primer 1.1: Preprosti fortranski program.

```
RATE = 4.00
HOURS = 3.0
PAY = RATE * HOURS
PRINT *, PAY
END
```

- program vtipkamo preko tipkovnice,
- vidimo ga na ekranu monitorja,
- kodirano obliko vtipkanih znakov pošljemo v računalnik,
- iz računalnika dobimo rezultat, tj. izračunano in izpisano vrednost za PAY.

V računalniku je niz elektronskih vezij za shranjevanje in prenos informacij. "Prostor" v katerega smo shranili naše informacije imenujemo spomin ali pomnilnik. Zelo enostavno si lahko pomnilnik zamislimo v obliki poštnih predalčkov, ki imajo svoja imena in v katere shranjujemo informacije zapisane na listu papirja. Vsak gornji stavek je zapisan na listu papirja in shranjen v poštni predal (slika 1.2). Strokovno pomnilnik predstavimo na način, zapisan v nadaljevanju.



Slika 1.2. Preprosta predstava pomnilnika¹.

1.1.1 Predstava spomina oziroma pomnilnika

Pomnilnik predstavljajo pomnilne celice, ki so sestavljene iz elementov. Vsak element lahko sprejme 1 bit informacije.

bit = <u>bi</u>nary digi<u>t</u> = dvojiška cifra, binarno število.

1 bit informacije je cifra 1 ali 0. Torej: z ničlami in enkami lahko zapišemo neko informacijo. Računalnik pretvori nam razumljiv zapis v sebi razumljiv zapis. Osnovna enota informacije, ki jo shranjujemo ali kličemo iz pomnilne celice je računalniška beseda. Beseda lahko obsega 16, 24, 30, 32, 36, 48 in 60 bitov. Na nekaterih računalnikih so besede s fiksno dolžino, ponekod s spremenljivo. Vsaka beseda ima svojo številko ali adreso.

¹Middlebrooks C., VAX Fortran, Reston Publishing Company, Inc., Reston, 1984.

8 bitov je 1 byte. Byte združujemo v besedo (word, 2 byta) ali v dolgo besedo (long word, 4 byti oziroma 32 bitov). 32 bitna beseda je sestavljena na naslednji način:

dolga beseda			
byte 3	byte 2	byte 1	byte 0

1.1.1.1 Binarno kodiranje

A) Nenumerične informacije

različne kombinacije 0 in 1

~~

Če kombiniramo samo 2 bita, tj. 0 in 1, potem dobimo 4 možne kombinacije zapisa, ki nam lahko prestavljajo samo 4 znake npr. A, B, +, in -.

00	\rightarrow	Α
01	\rightarrow	В
10	\rightarrow	+
11	\rightarrow	-

torej $2^2 = 4$. Če bi uporabili 3 bite za zapis znakov, bi lahko zapisali (kodirali) $2^3 = 8$ možnih kombinacij (znakov). Ker v fortranu uporabljamo okoli 50 različnih znakov, bo potrebnih vsaj 6 bitov za predstavitev vsakega znaka.

Primer 1.2 : n = 6 (število bitov) S šestimi biti lahko opišemo $2^n = 2^6 = 64$ znakov.

٨

Današnji računalniki uporabljajo v glavnem 8 bitne kode (ASCII Code \rightarrow <u>A</u>merican <u>S</u>tandard Code for Information Interchange oziroma ameriške standardne znake za izmenjavo informacij). Z osmimi biti lahko zapišemo 256 različnih znakov (28=256). Zapis ASCII kod je zbran v prilogi A.

B) Numerične informacije:

računalnik pretvori števila iz desetiškega sestava v binarni oziroma dvojiški sestav, kjer je vsako število predstavljeno z 1 in 0.

Pretvarjanje iz desetiškega sestava v dvojiški

Celi in decimalni del števila pretvarjamo ločeno.

cela števila, celi del realnega števila

Celi del pretvorimo tako, da ga delimo z osnovo 2. Ostanek označimo z a₀. Celoštevilčni del rezultata deljenja ponovno delimo z 2, ostanek je a1. Postopek ponavljamo, dokler ne pridemo do zadnjega ostanka a_n. To je takrat, ko postane količnik enak nič. Tedaj lahko število, pretvorjeno v sestav z osnovo 2, zapišemo takole:

 $a_n \dots a_1 a_0 (2)$

<u>Npr.:</u>

 $9_{(10)} = 1001_{(2)}$

<u>Izpeljava</u> :	9 = 4 · 2 + 1	a ₀
	$4 = 2 \cdot 2 + 0$	a 1
	2 = 1 · 2 + 0	a ₂
	1 = 0 · 2 + 1 ↑	a_3

$32_{(10)} = 100000_{(2)}$

<u>Izpeljava</u> :	32 = 16 · 2 + 0	a ₀
	16 = 8 · 2 + 0	a_1
	$8 = 4 \cdot 2 + 0$	a ₂
	$4 = 2 \cdot 2 + 0$	a ₃
	2 = 1 · 2 + 0	a_4
	1 = 0 · 2 + 1 ↑	a_5

• realna števila

Decimalni del števila pretvorimo v sestav z osnovo 2 tako, da ga množimo z osnovo 2. Pri množenju dobljeni celi del števila označimo z a₋₁. Nato ta celi del števila odrežemo in decimalni ostanek ponovno množimo z 2. Dobljeni celi del označimo a₋₂. Ta postopek ponavljamo, dokler se ne zadovoljimo s številom mest števila v novem zapisu oziroma dokler z množenjem decimalna mesta niso enaka nič. Nato pozicijsko zapišemo dobljeno število sestava z osnovo 2 takole²:

0,**a**₋₁ **a**₋₂ (2)

<u>Npr.:</u>

 $6,5625_{(10)} = 110,1001_{(2)}$

Izpeljava:	celi del		decimalni del	
	6 = 3 · 2 + 0	a₀	0,5625 · 2 = 1 ,1250 ↓	a.1
	3 = 1 · 2 + 1	\mathbf{a}_1	0,125 · 2 = 0 ,250	a.2
	1 = 0 · 2 + 1 ↑	a ₂	0,250 · 2 = 0 ,500	a.3
			$0,500 \cdot 2 = 1,0$	a.4
	$6_{(10)} = 110_{(2)}$		$0,5625_{(10)} = 0,1001_{(2)}$	

²Bratko I., Rajkovič V., Računalništvo s programskim jezikom PASCAL, Državna založba Slovenije, Ljubljana, 1984.

Pretvarjanje iz dvojiškega sestava v desetiški sestav

Vsako pozitivno realno število je možno zapisati na naslednji način:

$$N = a_n \cdot q^n + a_{n-1} \cdot q^{n-1} + \dots + a_1 \cdot q + a_0 + a_{-1} \cdot q^{-1} + \dots + a_{-m} \cdot q^{-m} + \dots$$

kjer je q osnova številskega sestava (v našem primeru 2). Za koeficiente velja omejitev $0 \le a_i < q$. V skladu z omejitvami za koeficiente a_i , sta binarni cifri 0 in 1. Število v dvojiškem sestavu torej razvijemo po gornji enačbi, nato pa uporabimo desetiško aritmetiko in izračunana vrednost N je vrednost števila v desetiškem sestavu².

Npr.:

$$101101_{(2)} = 1 \cdot 2^{5} + 0 \cdot 2^{4} + 1 \cdot 2^{3} + 1 \cdot 2^{2} + 0 \cdot 2^{1} + 1 \cdot 2^{0} =$$

32 + 0 + 8 + 4 + 0 + 1 = 45₍₁₀₎

in še:

$$101,011_{(2)} = 1 \cdot 2^{2} + 0 \cdot 2^{1} + 1 \cdot 2^{0} + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} = 5,375_{(10)}$$

1.2. Spoznavanje strojne opreme (Hardware)

K strojni opremi uvrščamo:

A) vhodne in izhodne enote, ki pretvarjajo informacije in povezujejo procesni del z okoljem npr. človekom.

Vhodne enote pretvorijo informacijo, ki je namenjena računalniku in je zapisana z našimi, standardnimi znaki, v ustrezna zaporedja binarnih električnih signalov. Nasprotno izhodne enote rezultate, izražene z zaporedji električnih signalov, pretvorijo v čim običajnejšo, človeku razumljivo obliko. Naštejmo le nekatere:

- tipkovnica, preko katere pošiljamo informacije v računalnik,
- **miška**, s katero pokažemo na določeno besedo, sliko ali ikono in tako računalniku povemo, kaj naj stori (omogoča hitro vnašanje ukazov),
- monitor ali zaslon, na katerem vidimo vtipkano našo informacijo ali računalnikovo sporočilo,
- tiskalnik, s katerim lahko izpišemo na papir informacije, ki jih vnesemo v računalnik in sporočila, ki jih računalnik pošilja. V glavnem poznamo tri vrste tiskalnikov:
 - iglične oziroma matrične (vrstične) tiskalnike, na katerih poteka tiskanje znakov na papir z mehanskim dotikom (z udarci iglic),
 - laserske tiskalnike, kjer poteka tiskanje brez mehanskega dotika, tj. na osnovi elektrokemijskih oziroma fotoelektričnih reakcij,
 - brizgalne (ink-jet) tiskalnike, ki delujejo na principu brizganja črnila. Kakovost tiska je podobna laserskemu izpisu, le da se sveže črnilo rado razmaže.

²Bratko I., Rajkovič V., Računalništvo s programskim jezikom PASCAL, Državna založba Slovenije, Ljubljana, 1984.

B) Računalniški pomnilniki (Memory)

Delimo jih na notranje in zunanje. Namenjeni so začasnemu oziroma trajnemu shranjevanju podatkov. Informacije lahko shranjujemo v:

- notranji ali glavni pomnilnik (RAM Random Access Memory in ROM Read Only Memory),
- zunanji pomnilnik na trdem disku, CD-romu, tračnih magnetnih enotah, disketah, USB prenosnih diskih-ključkih.

Glavni pomnilnik se nahaja znotraj računalnika. To je prostor, kjer je shranjen program, ki ga trenutno izvajamo. Čeprav ima lahko glavni pomnilnik veliko kapaciteto, je omejena in v njem ne moremo shranjevati programov trajno (v primeru motenj, npr. izpada elektrike, se programi, ki smo jih trenutno izvajali, uničijo).

Na trde diske in ostale enote, ki smo jih omenili, shranjujemo programe, ki jih trenutno ne potrebujemo. Če zmanjka elektrike ali nastopijo druge motnje na sistemu, se programi ne uničijo.

C) CPU-centralna procesna enota oziroma mikroprocesor

Osnovni del računalnika, s katerim smo povezani med delom, je centralna procesna enota (CPU) ali mikroprocesor. Na osebnih računalnikih (PC) ga označujejo kot 286, 386, 486, pentium itd. Predstavlja srce in možgane računalnika. Fizično je sestavljen iz niza elektronskih vezij. Sestavljajo ga:

- Aritmetična logična enota, kjer se izvajajo aritmetične operacije (seštevanje, odštevanje, množenje, deljenje), ter logične operacije (ugotavljanje enakosti, neenakosti, večjega, manjšega).
- Krmilna enota, ki nadzoruje in usklajuje delovanje posameznih enot tako, da se algoritem, ki je v obliki programa računalniku zaupan, pravilno izvede. Odvijanje programa v računalniku vodi tako, da analizira program korak za korakom. Glede na vrsto koraka sporoči, katera enota naj v danem trenutku deluje in kako naj deluje, da bo zagotovljeno pravilno izvajanje.
- **Glavni pomnilnik**, kjer shranjujemo informacije programa, ki ga trenutno izvajamo.

Vprašanja za preverjanje znanja:

- 1. Kaj je software?
- 2. Kaj je hardware?
- 3. Kaj je fortran?
- 4. Kaj je računalniški program?
- 5. Zakaj je programska oprema nadrejena strojni opremi?
- 6. Katere enote uvrščamo k strojni opremi računalnika?
- 7. Katere vhodne in izhodne enote poznate?
- 8. Naštejte vrste tiskalnikov in razložite princip delovanja?
- 9. Kako je sestavljena centralna procesna enota?

Problemi:

- **1.1.** Vadite binarno kodiranje. Pretvorite cela števila, 8, 15 in 24 desetiškega sestava v števila dvojiškega sestava!
- **1.2.** Vadite binarno kodiranje. Pretvorite realna števila 3,25 ter 12,4 in 2,475 v števila dvojiškega sestava!
- **1.3.** Vadite binarno kodiranje. Pretvorite število 110,111 dvojiškega sestava v število desetiškega sestava!

2. OSNOVE

Fortran je programski jezik. Njegova zgodovina sega v sredino prejšnjega stoletja, vendar ga še vedno dopolnjujejo in razvijajo.

2.1 Predstavitev fortrana

Fortran je bil prvi visoko razviti programski jezik, ki so ga uporabljali v znanosti, predvsem v tehniki. Beseda fortran je nastala iz začetnic besed **for**mula **tran**slation. Ime je ponazarjalo namen uporabe jezika in sicer v začetku predvsem za numerične izračune. Kasneje in danes ga lahko uporabljamo tudi za reševanje splošnih primerov.

Kratka zgodovina:

- leta 1954 so ga razvili v firmi IBM pod imenom Fortran IV,
- naslednja verzija je bila izdelana leta 1966. Tako je nastal standard FORTRAN66 (standardizirali so ga na: American National Standards Institute ⇒ ANSI),
- naslednja, izpopolnjena vezija je bila izdelana in standardizirana leta 1977 pod imenom Fortran77 (standardizirali so ga na: American National Standards Institute in mednarodni organizaciji za standardizacijo ISO),
- novejše, visoko razvite verzije so Fortran90 in Fortran95 ter Fortran2003 in Fortran2008.

Pri našem programiranju bomo uporabljali enostavnejšo verzijo fortrana, katerega osnova je Fortran77. V tem primeru bomo stavke zapisovali z naslednjimi znaki:

- črkami angleške abecede od A do Z,
- desetiškimi ciframi od 0 do 9,
- posebnimi znaki in sicer = + * / (), . ': < > presledek.

Fortranski program je sestavljen iz stavkov. Lahko so izvršljivi in neizvršljivi. Neizvršljivi stavki podajo neko navodilo prevajalniku in označujejo takšne lastnosti, ki jih v začetku določimo in veljajo ves čas izvajanja programa³. Neizvršljivi stavki so npr. REAL, INTEGER, CHARACTER in END.

Izvršljivi stavki opravljajo razne izračune ali prenose podatkov, katerih rezultat je odvisen od prejšnjih izvršljivih stavkov. Izvršljive stavke izvršujemo po vrsti, kot so zapisani, razen če tega ne spremenimo s krmilnimi stavki³. Izvršljivi stavki so npr. CONTINUE, READ in WRITE.

Poleg izvršljivih in neizvršljivih stavkov imamo v programu prisotne tudi aritmetične prireditvene stavke, ki jih potrebujemo za reševanje enačb.

³Žumer V., Fortran, TF Maribor, 1994.

Aritmetični prireditveni stavek.
<u>Splošna oblika</u> : s = ai
kjer je :
 s – ime spremenljivke, = – znak za prireditev in ai – aritmetični izraz.
Vrednosti na desni strani stavka morajo med izračunavanjem biti znane.

Razčlenimo fortranski program, ki ga prikazuje primer 2.1.

Primer 2.1: Izračun mase.

Znana je odvisnost med maso, gostoto in prostornino snovi. Npr., če želimo izračunati maso neke snovi, je odvisnost naslednja: $m = \rho \cdot V$. Zapišimo enačbo v fortranu in izpišimo vrednost mase snovi z gostoto 1000 g/L in prostornino 100 L !

```
RO=1000.0
VOLUMEN=100.0
XMASA=RO*VOLUMEN
PRINT *,XMASA
END
```

PRINT stavek lahko zapišemo bolj nazorno tako, da uporabimo opisne oznake za izpis:

PRINT *,'Masa=', XMASA

Določite vrste stavkov v programu:

Neizvršljivi stavki so:

Izvršljivi stavki so:

Aritmetični prireditveni stavki so:

In kako pričnemo s pisanjem programa? Pogovorite se o tem na predavanjih!

2.2 Konstante in spremenljivke

Konstanta je numerična vrednost, ki jo v programu definiramo. V stavku **RO=1000.0** je vrednost 1000.0 konstanta. Spremenljivka je neka vrednost, ki je pod določenim imenom shranjena v pomnilniku in se med izvajanjem lahko spreminja. Imena spremenljivk in konstant so ponazorjena z znaki, ki jih fortran pozna (največ 31 znakov).

2.3 Aritmetični izrazi

Aritmetični izraz je lahko konstanta, spremenljivka ali skupina konstant in spremenljivk, ki so povezane z aritmetičnimi operatorji in oklepaji. Izpeljan je po pravilih fortranskega jezika.

Aritmetični operator je simbol, ki določi vrsto aritmetične operacije. Aritmetične operacije so definirane z naslednjimi operatorji:

- + seštevanje,
- odštevanje,
- * množenje,
- ** potenciranje in
- / deljenje.

2.3.1 Vrstni red izvajanja aritmetičnih operacij v aritmetičnem izrazu

- a) **
- b) * in /
- c) + in –

Množenje in deljenje sta enakovredni operaciji, prav tako seštevanje in odštevanje. Če sta v aritmetičnem izrazu prisotni enakovredni operaciji, se izračun izvaja od leve proti desni.

Če so v aritmetičnih izrazih prisotni oklepaji je vrstni red naslednji:

- a) ()
- b) **
- c) * in /
- d) + in –

V primeru 2.2 sta prisotna dva aritmetična izraza.

Primer 2.2: Zaporedje izvajanja aritmetičnih operacij.

a) $x = b + d - \frac{c}{a}$ b) $x = a \cdot \frac{d}{b} \cdot c^{a} - d$

A=2 B=4 C=10 D=12 X=B+D-C/A PRINT *,X X=A*D/B*C**A-D PRINT *,X END

Vrstni red izvajanja aritmetičnih operacij je naslednji:

• prepišimo prvi aritmetični izraz z vhodnimi podatki. Vsa števila so realna.

4 + 12 - 10/2

• najprej se izvede deljenje. Po delenju ostane:

4 + 12 – 5

• seštevanje in odštevanje sta v aritmetičnem izrazu enakovredni operaciji. Izračun se izvrši od leve proti desni. Po seštevanju ostane:

16 – 5

• Rezultat je 11.

Ker smo operirali z realnimi števili in je tudi rezultat realno število, se na ekranu izpiše: **11.00000**.

Sedaj poskusite sami!

Primer 2.3: Zaporedje računanja, če so prisotni oklepaji.

a)
$$x = b + \frac{(d-c)}{a}$$

b) $x = \left(a \cdot \frac{d}{b} \cdot c\right)^a - d$
c) $x = \left(\frac{(a.d)}{(b.c)}\right)^a - d$

A=2 B=4	Koliko aritmetičnih izrazov je v programu?
D=12	Kje se izpiše rezultat?
X=B+(D-C)/A PRINT *,X X=(A*D/B*C)**A-D PRINT *,X X=((A*D)/(B*C))**A-D	Rezultat je:
PRINT *,X END	

2.4 Cela in realna števila

Tip števila shranimo v pomnilniku pod določenim imenom. Če se imena konstant in spremenljivk pričnejo s črkami od I do N, shranjujemo v pomnilniku cela števila. Cela števila uporabljamo v glavnem kot števce.

Če se imena konstant in spremenljivk pričnejo s črkami od A do H in od O do Z potem v pomnilniku shranjujemo realna števila. Za realna števila je značilna decimalna pika in ostanek za njo.

Primer 2.4: Cela števila.

N=20 M=3 L=N*M PRINT *,L J=N/M PRINT *,J END Rezultat je:

Primer 2.5: Računanje z realnimi števili.

```
BIGN=8765.99

SMALLN=0.039

CALC=BIGN*SMALLN

DIVISION=BIGN/SMALLN

PRINT *,'CALC=',CALC

PRINT *,'DIVISION=',DIVISION

END
```

Primer 2.6: Mešana števila.

```
X=33.9
Y=3.0
N=X/Y
PRINT *,N
I=5
J=9
D=212.0
T=I/J*(D-32.0)
PRINT *,T
I=5
U=9
D=212.0
T=I/U*(D-32.0)
PRINT *,T
END
```

Rezultat je:

2.4.1 Stavka INTEGER in REAL

Definirata način shranjenja vrednosti v pomnilniku preden izvršimo izračun. Če se imena spremenljivk in konstant začnejo s črkami od A do H in od O do Z, bi pa radi, da ponazarjajo celo spremenljivko oziroma konstanto, lahko definiramo ta imena s stavkom INTEGER.

Nasprotno lahko imena spremenljivk in konstant, ki se začnejo s črkami od I do N, definiramo kor realna števila s stavkom REAL.

Stavka INTEGER in REAL zapišemo vedno pred prvim izvršljivim stavkom v programu.

Primer 2.7: Uporaba stavkov REAL in INTEGER .

REAL N INTEGER Z X=33.9 Y=3.0 N=X/Y Z=X/Y PRINT *,'N=',N,'Z=',Z END Rezultat je:

Problemi:

2.1. Pri vsakem imenu označite z I, če ime predstavlja celo število oziroma z R, če predstavlja realno število:

NUM	CAS	J	CELOTA
XI	XYZ	IME	LAN

2.2. Ali naslednji zapisi veljajo za cele spremenljivke? Če ne, zakaj?

NEXT	
FIRST	
L124	
J+329	
2145	
N(3)M	

2.3. Ali naslednji zapisi veljajo za realne spremenljivke? Če ne, zakaj?

2.4. V fortranu zapišite naslednje matematične izraze:

a)	$(x + y) \cdot (u + v)$	
b)	3 (x + y)	
c)	$3 x \cdot y^2 - 2 \cdot x^2 \cdot y$	
d)	$a + \frac{b}{c^2}$	
e)	$\frac{a+b}{c \cdot d}$	

2.5. Predpostavimo naslednje fortranske stavke:

A=B+C	
B+C=A	
A=A+A	
A+B=C+D	
AB=CD	
A=ABS(A)	

Kateri so fortransko nesprejemljivi stavki? Predpostavimo, da so vse spremenljivke na desni strani enačaja (= znaka) definirane, tj. njihove vrednosti so že v pomnilniku pod določenim imenom.

2.6. Predpostavimo naslednje vrednosti za A, B, J in K:

A = 2.7 B = 3.5 J = 3 K = -2

Najdite vrednosti za X in L!

a)	X = A + J * K**2 + B L = A + J * K**2 + B	
b)	X = 5 * J/4 * K L = 5 * J/4.0 * K	
c)	X = K/3 * A/2 L = K/3 * A/2	

Vprašanja za preverjanje znanja:

- 1. Iz katerih stavkov je sestavljen fortranski program?
- 2. Kaj je značilno za neizvršljive stavke? Naštejte jih!
- 3. Kaj je značilno za izvršljive stavke? Naštejte jih!
- 4. Kako zapišemo aritmetični prireditveni stavek?
- 5. S katerimi črkami se prične ime konstante oziroma spremenljivke, če shranjujemo v pomnilnik cela števila?
- 6. S katerimi črkami se prične ime konstante oziroma spremenljivke, če shranjujemo v pomnilnik realna števila?
- 7. Kaj definiramo s stavkoma REAL in INTEGER?
- 8. Kaj je aritmetični izraz?
- 9. Katere aritmetične operatorje poznate?
- 10. Kakšen je vrstni red izvajanja aritmetičnih operacij v aritmetičnih izrazih?

3. POMEMBNE (matematične) FUNKCIJE

Imenujejo jih tudi notranje, vgrajene, standardne ali bibliotečne funkcije. Predstavljajo vrsto informacij, ki so shranjene v knjižnici pomembnih funkcij in so sestavni del fortranskega jezika. Pri klicu funkcije navedemo ime funkcije in v oklepaju argument ali več argumentov, če je potrebno. Najbolj znane funkcije so:

Matematični zapis ali namen uporabe	Ime funkcije z argumenti	Vrsta argumenta
\sqrt{x}	SQRT(X)	realno število > ali = 0.
e ^x	EXP(X)	realno število
ln (x)	LOG (X)	realno število > 0.
log (x)	LOG10 (X)	realno število > 0.
	ABS(X)	realno število
krajšanje, pretvorba v celo število	INT (X)	realno število
največja vrednost	MAX (X1,X2,X3) ali	cela ali realna števila
Max(i ₁ ,i ₂ ,)	MAX (I1,I2,I3)	
najmanjša vrednost	MIN (X1,X2,X3,X4) ali	cela ali realna števila
$Min(x_1, x_2, \ldots)$	MIN (I1,I2,I3)	
Min(i ₁ ,i ₂ ,)		
najbližje celo število	NINT (X)	realno število
pretvarjanje v realno število	REAL (I)	celo število
sin (x)	SIN (X)	realno število
cos (x)	COS (X)	realno število
tg (x)	TAN (X)	realno število
arc sin (x)	ASIN (X)	realno število
arc cos (x)	ACOS (X)	realno število
arc tg (x)	ATAN (X)	realno število

Opomba: Pri kotnih funkcijah je kot v radianih.

Primer 3.1: Primer uporabe funkcij SQRT in REAL.

Rezultat je:

X=22.2 Y=44.4 A=SQRT(X) PRINT *,A B=SQRT(X+Y) PRINT *,B N=67 C=SQRT(REAL(N)) PRINT *,C END

Primer 3.2: Uporaba funkcij LOG in LOG10.

```
XV=100.
A=LOG(XV)
PRINT *,A
B=LOG10(XV)
PRINT *,B
END
```

Rezultat je:

Problemi:

3.1. V fortranu napišite naslednje izraze in pri tem uporabite ustrezne matematične funkcije:

a) $\sqrt{5 \cdot x^2 + 8 \cdot y^2}$	
b) $sin(x-2 \cdot y) + e^{x \cdot y} - x^2 - y^2 $	
c) $e^{ a } - \frac{b^2}{ c }$	
d) $\ln(x + y)^2$	
e) $\log(a-b)^2$	

- **3.2.** Napišite računalniški program za naslednji primer: Imamo pravokotni trikotnik, za katerega velja $x^2 + y^2 = r^2$ oziroma $r = \sqrt{x^2 + y^2}$. Izračunajte in izpišite dolžino hipotenuze r, če sta dolžini stranic x = 3 in y = 4 in uporabite ustrezno matematično funkcijo!
- **3.3.** Napišite računalniški program, v katerem izračunajte in izpišite cos 60° in sin 30°. Stopinje pretvorite v radiane. Velja: $180^\circ = \pi = 3,141593$ radianov.

Namig: Eksperimentirajte s fortranom, kajti najboljša pot, da se ga naučite je, da ga uporabljate!

Vprašanja za preverjanje znanja:

- 1. Zakaj imenujemo matematične funkcije tudi vgrajene, notranje, standardne oziroma biliotečne funkcije?
- 2. Naštejte nekaj pomembnih matematičnih funkcij, ki so v fortranski knjižnici!
- 3. Kdaj uporabljamo funkciji log in log10?
- 4. Kdaj uporabljamo funkcijo sqrt ?
- 5. Kaj je značilno za funkciji min in max?

4. IZPISOVANJE IN BRANJE PODATKOV

4.1 Stavek WRITE – pisanje rezultatov oziroma sporočil preko terminala

Poleg stavka PRINT lahko za izpisovanje rezultatov uporabljamo stavek WRITE.

<u>Splošna oblika</u> :
WRITE (le, f) seznam
 le – številka logične enote preko katere želimo pisati rezultate, npr. 6 za terminal, f – oznaka stavka FORMAT ali zvezdica za prosti format, seznam – spremenljivke, znakovni izrazi in konstante, ki jih izpisujemo in jih med seboj ločimo z vejico.

Primer 4.1: Izpis preko terminala (številka logične enote je 6).

```
C1=780.00
C2=220.00
TOTALC=C1+C2
WRITE (6,*)TOTALC
END
```

Rezultat je:

Izpišete lahko tudi vhodne podatke. Kako bi definirali stavek WRITE v tem primeru?

WRITE (6,*)

4.2. Stavek READ – branje podatkov preko terminala (interaktivni vnos podatkov)

<u>Splošna oblika</u> :			
	READ (le, f) seznam		
le –	številka logične enote preko katere želimo brati podatke, npr. 5 za terminal,		
f – seznam –	številka stavka FORMAT ali zvezdica za prosti format, spremenljivke, ki sprejmejo vhodne podatke in jih med seboj ločimo z vejico.		

Slika 4.1 prikazuje vnos vhodnih podatkov preko terminala in izpis rezultatov preko terminala (na ekran).



Slika 4.1: Branje in izpisovanje preko terminala¹.



```
READ(5,*)C1
READ(5,*)C2
TOTALC=C1+C2
WRITE(6,*)C1
WRITE(6,*)C2
WRITE(6,*)TOTALC
END
```

Rezultat je:

4.3 Izpis na datoteko

 datoteka zajema vrsto informacij (programi, vhodni podatki, rezultati) shranjenih v pomnilniku.

Stavek OPEN:

- s stavkom specificiramo datoteko, na kateri so vhodni podatki oziroma rezultati,
- za izpis rezultatov se uporablja v kombinaciji s stavkoma WRITE in CLOSE (zapremo datoteko po izvajanju).

¹Middlebrooks C., VAX Fortran, Reston Publishing Company, Inc., Reston, 1984.

Splošna oblika:

OPEN (le, FILE = 'ime.tip', STATUS = 'NEW')

le – številka logične enote preko katere želimo izpis*,

FILE= – specificiramo ime datoteke, na katero želimo izpisovati rezultate npr. ime.rez, ime.out itd.,

STATUS = – odpiranje "nove" datoteke z rezultati, ki še ne obstajajo (NEW).

* Če v stavku OPEN zapišemo številko logične enote 6, je izpis rezultatov preko datoteke.

Primer 4.3: Izpis rezultatov na datoteko REZUL.DAT.

```
OPEN(60,FILE='REZUL.DAT',STATUS='NEW')
C1=780.00
C2=220.00
TOTALC=C1+C2
WRITE(60,*)TOTALC
CLOSE(60)
END
```

Rezultat je:

4.4 Branje z datoteke

• koristno, kadar vnašamo večje število podatkov.

Podatke vtipkamo v datoteko in jo shranimo, za tem jo lahko popravljamo in preglejujemo. V bistvu je vhodna datoteka nekega programa. V teh primerih uporabljamo kombinacijo stavkov READ in OPEN ter stavka CLOSE.

Splošna oblika:			
	OPEN (le, FILE = 'ime.tip', STATUS = 'OLD')		
le – FILE = – STATUS =	številka logične enote preko katere beremo*, ime datoteke na kateri so vhodni podatki, npr. ime.dat , ime.inp itd., – branje podatkov iz obstoječe datoteke (OLD).		

* Če v stavku OPEN zapišemo številko logične enote 5, je vnos podatkov preko datoteke.

Slika 4.2 prikazuje primer branja podatkov preko datoteke BOTHCOST.DAT in izpis rezultatov na datoteko TOTALOUT.DAT.



Slika 4.2: Branje z datoteke in pisanje na datoteko¹.

Primer 4.4: Vpis vhodnih podatkov na datoteko VHODNI.DAT.

780.0,220.0 - vhodna datoteka VHODNI.DAT

OPEN(50,FILE='VHODNI.DAT',STATUS='OLD')
READ(50,*)C1,C2
TOTALC=C1+C2
WRITE(6,*)TOTALC
CLOSE(50)
END

Rezultat je:

4.5 Komentarji v programu

- razložimo bistvo programa, spremenljivke, vhodne podatke itd.,
- posebej so pomembni, če program uporabljamo v časovnih zamikih (pozabimo kaj in kako računa),
- vnašamo dodatne informacije npr. o avtorju, ustanovi, kjer je bil razvit itd.

Komentar je fortranski stavek, ki se prične s črko **c** (tudi * in !) v prvem stolpcu. Fortranski prevajalnik ga ne zazna (ga ignorira). Lahko ga pišemo z velikimi ali malimi črkami. Za znakom ! lahko pišemo komentar od poljubnega stolpca dalje.

¹Middlebrooks C., VAX Fortran, Reston Publishing Company, Inc., Reston, 1984.

Primer 4.5: Komentarji v programu.

```
С
C
     NAPISANO DNE: 29. AVGUST 2008
C
     NAPISAL: M. KRAJNC
С
     USTANOVA: FKKT, MARIBOR
С
C
     PROGRAM RACUNA CENO SOLSKIH POTREBSCIN
C
     SPREMENLJIVKE
С
     C1 - cena knjig
C
     C2 - cena zvezkov
     C1=780.0
     C2=220.0
C
     TOTALC je celotna cena potrebscin
     TOTALC=C1+C2
С
     IZPIS REZULTATA
     WRITE (6,*)TOTALC
     END
```

Rezultat je:

Problemi:

4.1. Preglejte naslednje programe in razložite, od kod dobimo vhodne podatke oziroma kje so dosegljivi rezultati po izvajanju programa!

READ(5,*)URE READ(5,*)VREDNOST PLACILO=VREDNOST*URE WRITE(6,*)PLACILO END	Vnos podatkov pro	eko
OPEN(10,FILE='RH.DAT',ST READ(10,*)URE,VREDNOST PLACILO=URE*VREDNOST WRITE(6,*)PLACILO END	'ATUS='OLD')	Vnos podatkov preko Izpis rezultatov preko

4.2. V stavku OPEN, ki je povezan s stavkom READ, moramo pri možnosti STATUS datoteke vedno definirati STATUS = '......', ker program ne more brati podatkov z datoteke, ki še ne obstaja.

.

- **4.3.** V stavku OPEN, ki je povezan s stavkom WRITE, moramo pri možnosti STATUS datoteke vedno definirati STATUS = '......', ker program piše rezultate na datoteko, ki še ne obstaja.
- **4.4.** Preoblikujte naslednji program na naslednja načina:

CENA=0.9 STEVILO=10 SKUPAJ=CENA*STEVILO WRITE(6,*)SKUPAJ END

- program naj pošilja rezultat na datoteko IME.REZ, vhodne podatke naj bere preko terminala,
- program naj pošilja rezultat na datoteko IME.REZ, vhodne podatke pa naj bere z datoteke IME.DAT (opomba: datotekam dajte poljubno ime).

Vprašanja za preverjanje znanja:

- 1. Kdaj in kako uporabljamo stavka READ in WRITE?
- 2. Kaj je stavek OPEN in kako ga zapišemo?
- 3. S katerimi stavki povezujemo stavek OPEN?
- 4. Kaj pomenita besedi OLD in NEW v stavku OPEN?
- 5. Kakšna je prednost stavka WRITE pred stavkom PRINT?
- 6. Zakaj je koristno uporabljati komentarje?
- 7. Kako označimo komentar v programu?
- 8. Kaj pomeni interaktivni vnos podatkov?
- 9. Katero številko logične enote definiramo v stavku READ, ko želimo vnašati podatke interaktivno?
- 10. Katero številko logične enote definiramo v stavku WRITE, ko želimo izpis rezultatov na ekranu?
5. SISTEMATIČNI PRISTOP K UČINKOVITEMU PROGRAMIRANJU

Upoštevamo naslednje stopnje:

- a) Definiranje problema.
- b) Analizo problema:
 - definiramo informacije, ki jih imamo na voljo,
 - definiramo rezultate, ki jih želimo dobiti.
- c) Izdelavo algoritma in diagrama poteka (flowchart):
 - podajata logične stopnje, preko katerih pridemo do rezultata.
- d) Izdelavo fortranskega programa.

Na primeru poglejmo, kako pristopimo k posameznim stopnjam.

5.1 Problem

100 g dušika polnimo v posodo pri temperaturi 23 °C in tlaku 1,2 atm. Predpostavimo idealno obnašanje plina. Najprej preračunajte tlak v enoto bar nato pa izračunajte množino dušika ter prostornino posode (v litrih), ki je potrebna, da jo napolnimo z navedeno količino plina pri danih pogojih.

5.2 Analiza problema

Zberemo znane podatke:

- a) pretvornik med atm in bar (p_{pret}),
- b) splošno plinsko konstanto (R),
- c) maso dušika (m),
- d) temperaturo (t)
- e) tlak v atm (p_{atm}),
- f) molsko maso dušika (M_{dusik}),
- g) vnesemo podatke posameznih veličin.

Rezultati, ki jih želimo dobiti:

- h) izračunamo tlak v enoti bar (p_{bar}) ,
- i) izračunamo množino dušika (*n*),
- j) izračunamo temperaturo z enoto K (T),
- k) izračunamo prostornino posode (V),
- I) izpišemo rezultate za p_{bar} , n, T in V.

5.3 Algoritem in diagram poteka

Algoritem:

- a) vnos pretvornika za tlak, splošne plinske konstante, mase dušika, temperature, tlaka in molske mase,
- b) izračun tlaka v enoti bar (p_{bar}): tlak v atm × pretvornik,
- c) izračun temperature v enoto K (T): temperatura v enoti °C + 273.15,
- d) izračun množine dušika (n): masa / molska masa,
- e) izračun prostornine posode (*V*): (množina dušika × splošna plinska konstanta × temperatura (*T*)) / tlak (p_{bar}),
- f) izpis tlaka v bar, množine v mol, temperature v K in prostornine v L.

Nekateri grafični znaki oziroma simboli za risanje diagramov poteka:

\bigcirc	- začetek (start), konec (stop)
	- prikaz enačb za izračun (brez odločitev)
\diamond	- odločitve
	- vhod (read) in izhod (write)
\bigcirc	- začetek zanke DO
\bigcirc	- povezave posameznih delov programa, konec zanke

Slika 5.1 prikazuje diagram poteka za problem 5.1.

DO



Slika 5.1: Diagram poteka

5.4 Računalniški program

Primer 5.1: Primer programa v fortranu.

1.01325,0.08314,100,23,1.2,28 - vhodna datoteka PODAT20.DAT

```
REAL N,M,MDUSIK
      OPEN(10,FILE='PODAT20.DAT',STATUS='OLD')
C
C
     Vnos podatkov
C
     PPRET je pretvornik med atm in bar,
С
     R je splosna plinska konstanta v L bar/mol K,
С
     M je masa dusika v g, PATM je tlak v atm,
     MDUSIK je molska masa dusika
C
C
     READ (10,*)PPRET,R,M,T,PATM,MDUSIK
С
С
     Izracun posameznih velicin
C
     PBAR=PATM*PPRET
     N=M/MDUSIK
      TK=T + 273.15
     V=(N*R*TK)/PBAR
C
C
     Izpis rezultatov
C
     WRITE(6,*)' Tlak (bar) Mnozina (mol) T (K) Prostornina (L)'
      WRITE (6,*) PBAR, N, TK,V
      END
```

Rezultat je:

Vprašanja za preverjanje znanja:

- 1. Katere stopnje prikaza problema je primerno izvesti, da bi bilo programiranje res učinkovito?
- 2. Kaj je diagram poteka?
- 3. Kdaj uporabljamo grafični simbol romb?
- 4. Kaj zapisujemo v grafični simbol pravokotnik?
- 5. S katerima grafičnima simboloma prikažemo začetek in konec zanke DO?
- 6. Kdaj uporabljamo grafični simbol paralelogram?
- 7. S katerim grafičnim simbolom prikažemo začetek in konec diagrama poteka?

6. ZANKE

6.1 Zanka DO s števcem

• izvajanje zaporedja informacij oziroma fortranskih stavkov večkrat.

<u>Splošna oblika:</u> DO ime_števca = začetna_vrednost, končna_vrednost ponavljajoči stavki (telo zanke DO) END DO

V zanki DO se izvajajo enačbe oziroma stavki, ki so med stavkoma DO in END DO in sicer tolikokrat, kolikokrat je bilo v števcu podano. Števci so ponavadi cela števila. Pri vsakem izvajanju zanke prištejemo število 1 k vrednosti **ime_števca**, vse dokler ne dosežemo **končne_vrednosti**. Po končanem izvajanju zanke DO program nadaljuje izvajanje s prvim stavkom po zanki.

Opomba: Zaradi preglednosti je telo zanke malo zamaknjeno v desno.

Grafični prikaz zanke DO:



Primer 6.1: Primer izračuna celotne vrednosti (vsote) z zanko DO.

10 40 - vhodna datoteka PODAT2.DAT 15	25 30 10 40 15	- vhodna datoteka PODAT2.DAT
---	----------------------------	------------------------------

OPEN (50,FILE='PODAT2.DAT',STATUS='OLD')
TOTAL=0
DO ISTEV=1,5
 READ(50,*)CENA
 TOTAL=TOTAL+CENA
END DO
WRITE(6,*)'KONCNA VREDNOST',TOTAL
END

Primer 6.2: Zanka DO s števcem kot spremenljivko.

Program prikazuje spremembo tlaka idealnega plina s spremembo množine plina pri konstantni temperaturi in prostornini. Množina plina se spreminja po 1 mol. Vhodni podatki na datoteki TLAK.DAT so: vrednost splošne plinske konstante, temperatura in prostornina.

0.08314 273.15 22.4

```
open(10,file='tlak.rez',status='new')
open(20,file='tlak.dat',status='old')
read (20,*)r,t,v
write(10,*)'Sprememba tlaka v odvisnosti od mnozine snovi'
do n=1,5
    p=r*T*n/v
    write(10,*)'n=',n,' mol','    p=',p,' bar'
end do
end
```

Rezultat na datoteki TLAK.REZ:

Sprememba tlaka v odvisnosti od mnozine snovi n= 1 mol p= 1.013825 bar bar n= 2 mol p= 2.027651 3 mol p= 3.041476 bar n= 4 mol 4.055302 bar n= p= 5 mol 5.069128 bar n= p=

Primer 6.3: Preračunavanje °F v °C.

```
PETDEVETIN=5.0/9.0
DO STF=0.0,212.0
STC=PETDEVETIN*(STF-32.0)
WRITE(6,*) STF, STC
END DO
END
```

Rezultat je:

6.2 Zanka DO z inkrementom ali prirastkom

Splošna oblika:

DO ime_spremenljivke = začetna_vrednost, končna_vrednost, prirastek ponavljajoči stavki (telo zanke DO) END DO

S prirastkom določimo, kako se spreminja začetna vrednost. Začetna vrednost, končna vrednost in prirastek so lahko števila, imena, lahko jih izračunamo nekje v programu, lahko jih beremo s stavkom READ itd. Pojavljajo se lahko kot cela ali realna števila.

Primer 6.4: Zanka DO z inkrementom ali prirastkom.

MIN=14MAX=26 INC=4 DO I=MIN, MAX, INC WRITE(6,*)I END DO DO I=MIN*2, MAX+6, INC/2 WRITE(6,*)I END DO XMIN=0.5 XMAX=1.5 XSTEP=0.5 DO XVALUE=XMIN, XMAX, XSTEP WRITE(6,*) XVALUE END DO DO XVALUE=XMAX*2, XMIN+1.0, -0.5 WRITE(6,*) XVALUE END DO END

Koliko zank DO je v programu?

.....

Rezultat je:

6.3 Stavek GO TO

• omogoča brezpogojni skok na drugi stavek programa.

Splošna oblika:GO TO številka_stavka

Primer 6.5: Uporaba brezpogojnega stavka GO TO.

```
N=5
M=2
WRITE(6,*)' PRVI WRITE STAVEK', N,M
GO TO 60
N=0
M=0
WRITE(6,*)' DRUGI WRITE STAVEK', N,M
60 WRITE(6,*)'TRETJI WRITE STAVEK', N,M
END
```

Ali se na ekranu pojavi izpis DRUGI WRITE STAVEK ?

Primer 6.6: Uporaba stavka GO TO v zanki.

10 READ(5,*)NUMBER WRITE(6,*)NUMBER GO TO 10 END Rezultat je:

Rezultat je:

Opomba: Številko stavka READ zapišemo od prvega do petega stolpca.

Kako postopamo v primeru, ko "se zazankamo"?

Problemi:

- 6.1. S fortranskim programom izpišite svoje ime 10 krat!
- 6.2. Prikažite rezultate, ki jih dobite z naslednjimi fortranskimi programi:

```
LCELOT=0
DO LZANKA=1,3
LCELOT=LCELOT+LZANKA
PRINT *,LZANKA, LCELOT
END DO
END
```

```
DO I=3,8
IVREDNOST=I - 3
PRINT *,I,IVREDNOST
END DO
END
```

Rezultat je:

Rezultat je:

```
DO KSTEVEC=100,2000,100
LSTOLETJA = ((KSTEVEC-100)/100)+1
PRINT *,'STOLETJE:', LSTOLETJA
END DO
END
```

Rezultat je:

Rezultat je:

```
L=3
M=6
N=24
INDEKSDOD=10
ISKUPAJ=0
DO INDEKS=M,N,L
ICELOT=INDEKS+INDEKSDOD
PRINT *,INDEKS, ICELOT
ISKUPAJ=ISKUPAJ+ICELOT
END DO
PRINT *,ISKUPAJ
END
```

- **6.3.** Napišite program z zanko DO, v katerem izpišite vsa neparna števila (1, 3, 5,99) manjša od števila 100 !
- **6.4.** Napišite program s pretvornikom, kjer milje spremenite v kilometre. Pričnite z 10 mile in uporabite inkrement 10. Pretvarjanje vršite do 100 mile. Pretvornik je: 1 mile = 1,6 km. Rezultat izpišite v obliki tabele.
- **6.5.** Za cela števila od 1 do 15 napišite program, v katerem boste izpisali tabelo števil ter kvadratne in kubične vrednosti teh števil!

Vprašanja za preverjanje znanja:

- 1. Kdaj uporabljamo zanko DO?
- 2. Kako je zgrajena zanka DO?
- 3. Kateri grafični simbol uporabljamo za začetek zanke DO?
- 4. Kateri grafični simbol uporabljamo za konec zanke DO?
- 5. Kaj pomeni zapis npr. GO TO 10?
- 6. Kako prekinemo izvajanje programa v primeru "zazankanja"?

7. STAVEK FORMAT in FORMATNA DOLOČILA

• uporabljamo jih za branje in pisanje numeričnih in znakovnih podatkov v želeni obliki.

7.1 Stavek FORMAT

• specificira način zapisa vhodnih in izhodnih podatkov.



Stavek FORMAT se povezuje s stavkoma WRITE in READ s številko stavka. Krmilni znaki se uporabljajo pri izpisovanju. Pri našem delu bomo uporabljali krmilni znak '' (praznina med apostrofi), ki prepreči izgubo prvega znaka pri izpisovanju. Namesto tega znaka lahko uporabimo formatno določilo X.

7.2 Različna formatna določila

Določilo F

• omogoča zapis realnih števil v decimalni obliki. Pri izpisu so števila desno poravnana.

<u>Splošna oblika:</u>

Fw.d

w – skupno število razpoložljivih mest za realno število,

d – število decimalnih mest.

Določilo E

• omogoča zapis realnih števil v eksponentni obliki.

<u>Splošna oblika:</u>

Ew.d

- w skupno število razpoložljivih mest za realno število,
- d število decimalnih mest.

Določilo I

• omogoča zapis celih števil. Izpisana vrednost je desno poravnana.



Faktor ponavljanja (repetition factor)

Številka pred določili I, F in E pove, kolikokrat želimo uporabiti določen format (določilo) za izpis.

Določilo X

• ponazarja praznine med izpisanimi vrednostmi (premik za n mest desno).

Splošna oblika:

nΧ

n – cela pozitivna konstanta.

Določilo A

 omogoča zapis oziroma prenos alfanumeričnih znakov oziroma znakovnih podatkov. Izpisano besedilo je levo poravnano. Pri uporabi znakovnih podatkov je potrebno uporabiti stavek CHARACTER.



Poševna črta / (slash)

 v stavku FORMAT povzroči preskok v naslednjo vrstico. N poševnih črt pomeni (n-1) praznih vrstic. Primer 7.1: Uporaba različnih formatnih določil.

```
I=2
     J=23
     N=356
     X=27.87
     Y=179.003
     Z=0.00567
     WRITE(6,100)I,J,N,X,Y,Z
100
     FORMAT(' ',I1,I2,I3,F5.2,F7.3,F7.5)
     WRITE(6,200)I,J,N,X,Y,Z
200
     FORMAT(' ',15,15,15,F10.5,F10.5,F10.5)
     WRITE(6,300)I,J,N,X,Y,Z
     FORMAT(' ',315,F8.2,2F10.5)
300
     WRITE(6,400)I,J,N,X,Y,Z
400
     FORMAT(' ',15,5X,15,5X,15,5X,F5.2,5X,F7.3,5X,F7.5)
     END
```

Rezultat je:

Primer 7.2: Izpis niza znakovnih podatkov.

```
CHARACTER IMEFANTA*5, IMEDEKLETA*3
IMEFANTA='PETER'
IMEDEKLETA='ANA'
WRITE(6,100)IMEFANTA, IMEDEKLETA
100 FORMAT('',A,5X,A)
END
```

Rezultat je:

Problemi:

7.1. Ugotovite napake (če so prisotne) v naslednjih stavkih READ in FORMAT:

a)	11	READ (5,11) A,K,M,Z, FORMAT(F8.0,I15,I10,I15)
b)	11	READ (5,11)A,B,J,K,L FORMAT(3F8.1,2I8)

7.2. Ugotovite napake pri naslednjih stavkih WRITE in FORMAT:

a)	31	WRITE(6,31),A,B,N FORMAT(F10.2,3X,18,5X,16)
b)	32	WRITE (6,32)J,Z,K FORMAT(1X,I10,4X,2F10.3)

7.3. Predpostavimo, da je K=12345 in stavek WRITE, ki se izvrši, je:

WRITE (6,41) K

Kako bo izpisan rezultat v primeru:

- a) 41 FORMAT(I10)
- b) 41 FORMAT(4X, 18)
- c) 41 FORMAT(4X, I3)
- d) 41 FORMAT(15)

7.4. Predpostavimo, da je A=135.2837 in stavek WRITE, ki se izvrši, je:

WRITE (6, 42) A

Kako bo izpisan rezultat v primeru:

a)	42	FORMAT(6X,	F10.1)
b)	42	FORMAT(6X,	F12.2)
c)	42	FORMAT(6X,	F6.3)

- d) 42 FORMAT(6X, E15.6)
- 7.5. Predpostavimo, da je ID=112233 in PLACA=275.50. Kako bo izpisan rezultat:

a)	11	WRITE (6, 11) ID FORMAT(6X,'OSEBNA STEVILKA',2X,18)
b)	12	WRITE (6,12) PLACA FORMAT(6X,'TEDENSKA PLACA V USD',F8.2)

7.6. Napišite naslednje realne konstante v standardni eksponentni obliki:

123					• •											
12.3		•	•		• •		•	•	•	•	•	•	•	•	•	
5.63 *10 ⁻⁸					• •											
0.000347		•	•		• •		•	•	•	•	•	•	•	•	•	
- 0.356																
3 *10 ¹³					• •				•		•	•			•	

Standardna eksponentna oblika je: število med 0.1 in 1.0 oziroma -0.1 in -1.0 pomnožimo z 10^{n} .

7.7. Napišite računalniški program, v katerem za vsako vrstico števil napišite stavek READ in odgovarjajoči stavek FORMAT. Upoštevajte praznine. Prvi dve vrstici nad števili vam predstavljata številke kolon oziroma stolpcev in sta podani samo zaradi lažjega zapisa stavkov FORMAT in ju ne vpisujete v datoteko. Rezultate izpišite s stavkom WRITE in primernim stavkom FORMAT.

Program formirajte tako:

- berite in izpisujte podatke in rezultate v določenem formatu preko terminala,
- berite oziroma izpisujte podatke in rezultate v določenem formatu iz datoteke oziroma na datoteko.

Podatki so naslednji:

									1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	Številke
1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	kolon!
2	3	-	5	7			0		5	6	2					1	1					2	2	Vhodni
	-		-	-			· ·		9	•												~	~	VIIOUIII

Se še spomnite namiga: Eksperimentirajte s fortranom, kajti najboljša pot, da se ga naučite je, da ga uporabljate!

Vprašanja za preverjanje znanja:

- 1. S katerimi stavki je povezan stavek FORMAT?
- 2. Zakaj je pri izpisu pomembno uporabiti krmilni znak ''?
- 3. Katera formatna določila poznate in kdaj jih uporabljamo?
- 4. Katera formatna določila uporabljamo za zapis števil?
- 5. Kaj pomeni zapis 2F10.5?
- 6. Katero formatno določilo lahko uporabite v formatnem stavku namesto krmilnega znaka ' '?
- 7. Katero formatno določilo uporabljamo za zapis znakovnih podatkov?

8. POGOJNI STAVKI

• izražajo nek pogoj za vrednosti, ki jih izračunavamo in shranjujemo v pomnilniku.

8.1 Enostavni ali logični stavek IF

Fortranski stavek IF se imenuje pogojni stavek in omogoča ali onemogoča izvajanje dela programa glede na pogoj, ki je specificiran s TRUE oziroma FALSE opcijo.

<u>Splošna oblika:</u> IF (pogoj) izvršljivi stavek

Če je pogoj izpolnjen (TRUE), se izvede izvršljivi stavek, če pogoj ni izpolnjen (FALSE), se izvršljivi stavek ne izvrši. Pogoj lahko predstavlja dva aritmetična izraza (ali vrednosti), ki ju povezuje relacijski ali primerjalni operator.

Relaci	jski ali	o	Pomen	Matematični
primer	jaini op	erator		zapis
.LT.	ali	<	manjše (Less Than)	<
.GT.	ali	>	večje (Greater Than)	>
.LE.	ali	<=	manjše ali enako (Less than or Equal to)	\leq
.GE.	ali	>=	večje ali enako (Greater than or Equal to)	≥
.EQ.	ali	==	enako (EQ ual to)	=
.NE.	ali	/=	neenako (Not Equal)	≠

Opomba: Piki na obeh straneh tekstovnega operatorja sta obvezni!!

Pogoj lahko predstavlja logični izraz ali še en pogoj. Torej je lahko sestavljen iz večih pogojev. V takih primerih uporabljamo naslednje operatorje:

Npr.: .AND. - oba pogoja morata biti izpolnjena .OR. - en pogoj mora biti izpolnjen .NOT. - izpolnjeno je nasprotje podanega pogoja.

Ker primerjamo dva logična izraza, jih imenujemo logični operatorji.

Opomba: Realne vrednosti lahko primerjamo samo z realnimi vrednostmi in cele s celimi.

Primer 8.1: Uporaba različnih primerjalnih operatorjev.

```
REAL INCOME
INCOME=350.00
EXPENSE=310.00
FUNMONEY=INCOME-EXPENSE
WISHMONEY=100.00
IF(INCOME.GT.EXPENSE)WRITE(6,*)'MONEY LEFT OVER'
IF(INCOME.LT.EXPENSE)WRITE(6,*)'TROUBLE'
IF(WISHMONEY.LE.FUNMONEY)WRITE(6,*)'GREAT !!'
IF(WISHMONEY.GE.FUNMONEY)WRITE(6,*)'WISH FOR MORE'
IF((INCOME-EXPENSE).LT.(0.2*INCOME))WRITE(6,*)'SHOULD CUT EXPENSES'
END
```

Rezultat je:

Primer 8.2: Uporaba različnih logičnih operatorjev.

```
NUMBER=16
IF(NUMBER.GT.10.AND.NUMBER.LT.15)WRITE(6,100)
100 FORMAT(' ','.AND. IS TRUE')
IF(NUMBER.LT.100.OR.NUMBER.GT.1000)WRITE(6,200)
200 FORMAT(' ','.OR. IS TRUE')
X=7.8
IF(.NOT.X.EQ.5.0)WRITE(6,300)
300 FORMAT(' ','.NOT. IS TRUE')
END
```

Rezultat je:....

8.2 Blokovna IF struktura

Splošna oblika:

IF (pogoj) THEN zaporedje stavkov, ki so izvedeni, če je pogoj izpolnjen (TRUE) END IF

Primer 8.3: Uporaba blokovnega stavka IF.

Z datoteke preberemo ocene nekega izpita. Nato izpišemo določen tekst za študente, ki so dosegli pozitivno oceno.

87. 55. 75. 60. 45. - vhodna datoteka OCENE.DAT

```
OPEN(20,FILE='OCENE.DAT',STATUS='OLD')
DO I=1,5
    READ(20,*)OCENA
    IF (OCENA.GE.60.)THEN
        WRITE(6,*)'OPRAVIL'
        WRITE(6,*)'DOBER STUDENT'
        WRITE(6,*)'------'
    END IF
END IF
END DO
END
```

Rezultat je:

8.3. Struktura IF THEN ELSE

<u>Splošna oblika</u>: IF (pogoj) THEN zaporedje stavkov, ki se izvršijo, če je pogoj izpolnjen (TRUE) ELSE zaporedje stavkov, ki se izvršijo, če pogoj ni izpolnjen (FALSE) END IF

Strukturo sestavljata dva bloka: **IF blok**, ki se izvrši, če je pogoj izpolnjen in **ELSE blok**, ki se izvrši, če pogoj ni izpolnjen.

Primer 8.4: Uporaba strukture IF THEN ELSE.

87.	
55.	
75.	
60.	
45.	
	- vhodna datoteka OCENE1.DAT

Rezultat je:

OPEN(20, FILE='OCENE1.DAT',STATUS='OLD')
DO I=1,5
 READ(20,*)OCENA
 IF (OCENA.GE.60.)THEN
 WRITE(6,*)'OPRAVIL'
 WRITE(6,*)'DOBER STUDENT'
 WRITE(6,*)'------'
 ELSE
 WRITE(6,*)'NI OPRAVIL'
 WRITE(6,*)'-----'
 END IF
 END DO
 END

Prebrali smo ocene izpita z datoteke OCENE1.DAT. Nato smo izpisali tekst za študente, ki so dosegli pozitivno oceno (IF blok) in tudi za tiste, ki so dosegli negativno oceno (ELSE blok).

Problemi:

8.1. Kaj pomenijo naslednji primerjalni operatorji? EQ LE GE NE LT GT 8.2. K naslednjim trditvam vpišite primerni operator: Izpolnjeno mora biti nasprotje podanega pogoja. En pogoj mora biti izpolnjen. Oba pogoja morata biti izpolnjena. 8.3. Za vsak naslednji pogoj napišite stavek IF: če je N=0, je X=10; če je A>B, izpiši A; če je X>Y in A<B, je C=Z; če je X>0, je Y= \sqrt{X} . 8.4. Kaj je rezultat v naslednjih programih? X=25.2 Rezultat je: Y=16.3 IF(X.EQ.Y)WRITE(6,*)'Enak' IF(X.GT.Y)WRITE(6,*)'X je vecji' IF(X.LT.Y)WRITE(6,*)'Y je vecji' IF((Y+10.2).GT.X)WRITE(6,*)'Dodatek' END

N=12 M=10 IF((N.GT.M).AND.((M+5).LT.20))WRITE(6,*) 'AND je pravilen' IF((M.GT.N).OR.(N.GT.11))WRITE(6,*) 'OR je pravilen' END

Rezultat je:....

DO I=1,10 IF(I.GT.7) WRITE(6,*)I END DO END Rezultat je:

8.5. V nekem malem podjetju so zaposleni trije delavci. Za vsakega delavca imamo naslednje podatke: ime in priimek, število izvedenih ur in vrednost ure. Če delavec dela 40 ur ali manj, se mu plača izračuna enostavno tako, da število ur množimo z vrednostjo ure, nadure pa ovrednotimo kot 150 % vrednosti normalne ure. Napišite program, v katerem izračunajte in izpišite: ime in priimek, vrednost osnovne plače in vrednost nadur za vsakega delavca. Podatki so naslednji:

Ime in priimek	Število ur	Vrednost ure
JANEZ SREDA	46,5	11,2
EDVIN MARTIN	38,0	4,75
IVAN VIRT	41,4	7,5

Vprašanja za preverjanje znanja:

- 1. Kaj je značilno za pogojne stavke?
- 2. Katere primerjalne in logične operatorje poznate in kako jih uporabljamo?
- 3. Kaj pomenita opciji TRUE in FALSE?
- 4. Kako definiramo in kako deluje enostavni ali logični stavek IF ?
- 5. Kako deluje blokovna IF struktura?
- 6. Kako deluje struktura IF THEN ELSE ?
- 7. Kateri grafični simbol uporabljamo v diagramu poteka za odločitve?

9. ENODIMENZIONALNA POLJA

Polje predstavlja zaporedje vrednosti istega tipa, ki jim damo eno ime. Posamezne elemente polja dosegamo z indeksi. Enodimenzionalno polje je podatkovna struktura, kjer spreminjamo en indeks³.

Definiranje polja s stavki INTEGER, REAL in CHARACTER. Splošna oblika: INTEGER IMEPOLJA (n) REAL IMEPOLJA (n) CHARACTER IMEPOLJA (n)*število_mest_za_zapis_teksta kjer je: IMEPOLJA ime enodimenzionalnega polja število lokacij v polju

9.1 Uporaba elementov polja v zanki DO

Splošna oblika:

DO števec = 1, število_elementov_v_polju

kjer predstavlja števec indeks polja oziroma lokacijo.

9.2 Vhod (READ) in izhod (WRITE) z implicitno zanko DO



READ (le, format) (imepolja (kštevec), kštevec = kzačetna, končna)

WRITE (le, format) (imepolja (kštevec), kštevec = kzačetna, končna)

<u>Pomnite</u>: če so vhodni in izhodni stavki v kombinaciji s formatnimi stavki, le-ti krmilijo način zapisa števil.

³Žumer V., Fortran, TF Maribor, 1994.

Na enostavnem primeru poglejmo nujnost uporabe polja.

Primer 9.1: Uporaba enodimenzionalnega polja.

```
Na mizi stojijo tri posode s tekočino. V prvi posodi je 2 L vode (\rho = 1 \text{ kg/L}), v drugi so 3 L etanola (\rho = 0.789 \text{ kg/L}), v tretji posodi je 1 L glicerola (\rho = 1.26 \text{ kg/L}). Napišite program, ki bo računal maso in množino snovi v vsaki posodi! Pri tem uporabite enodimenzionalno polje za vnos podatkov kakor tudi izpis rezultatov.
```

```
231
 1.000 0.789 1.260
18.0
46.0
92.1
                         - vhodna datoteka TEKOCINE1.dat
character ime(3)*10
    integer v(3)
    real mmasa(3), masa(3), mnozina(3), ro(3)
    open (10, file='tekocine1.dat', status= 'old')
    open (20, file='tekocine1.rez', status= 'new')
    _____
С
С
    vnos podatkov: v=prostornina v L,ro=gostota v kg/L
С
          mmasa=molska masa v g/mol
    _____
С
    Razlicni vnosi vhodnih podatkov
С
    ime(1)='voda'
    ime(2)='etanol'
    ime(3)='glicerol'
    read (10, 100) v(1),v(2),v(3)
read (10, 200) (ro(i), i=1,3)
                                 ! vrednosti v vrstici
                                  ! vrednosti v vrstici
    read (10, 300) (mmasa(i), i=1,3)
                                  ! vrednosti v stolpcu
    write (20,*) 'Ime
                         m/kg
                                   n/mol'
    write (20,*)
    do k=1,3
      masa(k) = v(k) * ro(k)
      mnozina(k) = masa (k)/mmasa(k)*1000
      write (20,400) ime(k),masa(k), mnozina(k)
    end do
С
    _____
    izpis rezultatov: masa=kg, mnozina=mol
С
    _____
С
100
    format (3i2)
    format (3f7.3)
200
300
    format (f5.1)
400
    format (' ',A, F8.3,5x,f8.3)
    end
```

Rezultat na datoteki TEKOCINE1.REZ:

Ime	m/kg	n/mol
voda	2.000	111.111
etanol	2.367	51.457
glicerol	1.260	13.681

Problemi:

- **9.1.** Število lokacij, ki jih podamo v polju, definiramo na začetku programa s stavki
- **9.2.** Koliko lokacij je predvidenih in kakšna števila shranjujemo v te lokacije v naslednjih fortranskih stavkih:

INTEGER VALUES(25), REAL NUMBERS(50), CHARACTER FIRSTNAME(8)*15.....?

9.3. Kaj je izhod oziroma rezultat v naslednjih dveh fortranskih programih:

```
REAL X(4)
                                            Rezultat je:
 X(1) = 11.1
 X(2) = 22.2
 X(3) = 33.3
 X(4) = 44.4
 TOTAL=0
 DO I=1,4
   TOTAL=TOTAL+X(I)
 END DO
 WRITE(6,*)TOTAL
 END
INTEGER LX(4)
                                            Rezultat je:
LX(1) = 11.1
LX(2) = 22.2
LX(3) = 33.3
LX(4) = 44.4
TOTAL=0
DO I=1,4
  TOTAL=TOTAL+LX(I)
END DO
WRITE(6,*)TOTAL
END
```

9.4. Kaj je izhod oziroma rezultat naslednjega fortranskega programa:

```
INTEGER NUM(6)
DO I=1,6
NUM(I)=I*I
END DO
DO K=1,6
WRITE(6,*)K,NUM(K)
END DO
END
```

Rezultat je:

9.5. Kaj je izhod oziroma rezultat naslednjega fortranskega programa:

```
INTEGER N1(3),N2(3),N3(3)
N1(1)=50
N1(2)=60
N1(3)=70
N2(1)=70
N2(2)=80
N2(3)=90
DO I=1,3
    N3(I)=(N1(I)+N2(I))/2
END DO
DO I=1,3
    WRITE(6,*)I,N1(I),N2(I),N3(I)
END DO
END
```

Rezultat je:

9.6. Z datoteke berite naslednja števila:

55 65 60 65 75 80

Kombinacijo prvih treh števil shranite v eno polje, drugo kombinacijo treh števil pa v drugo polje. Za branje uporabite stavek READ z definiranim poljem (implicitna zanka DO). Nato izračunajte poprečno vrednost med številčnimi vrednostmi polj istega indeksa. Izpišite vsa števila in poprečne vrednosti parov števil. Prikaz rezultata naj bo naslednji:

Lokacija	Polje 1	Polje 2	Poprecje
1	55	65	60
2	65	75	70
3	60	80	70

Postopek ponovite z naslednjo kombinacijo števil:

55.5 65.5 60.5 65.5 75.5 80.5

Še veste: Eksperimentirajte s fortranom, kajti najboljša pot, da se ga naučite je, da ga uporabljate!

Vprašanja za preverjanje znanja:

- 1. Kaj je značilno za enodimenzionalna polja?
- 2. S katerimi stavki definiramo polje?
- 3. Kako definiramo velikost polja?
- 4. Kje v programu napišemo stavke REAL, INTEGER in CHARACTER?
- 5. Kaj je indeksirana spremenljivka?
- 6. Ali lahko v stavkih INTEGER, REAL in CHARACTER definiramo več lokacij kot vnesemo podatkov?

10. PODPROGRAMI

10.1 Podprogram FUNCTION (funkcija)

Podprogram predstavlja zaporedje stavkov, ki so napisani ločeno od glavnega programa. Uporaba funkcije je primerna, kadar podprogram vrne en sam rezultat. Zaporedje stavkov se prične z besedo FUNCTION in konča z besedo END v zadnji vrstici. Ime funkcije je lahko poljubno, vendar naj ne presega 31 znakov. Imena funkcij, ki se pričnejo s črkami I, J, K, L, M in N veljajo za cela števila, ostala za realna.

Splošna oblika:

Klic funkcije v programu:

ime = ime_funkcije (seznam argumentov)

Splošna oblika funkcijskega podprograma:

FUNCTION ime_funkcije (seznam parametrov) zaporedje izvršljivih stavkov ime_funkcije = končni_rezultat END

Primer 10.1: Uporaba podprograma FUNCTION.

Napišite fortranski program, ki bo računal in izpisal parni tlak (v bar) pri temperaturi $t = 30^{\circ}$ C za navedene komponente:

Komponenta	Α	В	С
etanol	18,5242	3578,91	- 50,50
voda	18,3036	3816,44	- 46,13
metanol	18,5875	3626,55	- 34,29

Parni tlak izračunajte po Antoinovi enačbi:

 $\ln p^{\rm nas} = A - \frac{B}{C+T} \ , \label{eq:phase}$

kjer sta parni tlak p^{nas} v mmHg in temperatura T v K.

57

303 18.5242,3578.91,-50.5 18.3036,3816.44,-46.13 18.5875,3626.55,-34.29

vhodna datoteka PARNITLAK.DAT

```
character spojina(3)*10
      open(10,file='parnitlak.dat',status='old')
      open(20,file='parnitlak.rez',status='new')
С
     Program za izracun parnega tlaka po Antoinu, koeficienti a,b in c
С
     veljajo za parni tlak v mmHg
С
С
     Vnos podatkov
С
С
      spojina(1)='etanol'
      spojina(2)='voda'
      spojina(3)='metanol'
С
     Temperatura T=K, 750=pretvornik med mmHg in bar
     read (10,*)T
     write (20,*)'ime
                            p/bar
                                       T/K'
     write (20,*)
     do i=1,3
       read(10,*)a,b,c
       p=ptlak(a,b,c,T)/750.
                                        ! klic funkcije ptlak, p=bar
       write (20,100) spojina(i), p, T
      end do
100
     format(' ',a,f7.3,2x,f6.1)
      end
      function ptlak (a,b,c,T)
      ptlak = exp(a-b/(c+T))
                                          ! izracun parnega tlaka v mmHg
      end
```

Rezultati na datoteki PARNITLAK.REZ:

ime	p/bar	T/K
etanol	0.103	303.0
voda	0.042	303.0
metanol	0.217	303.0

10.1.1 Uporaba podprograma FUNCTION in prenos podatkov

- imena argumentov so lahko v klicnem stavku enaka ali različna kot imena parametrov v podprogramu,
- število argumentov v klicnem stavku mora biti enako številu parametrov v funkcijskem podprogramu,
- ujemati se morajo argumenti in parametri, ki predstavljajo cela oziroma realna števila,
- če eden od argumentov oziroma parametrov predstavlja polje, potem mora biti polje dimenzionirano v obeh primerih, tj. v klicnem programu in funkcijskem podprogramu,
- če je klic funkcijskega podprograma v zanki DO v klicnem programu in je argument element polja, potem je v funkcijskem podprogramu parameter navadna, nedimenzionirana spremenljivka.

10.2 Podprogram SUBROUTINE

Računalniški program naj bi bil napisan kot zaporedje podprogramov, v katerih so definirani različni problemi. Struktura s podprogrami SUBROUTINE je najpogosteje uporabljena podprogramska struktura. SUBROUTINE je podprogram, ki je napisan kot posebna enota in je dosegljiv s klicnim stavkom CALL. Podatki se lahko prenašajo iz glavnega programa (ali drugih podprogramov) v podprogram in nasprotno.



Slika 10.1: V podprogramih SUBROUTINE so definirani različni problemi¹.

<u>Splošna oblika:</u>	
<u>Klic podprograma</u> :	CALL ime_podprograma (seznam argumentov)
<u>Splošna oblika podprograma</u> :	SUBROUTINE ime_podprograma (seznam parametrov) zaporedje stavkov za izračun nekega problema END

¹Middlebrooks C., VAX Fortran, Reston Publishing Company, Inc., Reston, 1984.

Podatki in rezultati se prenašajo preko argumentov in parametrov. Po dosegu stavka END se vsaka vrednost prenese v klicni stavek. Vrednosti so lahko iste ali spremenjene glede na začetne vrednosti. Ime podprograma je poljubno.

Primer 10.2: Uporaba podprograma SUBROUTINE.

S podprogramom določimo poprečje treh vrednosti, ki so podane v glavnem programu, kjer izpišemo rezultat.

```
C
     Vnos podatkov
      A1=100.0
      B1=200.0
      C1=300.0
      A2=200.0
      B2=300.0
      C2=400.0
C
      Klic podprograma POPRECJE
      CALL POPRECJE(A1,B1,C1,D)
      CALL POPRECJE(A2,B2,C2,F)
С
      Izpis D IN F vrednosti, ki smo ju izracunali s podprogramom
      PRINT *,D
      PRINT *,F
      END
      SUBROUTINE POPRECJE(A,B,C,POP)
      TOT=A+B+C
      POP=TOT/3.0
      END
```

Rezultat je:....

Problemi:

Podprogram FUNCTION:

10.1. Kaj je izhod v naslednjih fortranskih programih?

L=20 M=10 LMTOT=ITOTAL(L,M) WRITE(6,*)LMTOT END

FUNCTION ITOTAL(L,M)
ITOTAL=L+M
END

```
L=20
M=10
LMDIFER=IDIFER(L,M)
WRITE(6,*)LMDIFER
END
FUNCTION IDIFER(I,J)
IDIFER=I-J
END
```

```
REAL X(10)
DO I=1,10
    X(I)=I
END DO
XTOTAL=TOTAL(X)
WRITE(6,*)XTOTAL
END
FUNCTION TOTAL(X)
REAL X(10)
TOTAL=0
DO I=1,10
    TOTAL=TOTAL + X(I)
END DO
END
```

Rezultat je:

Rezultat je:

Rezultat je:

Podprogram SUBROUTINE:

10.2. Kaj je izhod v naslednjih fortranskih programih?

```
X=10.2
Y=15.3
CALL MNOZ(X,Y,Z)
WRITE(6,*)'Z=', Z
END
SUBROUTINE MNOZ(X,Y,Z)
Z=X*Y
END
```

Rezultat je:

```
INTEGER N(3)
                                           Rezultat je:
DO I=1,3
   N(I)=I*I
END DO
CALL DIV(N,A)
WRITE(6,*)'A=',A
END
SUBROUTINE DIV(NUM, AVE)
INTEGER NUM(3)
AVE=0.0
DO I=1,3
   AVE=AVE + NUM(I)
END DO
AVE=AVE/3
END
```

10.3. V podprogramu izračunajte vsoto dveh števil: A = 15,2 in B = 12,3. Podatke vnesite v glavni program, prav tako izpišite rezultat v glavnem programu!

Vprašanja za preverjanje znanja:

- 1. Katere vrste podprogramov poznate?
- 2. Kako kličemo podprograme v glavnem programu oziroma podprogramih?
- 3. Kakšna je zgradba podprograma SUBROUTINE?
- 4. Kako se prenašajo argumenti oziroma parametri v podprogramu SUBROUTINE?
- 5. Kakšna je zgradba podprograma FUNCTION?
- 6. Kako kličemo v klicnem programu podprogram FUNCTION in kako se prenašajo argumenti oziroma parametri?

11. OSTALO

11.1 Stavek COMMON

V prejšnjem poglavju smo videli, kako se prenašajo podatki med klicnim programom in podprogramom preko argumentov in parametrov. Tak način prenosa je tudi najbolj pogost. Druga možnost pa je prenos podatkov preko stavka COMMON. En stavek COMMON je lahko brez imena, če jih je več, pa morajo vsebovati razpoznavno ime.

Splošna oblika:
aliCOMMON seznam spremenljivk in imena polj
COMMON /ime/ seznam spremenljivk in imena polj

Opomba: Število spremenljivk definiranih v stavku COMMON v podprogramu je lahko manjše kot število spremenljivk definiranih v klicnem programu (nikoli pa obratno).

Primer 11.1: Uporaba stavka COMMON.

Toplotno kapaciteto dušika N₂ pri konstantnem tlaku p = 1,03 bar lahko izračunamo po enačbi: $C_p = 0,029 + 0,2199 \times 10^{-5} T + 0,5723 \times 10^{-8} T^2 - 2,871 \times 10^{-12} T^3$, kjer je $C_p \vee \text{kJ/(mol °C)}$ in $T \vee ^{\circ}$ C. Predpostavimo idealno obnašanje plina. Izračunajte C_p dušika pri temperaturi 20 °C.

```
_____
```

```
COMMON A,B,C,D,T,CP

T=20.

A=0.029

B=0.2199E-5

C=0.5723E-8

D=-2.871E-12

CALL TOPCAP

WRITE (6,100)T,CP

100 FORMAT (' ','t(C) =',F5.1,2X,'CP(N2) =',F9.6,' kJ/(MOL C)')

END

SUBROUTINE TOPCAP

COMMON A,B,C,D,T,CP

CP=A+B*T+C*T**2+D*T**3

END
```

Rezultat je: t(C) = 20.0 CP(N2) = 0.029046 kJ/(MOL C)

11.2 Stavek DATA

Lahko ga uporabljamo za branje oziroma vnos podatkov.

Splošna oblika:

DATA seznam konstant oziroma spremenljivk /vrednosti/

konstante oziroma spremenljivke ločimo z vejico, prav tako podane vrednosti ločimo z vejico.

DATA ime1 /vrednost1/, ime2 /vrednost2/,, imen /vrednostn/

- DATA stavek z implicitno zanko DO:

DATA (IMEPOLJA(števec), števec=1,n)/vrednost1, vrednost2,....vrednostn/

Poglejmo možne načine vnosa numeričnih podatkov s stavki DATA! Vnesimo numerične vrednosti 1, 5, 6, in 10 v pomnilnik.

a) <u>1. način</u>: DATA K1, K2, K3, K4/1, 5, 6, 10/

- b) <u>2. način</u>: DATA K1/1/, K2/5/, K3/6/, K4/10/
- c) <u>3. način</u>: V tem primeru shranjujemo podatke v polje ISTEVILA in pri vnosu uporabimo implicitno zanko DO:

DATA (ISTEVILA(I), I=1,4)/1, 5, 6, 10/

Seveda lahko s stavki DATA vnašate na enak način tudi realna števila!

Sedaj se spomnimo še drugih načinov vnosa podatkov, ki smo jih že obravnavali. Prikažite vnos zgornjih števil z naslednjimi načini:

Aritmetični vnos:

Vnos s stavkom READ preko terminala – interaktivni vnos:

Vnos s stavkom READ preko datoteke:

Primer 11.2 prikazuje različne načine vnosa znakovnih podatkov.
Primer 11.2: Vnos znakovnih podatkov s stavkom DATA.

```
CHARACTER PLIN(5)*7, KOVINA(4)*7
*
     Uporaba stavka DATA pri vnosu znakovnih podatkov
     DATA (PLIN(I), I=1,5)/'klor','butan','vodik','helij','argon'/
     OPEN (10, FILE='PBTEST29.REZ', STATUS='NEW')
!
     Izpis z zanko DO
     WRITE(10,*)'********* Izpis plinov **********
     DO I=1,5
        WRITE(10,*)PLIN(I)
     END DO
С
     Vnos znakovnih podatkov v polje z definiranjem indeksirane
     spremenljivke
С
     KOVINA(1)='baker'
     KOVINA(2)='svinec'
     KOVINA(3)='srebro'
     KOVINA(4)='zlato'
!
     izpis z zanko DO
     WRITE(10,*)'********* Izpis kovin ***********
     DO N=1,4
        WRITE(10,*)KOVINA(N)
     END DO
     END
```

Rezultat na datoteki PBTEST29.REZ:

```
********* Izpis plinov ********
klor
butan
vodik
helij
argon
********** Izpis kovin *********
baker
svinec
srebro
zlato
```

Problemi:

11.1. Kaj je izhod iz naslednjega programa?

```
COMMON D, E, F
D=2.3
E=1.2
CALL TOTAL
WRITE(6,*)'F=',F
END
SUBROUTINE TOTAL
COMMON P, Q, R
R=P + Q
END
```

Rezultat je:

11.2. Določite vhodne podatke, tj. vrednosti za A, B in C, če je:

DATA A, B /2*2.4/, C /1.5/

11.3. Z DATA stavkom, ki vsebuje implicitno zanko DO, vnesite v polje XX naslednja števila: 1, 12, 14, 7, 16, 0. Nato števila izpišite v obliki preglednice:

Lokacija	Vrednost
1	1
2	12
3	14
4	7
5	16
6	0

Vprašanja za preverjanje znanja:

- 1. Kdaj uporabljamo stavek COMMON?
- **2.** Kdaj uporabljamo stavek DATA?

2. DEL

MS EXCEL 2007 Kratka navodila za uporabo

12.OSNOVE DELA V PROGRAMU EXCEL 2007

Excel je program, v katerem lahko ustvarjamo tabele ter izračunavamo in analiziramo podatke. Taki programi se imenujejo programi za ustvarjanje preglednic.V Excelu je mogoče ustvariti tabele, ki samodejno izračunajo vsote vnesenih številskih vrednosti, natisniti urejene tabele in ustvariti preproste grafe. V Excelovo datoteko lahko vnesemo kup urejenih in neurejenih podatkov, ki jih z njegovo pomočjo organiziramo, analiziramo in na pregleden način grafično predstavimo. Z zbirko orodij, ki nam jo ponuja, lahko izvajamo tudi številne poslovne analize. Program vsebuje številne že izdelane delovne zvezke, ki jih lahko uporabimo pri delu. Hkrati lahko v programu izdelujemo in oblikujemo tudi nove delovne zvezke popolnoma po svojih željah.

Program Excel 2007 pogosto omogoča več različnih načinov za doseganje istega končnega rezultata. Uporabna opravila, ki jih uporabljajo študentje tekom izobraževanja, so predstavljena po korakih in z nazornimi ilustracijami. Veliko je predstavljenih primerov, skozi katere testiramo usvojeno znanje. Nekateri ukazi in zaslonske slike se morda ne bodo ujemali s tem, kar bo na zaslonu, saj imamo opravka z osebnim računalnikom, ki ga lahko nastavimo po lastni želji. V nekaterih primerih je uporabljena slovenska v nekaterih primerih pa angleška različica MS Excela 2007. To naj uporabnika navodil ne moti, saj je MS Excel program, ki ga v svetu množično uporabljajo in boste na ta način usvojili terminologijo angleške verzije in tako širili svoje znanje.

12.1 Zagon programa Microsoft Excel

Zagon Excela s praznim delovnim zvezkom (slika 12.1):

- 1. V opravilni vrstici kliknemo gumb Start.
- 2. Pokažemo na Vsi programi.
- 3. Kliknemo Microsoft Office.
- 4. Kliknemo Microsoft Office Excel 2007 in odpre se delovno okno (slika 12.2).

	S	📄 🕅 Orodia zbirke Microsoft Office 🔹 🕨	,
	m Microsoft Office		
	🛅 Total Commander	Microsoft Office Access 2007	
	astraSlim	🔀 Microsoft Office Excel 2007	
	📩 ScanManager	🖡 🔞 Microsoft Office Groove 2007	
	Adobe	🖡 🛃 Microsoft Office InfoPath 2007	
	m Ghostgum	🖡 🜆 Microsoft Office OneNote 2007	
	Ghostscript	🖌 😡 Microsoft Office Outlook 2007	
	m eMule	🖡 🔞 Microsoft Office PowerPoint 2007	
Vsi programi 👂	GrindEQ Math Utilities	🖡 🗊 Microsoft Office Publisher 2007	
	MathType 6	Microsoft Office Word 2007	
	Microcal Origin	Microsoft Office Tools	
🚪 start 👘 🙆 ն	m ArcSoft Connect	Microsoft Office Visio 2007	

Slika 12.1: Zagon Excela



Slika 12.2: Zagonski zaslon.

Odpiranje že izdelanega delovnega zvezka:

- 1. V opravilni vrstici kliknemo gumb Start.
- 2. Kliknemo Moj Računalnik.
- 3. Premaknemo se v mapo, ki vsebuje želen dokument.
- 4. Dvokliknemo datoteko, ki jo želimo odpreti.

12.2 Zapiranje programa Microsoft Excel

Zapiranje delovnega zvezka (slika 12.3):

- 1. V zgornjem desnem kotu Excelovega okna kliknemo gumb Zapri.
- 2. Če se nam odpre pogovorno okno z vprašanjem, ali želimo shraniti spremembe, ravnamo tako:
- Kliknemo **Da**, da shranimo delovni zvezek pod istim imenom in na isto mesto.
- Kliknemo Ne, da zavržemo vse spremembe, do katerih je prišlo, odkar smo delovni zvezek zadnjič shranili.
- Kliknemo Prekliči, da se vrnemo v delovni zvezek.

Osnown Vstavljanje Postavitev strani Formule Podatki Prejed Ogled Image: Construction of the strand of t	0.		7 - (4 -)	Ŧ		Zv	ezek1 - Micr	osoft Exce	el				- = X
A B C D E F G H I J K Lebrisi - Lebrisi	9	Osn	ovno Vst	tavljanje	Postavitev st	trani Fo	rmule Poo	datki l	Pregled	Ogled		0) _ = ×
A1 F G H I J K L 1 1 1 1 1 K L 1 K L 1 K L 1 K L 1 K L 1 K L 1 K L 1 K L 1 K L 1 L	Prilepi Odloži	ی انتان ا	Calibri K Z P Control Control	• 11 • • A * • A • •	■ ■ ■ ■ ■ ■ 律 律 ≹ ≹	■ ■ • • • • • • • • • • • • • • • • • •	Splošno * SP * % 000 S8 \$9 Število 🕞	B Poge B Obli Slog	ojno oblikov kuj kot tabel i celic * Slogi	anje + lo +	Sea Vstavi → Izbriši → Oblika → Celice	∑ - A Z - Z - Razvrst C - filtrira Urej	i in Poišči in j * izberi * anje
A B C D E F G H I J K L 1	1	A1		• ()	fx								*
1 2 3 3 4 4 5 4 6 4 7 4 8 4 9 4 10 4 11 4 12 4 13 4 14 4 15 4 16 4 17 4 18 4 19 4		А	В	С	D	E	F	G	Н	I	J	К	L
2 3 4 5 6 7 7 8 9 10 11 12 13 14 15 16 17 17 17 17 17 17 17 17 17 17	1		_!										
3 4 4 5 5 6 7 7 8 9 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 1 19 1 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 1 19 1 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 1 19 1 19 1 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18	2												
4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 10 11 12 13 14 15 16 17 17 18 19 11 12 13 14 15 16 17 17 17 18 19 11 12 13 14 15 16 17 17 18 19 11 10 11 12 13 14 15 16 17 17 18 19 11 11 11 12 13 14 </td <td>3</td> <td></td>	3												
3 6 7 7 8 9 9 10 11 10 11 12 13 14 15 15 16 17 16 17 14 15 16 17 16 17 14 17 15 16 16 17 14	4												
7 8 9 9 10 10 11 12 13 14 15 16 16 17 17 17 17 17 17 17 17 17 17 17 17 17	6												
8 9 9 10 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 1 17 1 18 1 19 1 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 1 19 1 10 1 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 1 19 1 10 1 10 1 11 1 12 1 13 1 14 1 15 1 16 1 17 1 18 <td>7</td> <td></td>	7												
9 10 11 11 12 13 14 15 16 17 17 17 17 18 14 15 16 17 17 17 17 17 17 17 17 17 17 17 17 17	8												
10 11 12 13 14 15 16 17 17 18 19 10 10 10 10 10 10 10 10 10 10	9												
11 12 13 14 15 16 17 17 18 19 10 10 10 10 10 10 10 10 10 10	10												
12 13 14 15 16 17 17 18 19 10 10 10 10 10 10 10 10 10 10	11												
13 14 15 16 17 16 17 19 14 15 16 17 16 17 16 17 16 17 16 17 16 17 16 17 16 17 16 17 16 16 17 16 16 17 16 16 17 16 16 16 17 16 16 16 16 16 16 16 16 16 16	12												
14 15 16 17 17 10 List1 / List2 / List3 / 22	13												
13 16 17 17 10 List1 / List2 / List3 / 22	14												- U
	15												
	17												
	10		1140	11110 /8						1		_	×
	Priprav	lien	ISTI / LIST2							m	I III 100% (·		

Slika 12.3: Zapiranje delovnega zvezka.

12.3 Excelovo delovno okno

Excelovo delovno okno je razdeljeno v vrstice in stolpce, podatke pa urejamo tako, da jih vnesemo v to razpredelnico. Polje, ki je presek vrstice in stolpca, se imenuje aktivna celica. Aktivne celice prepoznamo po črki stolpca in številki vrstice. Na sliki 12.4 vidimo, da je naša aktivna celica A1, ki označuje prvo celico v prvem stolpcu in se imenuje sklic na celico. Slika prikazuje pomembne funkcije okna programa Excel 2007.

	1		2				1						
2	0)·("·);	Ŧ		Z	vezek1 - Micro	osoft Exc	el				_ = ×
3	0	Osno	ovno Vst	avljanje	Postavitev st	rani F	ormule Poo	datki	Pregled	Ogled) _ = x
4	Pril	epi	Calibri K L P ···································	• 11 • • A A • •	E E E		Splošno ▼	Pog B Obli Slog	ojno oblikov ikuj kot tabe gi celic * Slogi	vanje * :lo *	Stavi ▼ Vstavi ▼ Izbriši ▼ Dblika ▼ Celice	∑ - A	ti in Poišči in aj ~ izberi ~ anje
		A1		• (•	f _x			NU-					×
		А	В	С	D	E	F	G	Н	1	J	K	L
	1												
	2			-						-			
	3												
	4		-				-			-		_	
	5												
2	6												
	/										_		=
	0												
	10										_		_
	11												
	12												
	13												
	14												
	15												
	16												-
	17												9
	10	→ H Li	ist1 List2	List3	2/						JUL		* 1
	Prip	ravljen									100% (9 0	() .:
					6					7		8	

Slika 12.4: Elementi delovnega lista.

- 1 Naslovna vrstica: prikazuje ime delovnega lista, ki ga urejamo in ime programa.
- 2 Orodna vrstica: prikazuje ukaze, ki jih pogosto uporabljamo npr: Shrani in Razveljavi.
- 3 Gumb »Microsoft Office«: kliknemo gumb, ko uporabljamo osnovne ukaze, kot so Novo, Odpri, Shrani kot, Natisni in Zapri.
- 4 Trak: sestavljen iz ukazov, ki jih potrebujemo pri delu. Ima enako vlogo, kot jo imajo »meniji« in »orodne vrstice« v starejših programih programa Excel.
- 5 Okno za urejanje: v njem je predstavljen delovni list, ki ga urejamo. Delovni list sestavljajo vrstice in stolpci. Vanje vnašamo podatke ali jih urejamo. Pravokotniki delovnega lista se imenujejo »celice«.
- 6 Vrstica stanja: v njem so prikazani podatki o delovnem listu, ki ga urejamo.
- 7 Zaslonski gumbi: Z njimi ustrezno spreminjamo način, v katerem je delovni list, prikazan na zaslonu.

- 8 Drsniki za povečavo: omogočajo, da spreminjamo nastavitev povečave v delovnem listu.
- 9 Drsni trak: omogoča, da spreminjamo mesto prikaza delovnega lista.

Vprašanja za preverjanje znanja:

- 1. Kaj omogoča program Excel?
- 2. Kako je razdeljeno Excelovo delovno okno?
- 3. Kako prepoznavamo aktivne celice?

13. OSNOVNE OPERACIJE

13.1 Vnos podatkov

Podatek vnesemo tako, da izberemo celico (kliknemo nanjo z miško ali se premaknemo s smernimi tipkami do nje) in začnemo tipkati (slika 13.1). Podatek lahko vnesemo v eno celico ali v že prej označeno področje celic. Podatek se pri vnosu začne izpisovati v vnosni vrstici. Če podatek vnesemo pravilno, vnos potrdimo s tipko **Enter**.

		· (* ·) ;	•		Z	vezek1 - N	licroso	oft Excel				- =	x
9	Osnovr	no Vsta	avljanje	Postavitev str	ani Fo	ormule	Podatk	i Pregled	Ogled			0 - 7	x
Prilepi Odložiš	če G	alibri K Z <u>P</u> Disava	* 11 * * A A A * 5	F F F F		Splošno Splošno Splošno Splošno Stevilo	*	Slogino oblik Oblikuj kot ta Slogi celic * Slogi	ovanje * belo *	G*= Vstavi → S* Izbriši → Oblika → Celice	Σ · A Z · Z A Razvrs G · Giltrin Ure	ti in Poišči ir aj v izberi v janje	n
	A1	•	(• × ✓	f _x Anali	za								≯
	А	В	С	D	E	F	1	G H	I	J	К	L	
1 An	aliza												
2													
3													

Slika 13.1: Vnos podatkov v celico.

13.1.1 Vnos besedil

Besedila največkrat vnašamo zato, da z njimi razložimo pomen številskih podatkov v stolpcih ali vrsticah, npr. artikli, dnevi, leta, podjetja, kraj. Besedila so znakovni podatki ali kombinacije različnih vrst podatkov.

Z besedili ne moremo računati, lahko pa jih združujemo, razvrščamo. Besedilni podatki se v celici samodejno levo poravnajo.

Če celica za vneseni podatek ni dovolj široka, se besedilo zapiše preko sosednjih (praznih) celic. Če te niso prazne, se del besedila skrije za druge celice.

Širino stolpca lahko prilagodimo podatkom. Z miško se postavimo na mejo med naslova stolpcev (B, C) in dobimo dvosmerno puščico (slika 13.2). S potegom miške stolpec B širimo ali ožimo, z dvojnim klikom pa njegovo širino samodejno prilagodimo najširšemu zapisu v tem stolpcu.

	B2	- (• fx	Vanja	
	А	B -	- с	D	
1		Ime	Priimek		
2		Vanja	Forjan		
3		Ana Maria	Kostajnšek		
4		Brabara	Lukanovič		
-					

Slika 13.2: Prilagoditev širine stolpca besedilu.

13.1.2 Vnos števil

Števila so podatki, sestavljeni iz števk in včasih še iz nekaterih drugih znakov: 0 1 2 3 4 5 6 7 8 9 + - (), / SIT % in tako naprej. S števili izvajamo številne matematične, statistične, finančne, logične in druge operacije. Rezultati obdelav so številske ali logične vrednosti. V celici se podatki samodejno desno poravnajo.

Vnesena števila lahko prikažemo v številnih oblikah. Prikaz valute, odstotkov in sloga tisočic lahko izvedemo iz orodne vrstice. Tudi število decimalnih mest lahko z ikonama spreminjamo (slika 13.3).

Za zapis odstotkov vnesemo število v relativni obliki (npr. 0,123). Po potrditvi vnosa z **Enter** ga označimo in spremenimo s pomočjo gumba v oblikovni vrstici v 12,3 % (slika 13.3).

Dolga števila lahko vnašamo tudi v eksponentni obliki.(npr. 1,2e6). Če je stolpec preozek, da bi v celici videli celo število, Excel najprej odreže decimalna mesta, potem prikaže število v eksponentni obliki, če še tega ne more, dobimo v celici izpisane znake ###. Prikaz za izračun ni pomemben, saj Excel vedno računa s pravim – vnesenim številom.

2	splo	sr	10	+		
	3	Ŧ	%	000		
	◆,0 ,00 ,00 →,0					
	Št	ev	ilo	- Fai		

Slika 13.3: Možen prikaz vrednosti v celici kot odstotek, valuta, ...

13.1.3 Vnos datuma in časa

Datum in čas sta v bistvu tudi številska podatka, saj se vsak datum zapiše kot zaporedna številka dneva od 1. 1. 1900 (=1) dalje. Datum lahko vnesemo v različnih zapisih. Najenostavnejši je vnos v obliki DD/MM/(LL)LL. Kot ločilo lahko uporabimo poševnico ali piko. Vneseni datum Excel zapiše v skladu z nastavitvami.

13.2 Premikanje po listih

Med podatki se lahko premikamo z miško ali s tipkovnico.

Premiki z miško:

- Po posameznih listih delovnega zvezka se sprehajamo z miško.
- S pomočjo vodoravnega ali navpičnega drsnika na zaslonu se premikamo levo-desno ali navzgor-navzdol po delovnem listu.
- Z neposrednim klikom na želeni celici pregledamo vsebino.

Pri vnašanju podatkov pa je premikanje s tipkovnico enostavnejše in hitrejše, saj pritisk na tipke pomeni:

smerne tipke GOR/DOL/ LEVO/DESNO	premik za eno polje v smeri tipke
 tipka CTRL + smerna tipka GOR/DOL tipka CTRL + smerna tipka LEVO/DESNO 	premik na zgornji/spodnji rob bloka podatkov v stolpcu premik na levi/spodnji rob bloka podatkov v vrstici
➢ tipka HOME	pomik na začetek vrstice
tipka CTRL + tipka HOME	pomik na začetek delovnega lista (v celico A1)
tipka PAGE UP	pomik za en zaslon navzgor
tipka PAGE DOWN	pomik za en zaslon navzdol

13.3 Premikanje in kopiranje podatkov

Premikanje ali kopiranje podatkov lahko izvajamo samo s pomočjo označevanja (slika 13.4).

Področje celic lahko označimo na več načinov:

- Eno celico označimo tako, da na njej kliknemo z levim miškinim gumbom ali se do nje prebijemo s smernimi tipkami.
- Če kliknemo na ime stolpca, se obarva celoten stolpec celic. Na enak način označimo tudi celo vrstico.
- Postavimo se v celico A1, držimo levo tipko na miški in povlečemo do A7. Označili smo področje celic A1:A7, kot vidimo na sliki 13.4.
- Celoten delovni list označimo s klikom v presečišču imen stolpcev in vrstic (siv pravokotnik v levem zgornjem kotu).



Slika 13.4: Označitev celic.

Posamezno celico ali področje celic lahko premaknemo na več načinov:

- Postavimo se na rob celice ali označenega področja, držimo levi gumb in ga spustimo šele na želenem mestu.
- Najprej izberemo celico oziroma področje, nato izberemo gumb Izreži, postavimo se na želeno mesto in kliknemo gumb Prilepi.

Vsebina celic ali področja se izpiše na novem mestu.

Kopiranje celice ali področja celic poteka na podoben način kot premikanje:

 Najprej označimo celico oziroma področje, izberemo gumb Kopiraj, postavimo se na želeno mesto in izberemo gumb Prilepi.

Na izbranem mestu se pojavi kopija prej označenih celic ali področja celic s podatki.

13.4 Spreminjanje ali brisanje podatkov

Če nam podatki v celici ne ustrezajo več, jih nadomestimo z novimi ali jih izbrišemo.

- Z enojnim klikom na celici pišemo v vnosno polje in staro vsebino celice nadomestimo z novimi podatki.
- Če na celico dvojno kliknemo, pišemo naravnost v celico.
- Vsebino označenih celic izbrišemo s pritiskom na tipko Delete.

13.5 Vstavljanje in brisanje celic, vrstic ali stolpcev

Včasih se nam pri vnašanju podatkov zgodi, da pozabimo vpisati nekaj podatkov (npr. vnesli smo tabelo s podatki in ugotovili, da smo pozabili vnesti cel stolpec podatkov).

Celice vstavimo tako, da:

- desno kliknemo na celico, izberemo Vstavi ali
- določimo, kam naj se že vpisani podatki premaknejo.

Stolpce vstavimo tako, da:

- z miško kliknemo na oznako stolpca, pred katero želimo vnesti novi stolpec
- desno kliknemo in izberemo ukaz Vstavi ali
- kliknemo na celico, desno kliknemo in izberemo ukaz Vstavi (slika 13.5).

Na enak način lahko vstavimo tudi vrstice:

- z miško kliknemo na oznako vrstice, pred katerega želimo vnesti novi vrstico
- desno kliknemo in izberemo ukaz Vstavi ali
- kliknemo na celico, desno kliknemo in izberemo ukaz Vstavi (slika 13.5).

¥	Iz <u>r</u> eži			
Ð	<u>K</u> opiraj			
2	<u>P</u> rilepi			
	Po <u>s</u> ebno lepljenje			
	O <u>s</u> veži		-	
	<u>V</u> stavi		I ^T II	Stolpci tabele na <u>l</u> evo
	<u>I</u> zbriši	•	-	Vrstice tabele <u>zg</u> oraj
	Izberi	•		Vrstica tabele <u>s</u> podaj



Celic, stolpcev in vrstic, ki jih ne potrebujemo se znebimo tako, da:

- Označimo področje celic, v meniju Urejanje izberemo ukaz Počisti. V pogovornem oknu se odločimo med možnostmi izbrisa.
- Lahko pa označimo področje celic, desno kliknemo in izberemo ukaz Izbriši (slika13.6).

*	Iz <u>r</u> eži				
	<u>K</u> opiraj				
	<u>P</u> rilepi				
	Po <u>s</u> ebno lepljenje				
	O <u>s</u> veži				
	<u>V</u> stavi	•	-		
	<u>I</u> zbriši		¥	<u>S</u> tolpci tabele	ł
-	Izberi	×	*	<u>V</u> rstice tabele	ł
-	De Zieki week in e	I		1	4

Slika 13.6: Brisanje stolpcev ali vrstic.

13.6 Operacije z delovnimi listi

13.6.1 Pregledovanje listov

Posamezen list odpremo enostavno z miškinim klikom na njegov jeziček (slika 13.7). Če raje uporabljamo tipkovnico, se sprehajamo s pomočjo kombinacije tipk **CTRL+PAGE DOWN** ali **CTRL+PAGE UP**. Po izbranem listu se NAVZGOR/NAVZDOL ali LEVO/DESNO najhitreje pomikamo z vodoravnim ali navpičnim drsnikom na zaslonu.

23			
24			
25			
H 4	List1 Lis	t2 🖉 List3 🖉 🎾	
Pripra	wljen		

Slika 13.7: Delovni listi.

13.6.2 Preimenovanje delovnih listov

Vsak list je poimenovan kot List1, List2,... (ali Sheet1, Sheet2,), kar pa nam ne pove veliko o vsebini podatkov na njem. Zaradi boljše preglednosti in orientacije spremenimo ime posameznega lista tako, da:

- desno kliknemo na jezičku lista ukaz Preimenuj (slika 13.8), ki zahteva od nas vnos novega imena lista ali
- dvakrat kliknemo na ime lista in vtipkamo njegovo novo ime.



Slika 13.8: Preimenovanje delovnih listov.

13.6.3 Vstavljanje in brisanje listov

Ko vse liste v delovnem zvezku napolnimo s podatki, lahko dodamo nove:

- desno kliknemo na jeziček lista, pred katerega želimo vstaviti nov list, ter izberemo ukaz Vstavi in
- izberemo Delovni list.

Odvečne delovne liste enostavno odstranimo tako, da:

• desno kliknemo na jeziček lista, ki ga želimo izbrisati, ter izberemo ukaz Izbriši.

Excel nas opozori, da bo list izbrisan za vedno, zato moramo brisanje posebej potrditi. Po brisanju lista namreč ne moremo uporabiti ukaza za preklic.

13.6.4 Premikanje in kopiranje listov

Vrstni red listov v Excelovem zvezku spremenimo tako, da:

- desno kliknemo na jeziček lista, ki ga želimo premakniti,
- izberemo ukaz Premakni,
- določimo, pred kateri list naj se želeni list premakne (lahko tudi premik na konec delovnega zvezka) ali
- list lahko tudi kar odvlečemo z miško in na želenem mestu spustimo.

Če želimo list kopirati, vključimo možnost **Ustvari kopijo**. Delovni list lahko premaknemo ali kopiramo tudi v nov ali drug že odprt delovni zvezek.

Problemi:

Sestava mineralne	Koncentracija
vode Donat Mg	c/(mg/L)
kationi	
Mg ²⁺	1030
Na ⁺	1500
Ca ²⁺	180
anioni	
HCO ₃ ⁻	7700
SO4 ²⁻	2400
Cl	59
CO_2	3500

13.1 V izbrani delovni list vnesite naslednje podatke:

Vprašanja za preverjanje znanja:

- 1. Kako vnašamo podatke na delovnem listu?
- 2. Kakšne podatke lahko vnašamo v celice?
- 3. Katere so najpogostejše operacije z delovnimi listi?

14. OBLIKOVANJE PODATKOV

Pomembno je, da smo pozorni na preglednost podatkov, ter da s primernim oblikovanjem poudarimo ključne podatke – informacije. To dosežemo z izbiro oblike zapisa podatkov, s poravnavo, z oblikovanjem pisave, z obrobami in s senčenjem celic.

14.1 Uporaba traku Osnovno

Preden oblikujemo podatke v celicah ali celice same, celice označimo. Najpogosteje uporabljena oblikovalna orodja so v skupinah **Pisava**, **Poravnava** in **Število** na kartici **Osnovno** (slika 14.1).

Osn	ovno Vstavljanje	Postavitev strani	Formule Pod	atki Pregled Ogled		@ _ = ×
Prilepi	Calibri \cdot 11 \cdot K L P \cdot A \cdot N \cdot A \cdot Pisava		Splošno * Image: splosing transform % 000 * % 000 * % 000 * % % Stevilo Image: splosing transform Image: splosing transform	Pogojno oblikovanje * BOblikuj kot tabelo * Slogi celic * Slogi	Gara Vstavi ▼ Marka Vstavi ▼ Izbriši ▼ Goblika ▼ Celice	∑ · Razvrsti in Poišči in ¢ · riltriraj · izberi · Ureianie

Slika 14.1: Trak sestavljen iz ukazov, ki jih potrebujemo za delo.

14.2 Uporaba okna Oblikovanje celice

Če oblikovalne možnosti, zajete na traku **Osnovno**, niso zadostne za želeno oblikovanje, v skupinah **Pisava**, **Poravnava** ali **Število** kliknemo na zaganjalnik pogovornega okna, ki nam ponuja več možnosti oblikovanja (slika 14.2):

Splo	-		
9	%	000	
◆,0 ,00 ,00 ◆,0			
Število 🗔			

Slika 14.2: Zaganjalnik pogovornega okna.

14.2.1 Številke

Obliko številskih podatkov določimo v Zvrsti. Vsako zvrst še dodatno prilagodimo. Izberemo:

- število decimalnih mest,
- obliko zapisa negativnih vrednosti,
- valutni simbol npr. EUR oziroma €,
- obliko zapisa datuma in časa,
- obliko zapisa ulomka ali
- kreiramo obliko po meri.

Za izničenje vseh nastavitev izberemo Splošno (General) (slika 14.3).

Oblikovanje celic						? 🗙
Oblikovanje celic Stevilke Poravnava Zvrst: Splošno Stevilka Valuta Računovodsko Datum Čas Odstotek Ulomek Znanstveno Besedilo	Pisava Vzorec Splošna	Obroba oblika celic	Polnilo nima določe	Zaščita ne oblike :	za števila.	? ×
Posebno Po meri					V redu Pi	rekliči

Slika 14.3: Oblikovanje celic – Številke.

14.2.2 Poravnava

Pri poravnavi podatkov v celicah lahko izberemo (slika 14.4):

- poravnavo v vodoravni smeri (levo, sredinsko...),
- poravnavo v navpični smeri (na vrh, na sredino ...),
- usmerjenost izpisa pod poljubnim kotom.

Prelomi besedilo pomeni, da besedilo prilagodimo širini celice, če je celica preozka. Besedilo se izpiše v več vrsticah.

Oblikovanje celic	? 🛛
Številke Poravnava Pisava Obroba Polnilo Zaščita	
Poravnava besedila	Usmerjenost
Vodoravna:	• •
Splošno 🛛 Zamik:	в •
Navpična: 0	e ·
Na dno 🖌	s ·
Obojestransko porazdeljeno	d Besedilo — •
Nadzor besedila	- i
Prelomi besedilo	•
🔲 Skrči, da ustreza	• ·
Spoji <u>c</u> elice	0 😂 stopinj
Od desne proti levi	
S <u>m</u> er besedila:	
Kontekst	
L	V redu Prekliči

Slika 14.4: Oblikovanje celic – Poravnava.

14.2.3 Pisava

Na zavihku Pisava lahko izbiramo (slika 14.5):

- vrsto pisave,
- slog pisave (navadno, ležeče, krepko ...),
- velikost znakov,
- podčrtavanje (enojno, dvojno ...),
- barvo znakov,
- učinke (prečrtano, nadpisano, podpisano).

Za izničenje vseh nastavitev izberemo Navadna pisava.

Oblikovanje celic	? 🗙
Številke Poravnava Pisava Obroba	Polnilo Zaščita
Pisava:	Slog pisave: Velikost:
Calibri	Običajno 11
T Cambria (Naslovi) T Calibri (Telo) Adobe Caslon Pro Adobe Caslon Pro Bold Adobe Fangsong Std R Adobe Garamond Pro	Običajno 8 Ležeče 9 Krepko 10 Krepko ležeče 11 ✓ 12 ✓ 14
Pod <u>č</u> rtavanje:	Barva:
Brez	👻 💌 Na <u>v</u> adna pisava
Uänki	Predogled
Prečr <u>t</u> ano <u>N</u> adpisano Po <u>d</u> pisano	AaČčŠšŽž
To je TrueType pisava. Za zaslon in tiskalnik se	e bo uporabljala ista pisava.
	V redu Prekliči

Slika 14.5: Oblikovanje celic – Pisava.

14.2.4 Obroba

Najprej izberemo slog črte in barvo obrobe. S kliki na gumbe določimo pozicijo obrob za izbrane celice. Če želimo različne obrobe, postopek ponovimo za vsak slog in barvo obrob. Za izničenje vseh obrob kliknemo gumb **Brez**.

14.2.5 Polnilo

Vzorec senčenja določimo tako, da izberemo barvo v razpoložljivi paleti. Za izničenje senčenja kliknemo na **Brez barve**.

14.3 Urejanje podatkov v tabele

Podatke urejamo v Excelove tabele, sestavljene iz stolpcev in vrstic, ki vsebujejo podatke o določeni zadevi. Tabele v programu Excel 2007 olajšajo vnašanje in povzemanje podatkov. Na naslednjem primeru bomo spoznali enostavno in hitro ustvarjanje tabele.

Primer 14.1: Ustvarjanje tabele.

1. Izberemo obseg celic (slika 14.6), kjer želimo ustvariti tabelo. Kliknemo **Vstavljanje** in **Tabela**.



Slika 14.6: Izbor celic.

2. Prikaže se pogovorno okno Ustvari tabelo (slika 14.7). Kliknemo V redu.

Ustvari tabelo 🛛 🛛 🔀
Kje so podatki za vašo tabelo?
<u>=\$B\$2:\$D\$6</u>
🔄 Moja tabela ima glave
V redu Prekliči

Slika 14.7: Vnos podatkov za tabelo.

3. Izbrane celice se pretvorijo v tabelo (slika 14.8).



Slika 14.8: Ustvarjena tabela s samodejno dodano naslovno vrstico.

Ko ustvarimo tabelo, program v prvo vrstico izbranega obsega celic samodejno vnese »Stolpec1«, »Stolpec2« in »Stolpec3«. To se imenuje naslovna vrstica.

4. Če želimo v obseg celic, iz katere ustvarjamo tabelo, že vključiti naslovno vrstico, v pogovornem oknu **Ustvari tabelo** potrdimo **Moja tabela ima glavo** (slika 14.9). Če ne potrdimo Moja tabela ima glavo, bo naslovna vrstica dodana samodejno, kot je prikazano v koraku 2.

Ponovimo korak od 1 do 3, kjer v pogovornem oknu Ustvari tabelo, potrdimo Moja tabela ima glavo in kliknemo V redu.



Slika 14.9: Ustvari tabelo.

5. Besedilo v naslovni vrstici lahko spremenimo tako, da kliknemo v celico, vnesemo želeno besedilo v vnosno vrstico in potrdimo z Enter (slika 14.10).

C - - - Zve					ezek1 - Mic	rosoft Excel
C	Osno	vno Vstav	ljanje Post	avitev strani	Formule	Podatki
Ime tabele: Tabela7 ·↓· Spremeni velikost tabele		Povzemi z vrtilno tabelo Odstrani dvojnike Pretvori v obseg		Izvozi C	Dsveži Š	
	B2	• ((× ✓ f _x	Analiza 1	<u></u>	
	А	В	С	D	E	F
1						
2		Analiza 1	Stolpec2 🔽	Stolpec3 💌		
3						
4						
5						
6						
7						

Slika 14.10: Spreminjanje besedila v celici tabele.

14.4 Samooblikovanje tabele

Za samooblikovanje tabele oziroma preglednice uporabimo gumbe v skupini **Slogi** na kartici **Osnovno** (slika 14.11). Excel ponuja že pripravljene oblike tabel (slika 14.12). Če določenega oblikovanja ne želimo, ga izključimo, tj.izbrano obliko pisave, senčenja, obrob, oblik izpisa podatkov, poravnav. Vsaka izmed vnaprej pripravljenih oblik ima svoje ime.Tako oblikovane tabele lahko nadalje po želji oblikujemo. Vse oblikovne nastavitve preglednice izničimo, če za izbrane celice v seznamu **Oblika tabele** izberemo **Brez**.



Slika 14.11: Slogi samooblikovanja tabel.



Slika 14.12: Pripravljene oblike preglednic.

14.5 Določanje širine stolpcev in vrstic

Preglednice se največkrat začnejo z besedilom (naslovom), kateremu sledijo tekstovni in številčni podatki. Besedila vnašamo zato, da z njimi razlagamo pomen številčnih podatkov to so npr. meseci, leta, artikli, prodajalne, države. Natipkano besedilo se izpisuje preko desnega roba celice, če je sosednja celica prazna oziroma se pokaže le del besedila (toliko, kolikor se ga da zapisati v izbrano širino stolpca). Če je stolpec preozek, je preostali del besedila seveda neviden (se skrije). Čezenj se izpiše vsebina sosednje (desne) celice. V takih primerih prilagodimo širino celice njeni vsebini:

- z miško se postavimo na mejo med stolpcema ter odvlečemo rob ali
- dvakrat kliknemo na meji med stolpcema.

Na enak način določimo višino vrstic. Več možnosti za določanje bomo našli na kartici **Osnovno** s klikom na ukaz **Oblika** (celice, slika 14.13) oziroma desnim klikom na izbranih stolpcih oziroma vrsticah (slika 14.14):





Slika 14.14: Označimo stolpec in desni klik.

Slika 14.13: Oblika celic.

Problemi:

14.1 Na primeru 13.1 vadite oblikovanje podatkov:

- poravnavo,
- izbiro pisave,
- obrobo,
- polnilo.

14.2 Sedaj kreirajte tabelo z eno od pripravljenih oblik tabel (slogi). Vadite kako:

- brišemo celice,
- vrinemo celice,
- spremenimo barvo celic.

15. FORMULE IN FUNKCIJE

V delovnih zvezkih programa Microsoft Excel lahko naredimo veliko več, kot samo shranimo in uredimo podatke. Ena izmed pomembnih funkcij je, da lahko vrednosti v posameznih celicah seštejemo. Izračunamo lahko poprečno vrednost, določimo najnižjo ali najvišjo vrednost v skupini ali s podatki izvedemo nešteto drugih izračunov. Pogosto se zgodi, da podatke v isti celici uporabimo pri več kot enem izračunu. Excel omogoča preprosto sklicevanje na številne celice hkrati in s tem izdelavo izračunov.

15.1 Formule

Ko podatke dodamo v delovni zvezek, jih povzamemo s formulami. Formula je izraz, s katerim s podatki izvedemo izračun.

Vnos računskih operacij in formul se vedno začenja z enačajem (znak =). Če znaka ne vpišemo, Excel ne ve, da vnašamo formulo (misli, da želimo vnesti besedilo, število ali še kaj drugega). Pravilo velja tako za vnos formul kot funkcij.

15.1.1 Vrste formul

Vrsta	Primer
Aritmetične formule : - rezultat je številčna vrednost	=A1*200/A5 =SUM (A10:G10) =A5+3
Znakovne formule : - obdelujejo znakovne podatke, - rezultat je znakovni podatek.	="vreme"&"je lepo" =LEFT (A1;3) =B1&B2
Logične formule: - rezultat je logična vrednost TRUE (pravilno) ali FALSE (nepravilno).	=A1>19

15.1.2 Uporaba računskih operatorjev v formulah

Operatorji določajo vrsto operacije, ki jo želimo izvesti z elementi formule. Obstaja privzeti vrstni red po katerem poteka računanje, vendar ga lahko spremenimo z uporabo oklepajev.

Obstajajo štiri vrste računskih operatorjev: aritmetični operatorji, operatorji primerjave, besedilni operatorji spajanja in operatorji sklicev.

ARITMETIČNI OPERATORJI

Za izvajanje osnovnih matematičnih operacij, kot je npr. seštevanje, odštevanje, množenje, sestavljanje številskih vrednosti in ustvarjanje številskih rezultatov, uporabljamo naslednje aritmetične operatorje:

Aritmetični operator	Pomen	Primer
+ (znak plus)	Seštevanje	4+4
- (znak minus)	Odštevanje	4-2
* (zvezdica)	Množenje	2*2
/ (poševnica)	Deljenje	2/2
 ^ (strešica) 	Potenciranje	3^2
% (znak za odstotek)	Odstotek	20 %

BESEDILNI OPERATOR SPAJANJA

Uporabljamo znak & za združevanje ali spajanje enega ali več besedilnih nizov.

OPERATORJI PRIMERJAVE

Z naslednjimi operatorji primerjamo dve vrednosti. Rezultat je logična vrednost TRUE ali FALSE.

Operator primerjave	Pomen	Primer
= (enačaj)	Enako	A1=A2
> (znak večji od)	Večji od	A1>A2
< (znak manjši od)	Manjši od	A1 <a2< td=""></a2<>
>= (znak večji ali enak)	Večji kot ali enak	A1>=A2
<= (znak manjši ali enak)	Manjši kot ali enak	A1<=A2
<> (neenačaj)	Ni enako	<>

OPERATORJI SKLICEV

Za združevanje obsega celic uporabljamo operatorje sklicevanja. Poznamo naslednje operatorje:

Operator sklicev	Pomen	Primer
: (dvopičje)	Operator obsega, ki ustvari en sklic na vse celice med obema začetnima sklicema, vključno s tema.	C6:C15
; (podpičje)	Operator unije, ustvari en sklic. Ta vsebuje oba začetna sklica.	SUM (C6:C15; D5:D10)
(presledek)	Operator preseka, ki ustvari en sklic na celice, ki so skupne obema začetnima sklicema. V tem primeru je celica B7 skupna obema obsegoma.	SUM (B5:B15 A5:D7)

15.1.3 Vrstni red izvajanja operacij v formulah

V nekaterih primerih lahko vrstni red izvajanj računskih operacij vpliva na izračunano vrednost formule, zato je pomembno, da razumemo kako določiti vrstni red in kako ga spremeniti, da dobimo želeni rezultat.

Formule računajo vrednosti po določenem vrstnem redu. Kot smo že povedali, se formula vedno prične z znakom za enačaj (=). Ta pove Excelu, da naslednji znaki določajo formulo.

Če imamo v eno formulo vključenih več operatorjev, Excel izvaja operacije po določenem vrstnem redu, ki ga sam že pozna. Če formula vsebuje enakovredne operatorje, Excel preračunava operatorje od leve proti desni.

Če želimo spremeniti vrstni red preračunavanja, postavimo del formule, ki naj se najprej izračuna, v oklepaje. Excel tako najprej opravi vse operacije v oklepajih. Če je več oklepajev rešuje najprej najbolj notranje oklepaje, ter računa dokler ne doseže najbolj zunanjih oklepajev.

Primer 15.1: Izdelovanje preprostih formul.

Formula je zaporedje matematičnih operacij, npr. 2+3*4. Da nam Excel izračuna rezultat formule, moramo v celico vtipkati =2+3*4, potrdimo z **Enter** in dobimo število 14.

Primer 15.2: Izdelovanje kompleksnejših formul.

Dano imamo nekoliko kompleksnejšo formulo: $\frac{(C2+D2)\times 12+A2^{B2}}{E2}$

Formulo oblikujmo tako, da jo Excel izračuna: = $((C2 + D2) * 12 + (A2^{\land}B2)) / E2$

V našem primeru je Excel najprej seštel celici C2 in D2. Nato je izvedel računanje v naslednjem oklepaju oziroma potenciranje B2 na A2. Potem je rezultat C2+D2 pomnožil z 12 in nato se ta oba rezultata (C2+D2)*12 in (A2^B2) seštejeta. Na koncu se rezultat deli z E2.

15.1.4 Uporaba sklicev v formulah

Vsaka formula ima poleg operatorjev tudi vrednosti. Namesto vrednosti pa lahko v formulo vstavimo sklice. Sklic določa, v katerih celicah so vrednosti ali podatki, ki jih želimo uporabljati v formuli. S sklici lahko podatke, ki so v različnih delih delovnega lista, uporabimo v eni formuli ali pa uporabimo vrednosti ene celice v več formulah. Prav tako se lahko sklicujemo na celice, ki so v drugih delovnih listih istega delovnega zvezka in na druge delovne zvezke. Primeri sklicev so prikazani v nadaljevanju.

Za sklic na	V vnosno vrstico pišemo
Celica v stolpcu A in vrstici 10	A10
Obseg celic v stolpcu A in vrsticah od 10 do 20	A10:A20
Obseg celic v vrstici 15 in stolpcih od B do E	B15:E15
Vse celice v vrstici 5	5:5
Vse celice v vrsticah 5 do 10	5:10
Vse celice v stolpcu H	H:H
Vse celice v stolpcih H do J	H:J

Poznamo tri načine sklicevanja:

1. Relativno sklicevanje

Relativen sklic v formuli temelji na relativnem položaju celice, ki vsebuje formulo in celico, na katero se sklic nanaša. Če se mesto celice, ki vsebuje formulo, spremeni, se spremeni tudi sklic. Če formulo prekopiramo ali zapolnimo prek vrstic ali preko stolpcev, se sklic samodejno prilagodi (slika 15.1).

	Α	В
1		
2		=A1
3		=A2

Slika 15.1: Prekopirana formula z relativnim sklicem.

2. Absolutno sklicevanje

Včasih želimo, da se prekopirane formule sklicujejo na popolnoma isto celico kot original. To dosežemo tako, da pred sklic napišemo znak \$. Če se mesto celice, ki vsebuje formulo, spremeni, absoluten sklic ostane nespremenjen. Če formulo prekopiramo ali zapolnimo preko vrstic ali preko stolpcev, se absolutni sklic ne spremeni (slika 15.2).

	A	В
1		
2		=\$A\$1
3		=\$A\$1

Slika 15.2: Prekopirana formula z absolutnim sklicem.

3. Mešano sklicevanje

Mešani sklic vsebuje ali absolutni stolpec in relativno vrstico ali absolutno vrstico in relativni stolpec (slika 15.3). Sklic na absolutni stolpec jemlje oblike \$A1, \$B1, \$C1 itd. Sklic na absolutno vrstico jemlje oblike A\$1, B\$1, C\$1 itd. Če se položaj celic, ki vsebujejo formulo, spremeni, se relativni sklic spremeni, medtem ko se absolutni sklic ne. Če formulo prekopiramo ali zapolnimo preko vrstic in prek stolpcev, se relativni sklic samodejno prilagodi, absolutni pa ne.



Slika 15.3: Prekopirana formula z mešanim sklicem.

15.2 Funkcije

Pri obdelavi podatkov se nekatere formule ves čas ponavljajo (vsote, povprečne vrednosti). Spet druge formule so prezahtevne, ker imajo tako dolg zapis, da ga ne moremo več nadzorovati. To težavo lahko rešimo z že vgrajenimi funkcijami, ki jih ima program Excel.

Funkcije se od formul razlikujejo po tem, da so že vnaprej določene. Funkcija je v bistvu zamenjava za točno določen vrstni red operacij med podatki. Določena je z imenom funkcije (ki pove, kaj naj izvedemo s podatki) ter parametri (podatki ali naslovi celic, s katerimi računa).

Excelove funkcije so razporejene v več razredov:

- Matematična funkcija (najmanjše ali največje vrednosti, povprečja, trigonometrične funkcije, asimptote, absolutne vrednosti,...).
- Statistične funkcije (povprečne vrednosti, standardne deviacije, ...).
- Znakovne funkcije (rezanje, lepljenje, štetje znakov,...).
- Logične funkcije (testiranje kriterijev, logični testi).
- Finančne funkcije (obračun amortizacije, kredita, obresti,...).
- Datumske in časovne funkcije (število dni med dvema datuma, dan v tednu,...).
- Funkcije podatkovne zbirke (nanašajo se na izdelano podatkovno bazo).

15.2.1 Izbor funkcije

Funkcije lahko izbiramo:

- med desetimi najpogosteje uporabljenimi funkcijami,
- po posameznih skupinah funkcij ali
- neposredno po abecednem seznamu funkcij.

Opis nekaterih matematičnih funkcij:

Funkcije	Opis	Argument	Primer
ABS	Vrne absolutno vrednost števila, število brez predznaka	realno število	=ABS(-6) Matematično: $\left -6\right = 6$
COS	Izračuna kosinus kota.	v radianih	=COS(2) Matematično: cos (2) = - 0,416
EXP	Vrne e na potenco navedenega števila.	Število je eksponent z osnovo e.	=EXP(1) Matematično: e ¹ = 2,71828
INT	Število zaokroži navzdol do najbližjega celega števila.	realno število	=INT(15,7) Rezultat: 15
LOG10	Vrne desetiški logaritem števila.	pozitivno, realno število	=LOG10(2) Rezultat: 0,3010
MDETERM	Vrne determinanto matrike.	Matrika je številska matrika z enakim številom vrstic in stolpcev.	
POWER	Vrne potenco števila.	Katerokoli realno število.	=POWER(2;3) Matematično: 2 ³ = 8
SQRT	Vrne pozitivni, kvadratni koren števila.	Realno število večje ali enako nič.	=SQRT(4) Matematično: $\sqrt{4} = 2$

Opis nekaterih **statističnih funkcij**, ki jih najpogosteje uporabljamo:

Funkcije	Opis	Primer
AVERAGE	Izračuna povprečje	=AVERAGE(A1:A5) nam izračuna povprečje vrednosti v bloku celic od A1 do A5
SUM	Sešteje vsa števila v obsegu celic	=SUM(A1: B1) nam sešteje vrednosti celic A1 in B1
COUNT	Prešteje celice, ki vsebujejo števila	=COUNT(A1:A7) nam prešteje celice v bloku od A1 do A7
MAX	V bloku celic najde največje število	=MAX(A1:A7) nam izpiše največje število iz bloka od A1 do A7
MIN	V bloku celic najde najmanjše število	=MIN(A1:A7) nam izpiše najmanjše število iz bloka od A1 do A7

Če želimo bolje spoznati funkcije, ki so na voljo v Excelu, odpremo pogovorno okno **Vstavi** funkcijo in si na seznamu ogledamo funkcijo ter tiste, ki nas zanimajo, s klikom odpremo (slika 15.4).

	- (°I -)	Ŧ		Zvezek1 - Microsoft Excel	
Osno	vno Vst	avljanje	Postavitev s	Vstavi funkcijo	? 🗙
Vrtilna Tabel tabela *	a Slika	 Izrezki Oblike ▼ SmartArt 	Stolpčni	Iskanje funkcije: Na kratko napišite, kaj želite storiti, in nato kliknite »Pojdi«	Pojdi
Tabele	Ilu	stracije		Izberite zvrst: Vse 🗸 🗸 🗸	
C4		• (• 🗙 🗸	<i>f</i> _x =	Izberite funkcijo: Vse	
A 1 2 3 4 5 6 7 8	B	C	D	COMBIN COMPLEX CONPLEX CONCATENAT CONFIDENCE CONVERT CORREL Zbirka podatkov COS Besedilo Logične Informacije Vrne absolutno Inženiring	
9 10	+1 /Lin+2	/1ict2 /0		Pomoč za to funkcijo V redu	Prekliči

Slika 15.4: Funkcije, ki so na voljo v Excelovi knjižnici.

Podatke o funkcijah dobimo tako, da si ogledamo zaslonski namig, ki se prikaže ob funkciji oziroma kliknemo izbrano funkcijo in **Pomoč** za to funkcijo (slika 15.5).



Slika 15.5: Zaslonski namig funkcije COUNT.

Z naslednjim primerom bomo prikazali delovanje računskih operacij in statističnih funkcij v programu Excel 2007.

Primer 15.3: Uporaba statističnih funkcij.

Pri analizi dane molekule smo dobili naslednje vrednosti koncentracij, ki so prikazane v tabeli:

Številka analize	1	2	3	4	5	6	7	8
Koncentracija c/(ng/mL)	250,1	260,8	251,5	254,9	248,7	255,4	258,1	251,2

- Oblikujmo tabelo v programu Excel.
- Izračunajmo število meritev.
- Izračunajmo povprečno vrednost meritev.
- Izračunajmo minimalno vrednost meritev.
- Izračunajmo maksimalno vrednost meritev.

Potek dela:

- 1. Ročno vnesemo številke in koncentracije meritev v Excelu 2007 pripravljeno tabelo.
- **2.** Izvedemo statistično obdelavo analiznih rezultatov z uporabo Excelovih funkcij COUNT, AVERAGE, MIN in MAX, kot prikazuje slika 15.6.

	B12 \checkmark f_{x} =AVERAGE(B3:B10)					
	А	В	С	D	E	F
1						
		Izmerjena koncentracija				
2	Številka analize	(ng/mL)	Exce	elova funkci	ja	
3	1	250,1		UNT(B3:B10)	
4	2	260,8	/ 			
5	3	251,5		Excel	ova funkcija	za
6	4	254,9		izraču	in povprečn	e
7	5	248,7		vredn	OSTI: PACE(B3-B1	0)
8	6	255,4	1 /	-		•/
9	7	258,1	$\left \right $			
10	8	251,2	$\left[\right] /$	Excelova	unkcija za	
11	Število meritev:	8		vrednosti	inimaine	
12	Povprečna vrednost:	253,8	\checkmark	=MIN(B3:	310)	
13	Min	248,7		-		
14	Max	260,8				
15			Exo	elova funkc	ija za	
16			vre	un maxima dnosti:	ine	
17			=M	AX(B3:B10)		
18						
19						

Slika 15.6: Rešitev primera.

15.3. Vnos podatkov z enakomernim prirastkom

Če želimo npr. izračunati vrednosti neke odvisne spremenljivke pri različnih vrednostih neodvisne spremenljivke in se le-ta enakomerno spreminja (pada ali narašča), lahko uporabimo opcijo **Fill** v zavihku **Home**.

Najprej na vrhu stolpca vnesemo prvo vrednost spremenljivke, nato označimo stolpec, v katerem bodo vnešene še ostale vrednosti. Kliknemo opcijo **Fill** in iz padajočega seznama izberemo možnost **Series**. V oknu, ki se pojavi izberemo **Columns**, nato **Linear** (če želimo enakomerni prirastek vrednosti). Pri **Step** vpišemo vrednost prirastka in kliknemo **OK**.

Primer 15.4: Vnos zaporedja vrednosti z enakomernim prirastkom. Pri šestih različnih temperaturah izračunajte parni tlak vode po Antoinovi enačbi. Začetna temperature naj bo 10 °C in naj enakomerno narašča po 10 stopinj.

Enačba je naslednja:

$$\log p^{\rm nas} = A - \frac{B}{C+T} ,$$

temperatura *T* naj bo v °C, p^{nas} pa v mmHg. Vrednosti Antoinovih konstant so naslednje: A = 8,10765, B = 1750,286 in C = 235.

Rešitev:

V osnovnem Excelovem oknu vnesemo prvo vrednost za temperaturo (A5) nato pa označimo stolpec, v katerega bi radi vnesli prirastke temperature (A5 do A10). V zavihku **Home** izberemo opcijo **Fill**. Odpre se okno **Series** v katerem izberemo **Columns** in **Linear**. Pri **Step value** za naš primer vpišemo vrednost 10 (slika 15.7).

	A5	- (fs fs	10	
	А	В	С	D	E
1					
2	Izračun pa	rnega tlaka:		Antoinove	e konstante:
3				A=	8,10765
4	T/°C	p ^{nas} /mmHg		B=	1750,286
5	10			C=	235
6		Series			? ×
7		-Series in-		D;	ate unit
8		C Down	(All all all all all all all all all all		Dau
9				rowth (- Weekday
10				ate (Month
11				utoFill	Year
12					
13		Trend			
14		Step value:	10	St <u>o</u> p value:	
15			, ·		1
16				OK _	Cancel
17					

Slika 15.7: Uporaba opcije Fill za vnos zaporedja števil.

Ko kliknemo **OK** se pojavijo vrednosti temperatur. V stolpcu B5 do B10 izračunamo parni tlak.

	B5	- (fs fs	=10^(\$E	\$3-\$E\$4/(\$E\$	5+A5))
	А	В	С	D	E	F
1						
2 Izračun parnega tlaka vode:			Antoinove	e konstante:		
3				A=	8,10765	
4	T/⁰C	p ^{nas} /mmHg		B=	1750,286	
5	10	9,20		C=	235	
6	20	17,53				
7	30	31,83				
8	40	55,33				
9	50	92,53				
10	60	149,44				

Slika 15.8: Izračun parnega tlaka vode po Antoinovi enačbi.

Problemi:

15.1 Izračunajte vsoto števil v stolpcih A in B. Uporabite ustrezno funkcijo!

Α	В
13	7
12,3	1,7
11	19
-15	20

15.2 V tabeli izračunajte vrednost izraza: a + b - c =, če so podatki naslednji:

а	b	С
10	12	2
3	6	8
18	2	10

15.3 V tabeli izračunajte vrednosti izraza: $a^2 + b^2 = a$, če so podatki naslednji:

а	b
1	1
2	2
3	3
4	4

15.4 V tabeli izračunajte vrednosti izraza: (a+b) – (c+d) = , če so podatki naslednji:

а	b	С	d
1	2	1	3
2	2	2	3
3	2	3	3

15.5 Parni tlak kemijskih spojin pri določeni temperaturi lahko izračunamo iz Antoinove enačbe:

$$\log p^{\text{nas}} = A - \frac{B}{C+t} \quad \text{,} \qquad \text{kjer je } p^{\text{nas}} \text{ v mmHg in } t \text{ v } \circ \text{C!}$$

Z Excelom izračunajte parni tlak vode in etanola pri $t = 50 \degree$ C, če so podatki naslednji⁴:

	A	B	С
voda	8,10765	1750,286	235,0
etanol	8,1122	1592,864	226,184

Vprašanja za preverjanje znanja:

- 1. Na kakšne načine lahko obdelujemo podatke na delovnih zvezkih?
- 2. S kakšnim znakom se prične vnos računskih operacij in formul?
- 3. Kakšne vrste formul poznate pri delu z Excelom?
- 4. Kateri računski operatorji se uporabljajo v Excelu?
- 5. Katere aritmetične operatorje poznate?
- 6. Kakšno nalogo opravlja besedilni operator?
- 7. Kakšno nalogo imajo operatorji primerjave? Naštejte jih!
- 8. Katere operatorje sklicev poznate? Kdaj jih uporabljamo?
- 9. Katere načine sklicevanja na celice poznate?
- 10. Katere funkcije so že vgrajene v Excel?

⁴ Felder R. M., Rousseau R. W., Elementary Principles of Chemical Processes, Third Edition, John Wiley & Sons, Inc.New York, 2000.
16. GRAFIKONI

Z vnašanjem podatkov na delovni list programa MS Excel izdelamo zapis pomembnih dogodkov, npr. prodaje posameznih izdelkov, cene izdelkov. Vendar pa dolgi seznam vrednosti v celicah ne predstavlja splošnih trendov podatkov. Najboljši način za prikaz trendov v obširnih zbirkah podatkov je izdelava grafikonov oziroma grafov, ki vizualno povezujejo podatke. Kako na primer z grafikoni podatke predstavimo učinkoviteje, je razvidno iz naslednje tabele za beleženje stroškov proizvodnje etanola (slika 16.1).

Kategorija	Strošek (EUR/leto)
Surovine	2688815
Pogonska	30684598
sredstva	
Zaposleni	5179000
Laboratorij	800000
Ostalo	15000000

Slika 16.1: Tabelarični prikaz podatkov.

Podatki v nekem smislu povedo svoje. Posamezni podatek je potrebno razumeti, si ga zapomniti in ga primerjati z drugimi rezultati. Ko rezultate predstavimo z grafikonom ali grafom, kot je prikazano v naslednjem stolpičnem grafikonu (slika 16.2), je primerjanje vrednosti lažje.



Slika 16.2: Grafično prikazovanje podatkov.

Grafikoni so dinamični, kar pomeni, da sprememba podatkov v preglednici pomeni samodejno spremembo grafikona.

16.1 Elementi grafikona

Grafikon je slikovna predstavitev podatkov iz preglednice v dvo ali trirazsežnem koordinatnem sistemu. Dvodimenzionalni grafikoni prikazujejo na osi X neodvisne podatke, na osi Y pa odvisne podatke. V tridimenzionalnih grafikonih se pojavi še tretja oziroma Z os.

Ni grafikona brez podatkov. Za grafikon tako potrebujemo vsaj dva niza podatkov, odvisne in neodvisne podatke. Najenostavnejši izmed vseh grafikonov je tortni, kateremu zadošča tudi samo en niz podatkov.

Grafikoni morajo biti opremljeni tako, da je iz njih jasno razvidno, kaj in v katerih merskih enotah so prikazani podatki. Imeti morajo nedvoumen naslov in po možnosti vir podatkov. Vsak grafikon sestavlja več elementov: osi, naslov grafikona, naslovi osi X in Y, legenda, ozadja (slika 16.3).



Slika 16.3: Elementi grafikona.

1. NASLOV GRAFIKONA

Naslov naj bo kratek, vendar jasen in točen. Pomen podatkov se lahko hitro spremeni, če naslov grafikona ni pravilen ali je pomanjkljiv. Kot naslov lahko uporabimo tudi vprašanje ali pa navadno trditev. Naslov naj bo lepo grafično oblikovan, vsako oblikovno pretiravanje pa lahko škodi sporočilnosti informacije.

2. NASLOV OSI X

3. NASLOV OSI Y

Naslov osi kratko opisuje vrsto podatkov, prikazanih na osi. Pomembno je, da zapišemo mersko enoto, v kateri so podatki predstavljeni. Vrsto pisave in položaj izpisa lahko poljubno spreminjamo.

4. LEGENDA

Opisuje posamezne nize podatkov.

5. OS X

Prikazuje neodvisne podatke, nanjo lahko namestimo:

- znakovne podatke (imena podjetij, držav, artiklov, oseb, ...),
- datumske ali časovne podatke (dnevi, meseci, leta),
- številčne podatke (seznam števil).

Oblikovanje osi X določi program sam, lahko pa tudi sami spremenimo črtice na osi, vrsto in orientacijo izpisa podatkov, merilo, pisavo.

6. OS Y

Prikazuje odvisne podatke (npr. nabavne stroške za posamezne artikle, vrednost temperature po krajih, gibanje deviznih tečajev,...). Podatki na osi Y so največkrat prikazani v:

- decimalni obliki,
- denarni obliki,
- odstotkovni obliki.

7. POVRŠINA GRAFIKONA

8. RISALNA POVRŠINA

Oblikujemo jo za lepši videz grafikona (obarvano, občrtano).

9. NIZI PODATKOV

Vsak stolpec, tortni ali kolobarni izsek, krivulja na grafikonu predstavlja en niz podatkov. Na posamezne nize podatkov lahko zapišemo njihove številčne ali znakovne vrednosti.

Vsakemu od teh elementov lahko spremenimo lastnosti (vrsta ali oblika pisave, barva ozadja, poravnava, ...) in tako oblikujemo grafikon po svoji meri.

16.2 Vrste grafikonov

Izbira vrste grafikona ni enostavna. Podatki zahtevajo točno določen tip grafikona in tako moramo zmeraj dobro premisliti, kako bi z izbrano vrsto grafikona najučinkoviteje predstavili podatke. Najljubši tip grafikona ni vedno najprimernejši.

Excel ponuja paleto osnovnih vrst grafikonov, vsaka od njih pa ima še svojo podvrsto (podtip grafikona). V osnovi ločimo tri vrste grafikonov:

- 1. histograme (stolpčni, palični),
- 2. površinske grafikone (tortni, ploščinski, kolobarni),
- 3. linijske grafikone (črtni, polarni).

16.2.1 Stolpčni grafikon

Stolpčni grafikon prikazuje vsako serijo kot zaporedje enako obarvanih stolpcev. Uporabljamo ga takrat, ko prikazujemo med seboj neodvisne podatke (primerjavo med podatki).

Najprimernejši so za prikaz: učinkovitosti, kakovosti, dosežkov, donosov vertikalnih podatkov (višina padavin, višina objektov ...) in njihovih nasprotij (slika 16.4).



Slika 16.4: Stolpčni grafikon.

16.2.2 Palični grafikon

Palični grafikon je za 90° obrnjen stolpčni grafikon. Zelo nazoren je v primerih, ko so na X osi podani podatki tekstualnega tipa (zelo lahko izpišemo daljše tekstualne nazive). Poudarek je predvsem na vrednosti in manj na času.

Najprimernejši so za prikaz: hitrosti (vetra, tekmovalcev, avtomobilov ...), tekmovalnosti, horizontalnih podatkov (dolžina objektov, razdalja ...), reakcijskih časov (slika 16.5).



Slika 16.5: Palični grafikon.

16.2.3 Črtni grafikon

Črtni grafikon je primeren za prikaz gibanja pojava v daljšem časovnem obdobju z enakomernimi presledki. Linija, ki povezuje podatkovne točke, je zvezna, če je obdobje dovolj dolgo.

Najprimernejši so za prikaz (slika 16.6): gibanja tečajev valut, delnic, populacijskih sprememb (prebivalcev, živali), trendov naraščanja in padanja (proizvodnje, prodaje, menjave).



Slika 16.6: Črtni grafikon.

16.2.4 Tortni grafikon

Tortni grafikon prikazuje razmerje med deli, ki predstavljajo celoto. Vedno prikazuje eno samo serijo podatkov, ki skupaj predstavljajo 100 %. Pri tortnem grafikonu lahko uporabimo "eksplozijo rezin".

Najprimernejši so za prikaz: deležev in sestave (slika 16.7).





16.2.5 Ploščinski grafikon

S ploščinskim grafikonom najlepše ponazorimo spremembe razmerij med deli v določenem času in razmerje delov glede na celoto.

Najprimernejši so za prikaz: gibanja moške in ženske populacije, gibanja generacijskih populacij, spreminjanja deležev, spreminjanja sestave (slika 16.8).





16.2.6 Polarni grafikon

Polarni grafikon je zelo uporaben v statistiki. Njegova značilnost je, da ima vsak podatek svojo Y os. Najprimernejši je za prikaz: analiz anketnih vprašalnikov (slika 16.9).



Slika 16.9: Polarni grafikon.

16.2.7 Tridimenzionalni grafikoni (3D)

Tridimenzionalni grafikoni so zelo podobni dvodimenzionalnim predhodnikom. Dodana jim je še tretja dimenzija. Zelo opazni so zaradi svojega izgleda, vsebinsko pa so skoraj enaki. Ločimo:

- 3D ploščinski grafikon,
- 3D palični grafikon,
- 3D stolpčni grafikon,
- 3D črtni grafikon,
- 3D tortni grafikon,
- 3D površinski grafikon.

Najprimernejši so za prikaz (slika 16.10): podatkov, ki se po vrednosti zelo razlikujejo (delne vrednosti in skupna vrednost, dva cenovna razreda enakih artiklov).





16.3 Izdelava grafikonov

16.3.1 Hitro izdelovanje grafikonov

Če želimo Excelove podatke grafično predstaviti, označimo celice, ki jih želimo povzeti, kliknemo jeziček Vstavljanje (Insert) in nato izberemo vrsto grafikona, ki je najprimernejši za podatke in sporočilo, ki ga želimo prenesti drugim.

V Excelu 2007 je izdelovanje privlačnih grafikonov veliko lažje kot v prejšnjih različicah programa. Ne samo, da s kontrolniki na traku hitro izdelamo grafikone, ampak lahko z njimi izberemo tudi celotno postavitev in slog grafikona. Ko nastavimo splošni videz grafikona, brez težav vnesemo katero koli želeno spremembo.

Primer 16.1: Izdelava grafikonov na osnovi spodnjih podatkov (slika 16.11).

Celotni ocenjeni stroški za proizvodnjo etanola, za leto 2005 so naslednji:

- Surovine: 2 688 815 EUR
- Pogonska sredstva: 30 684 598 EUR
- Zaposleni: 5 179 000 EUR
- Laboratorij: 80 000 EUR
- Ostalo: 15 000 000 EUR



Izdelovanje grafikona:

- Kliknemo celico na seznamu podatkov, ki ga želimo povzeti.
- Kliknemo jeziček Vstavljanje.
- 3. Kliknemo vrsto grafikona.
- Kliknemo želeno podvrsto grafikona.
- 5. Izris grafikona.

Slika 16.11: Prikaz hitrega izdelovanja grafikonov.

G		🚽 🌒 = (°! =)	Ŧ	Izraču	n 4.1 - Mi	icrosoft Ex	cel		Orodja za	grafikone		-		x
<u> </u>	2	Osnovno Vstav	ljanje Postavite	v strani	Formule	Podatki	Pregled	Ogled	2 Načrt	Postavitev	Oblika	0.		х
Spr	reme grafik	ni vrsto Shrani kot predlogo Vrsta	Preklopi vrstico/stolpec Podat	Izberi podatke	3 H Pos	litra tavitev •	H slo	itri ogi •	Premakni grafikon Mesto					
	G	rafikon 4	▼ (* f;	c		i "i								≯
	Α	В	С	D				1 Providencial		G	H	ł	- I	
1								Postavi	Lev 5					
2		Kategorija	Strošek (EUR/leto)											
3		Surovine	2688815			Fa L.	a la	-11-8			1		-	1
		Pogonska	30684598		1-8				EUI)	R/leto)	1			
4		sredstva			254		_		1					
5		Zaposleni	5179000		30		T.							
6		Laboratorij	800000	_	250				•					
7		Ostalo	15000000	•	200000	000		_						
8	-Г.	5				-								.≡
9	'	5	Stroš	ek (E	UR/le	eto)								
10	_	35000000									Strošek	(EUR/I	eto)	
12		3000000		_					- &	>0				
12		25000000							- rator	Ostai				H
14		2000000							— í					H
15		15000000							_					
16		10000000							-					Į I
17		5000000												9
18		0							-					
19		Su	urovine Pogo	onska	Zaposleni	Labora	torij	Ostalo						
20			sred	lstva										
14				Strošek (FUR/leto)								Þ	
Prip	ra			Sabser	2011/12/07					□ □ 100%	0	U		.:

Slika 16.12: Prikaz spreminjanja postavitve grafikona.



<u>Spreminjanje</u> postavitve grafikona (slika 16.12):

- Kliknemo grafikon, ki ga želimo spremeniti.
- 2. Kliknemo jeziček Načrt (Design).
- 3. Kliknemo Hitra postavitev.
- Izberemo želeno postavitev.
- 5. Izris želenega grafikona.

<u>Spreminjanje sloga</u> <u>grafikona (slika</u> <u>16.13):</u>

- Kliknemo grafikon, ki ga želimo spremeniti.
- 2. Kliknemo jeziček Načrt (Design).
- Kliknemo Hitri slogi.
- Izberemo želeni slog.
- 5. Izris grafikona.

Slika 16.13: Prikaz spreminjanja sloga grafikona.

16.3.2 Spreminjanje videza grafikona

Ko grafikon izdelamo, lahko kateri koli del videza spremenimo, vključno z vrsto grafikona. Spremenimo lahko tudi barvo, pisavo in druge lastnosti katerega koli elementa grafikona. Če na primer želimo naslov grafikona prikazati v uradni pisavi podjetja, ga preprosto oblikujemo.



Slika 16.14: Prikaz spreminjanja vrste grafikona.

C	🚽 🍤 -	(°" -) =	Izraču	Izračun 4.1 - Microsoft Excel						Orodja za grafikone			ε.
9	Osnovno	Vstavljanje	Postavitev strani	Formule	Podatki	Pregled	Ogled	Načrt	Postavitev	1 Oblika	0 -	. • · ·	ĸ
Obmo	čje grafikon	a				- 🙆 -		<u>A</u> -	🖫 Postavi	i v ospredje	- 🛱 -		
🥎 Izb	or oblike		Abc	Abc	Abc	- 🗹 -	- 2	2 -	📲 Pošlji v	ozadje 🔻	-⊕	Valilia	
餐 Ponastavi na ujemanje sloga						₹ 📿 -	slogi	- A -	Podok	no izbora	41-	venkos	
Trenutni izbor				Slogi oblik 🕞			Slogi WordArt 😼 🛛 Razporedi						

Slika 16.15: Spreminjanje oblikovanja vrste grafikona.

V primeru oblikovanja grafikona, kliknemo jeziček Oblika (Layout). S kontrolniki v skupini Slogi oblik oblikujemo grafikon.

16.3.3 Oblikovanje legend in naslovov grafikonov

Pri izdelavi poučnega in preprosto berljivega grafikona je zelo pomembno opisovanje vsebine grafikona. Legende, naslovi, podatkovne oznake in opombe so elementi grafikona, ki jih uporabimo pri opisovanju Excelovega grafikona.



Prikazovanje legende grafikona (slika 16.16):

- Kliknemo grafikon, ki ga želimo oblikovati.
- Kliknemo jeziček Postavitev (Layout).
- Kliknemo Legenda in želeno možnost prikaza legende.

Slika 16.16: Prikazovanje ali skrivanje legende grafikona.

C	📙 🗳 - (2 -	- =	Izračun 4.1 - Microsoft Excel Orodja za grafikone – 📼 🗙
	Osnovno Vs	tavljanje Postavit	ev strani Formule Podatki Pregled Ogled Načrt 2 Postavitev Oblika 🐵 – 📼 🗙
Naslo	v grafikona	• 🖓	3 🔜 🛛 🖬 Legenda *
🥎 Izl	bor oblike	Notes 1	Cznake podatkov *
and Po	nastavi na ujema	nje sloga 👻	grafikona v osi v 👜 Podatkovna tabela v v črte v v v v
	Trenutni izbor		Brez
	Grafikon 8	▼ (• .	K Ne prikaži naslova grafikona 🛛
	В	С	Prekrivni naslov na sredini G H I
1			Prekrivni naslov na sredini grafikona brez spremembe velikosti grafikona
		Strošek	Nad grafikonom
2	Kategorija	(EUR/leto)	Prikaži naslov na vrhu območja
3	Surovine	2688815	
	Pogonska	30684598	
4	sredstva		
5	Zaposleni	5179000	1 4 Strošek (FLIR /leto)
6	Laboratorij	800000	
7	Ostalo	15000000	35000000
8			3000000
9			2500000
10			
11			
12			5000000 Strošek (FLIR/leto)
13			
14			ine we entry in all
H + +	H List1 Li	st2 🖌 List3 📈 🖏	
Priprav	/ljen		🔲 🛄 100% 🕞 —

Slika 16.17: Dodajanje naslovov.

Dodajanje naslovov (slika 16.17):

- Kliknemo grafikon, ki ga oblikujemo.
- Kliknemo Postavitev (Layout).
- Kliknemo Naslov grafikona (Chart Title) in želeno možnost prikaza naslova.
- Kliknemo naslov in vtipkamo novi naslov.

16.3.4 Spreminjanje telesa grafikona

Z dodajanjem oznak, legend in opomb spremenimo obrobje grafikona, medtem ko na prikaz podatkov s tem ne vplivamo.

Če želimo prikaz podatkov v telesu grafikona vseeno spremeniti, na primer s prikazovanjem ali skrivanjem mrežnih črt ali spreminjanjem merila grafikona, lahko to naredimo z drugimi načini. Z dodajanjem mrežnih črt lahko gledalcem olajšamo razlikovanje med vrednostmi, tudi če ti gledajo grafikon z drugega konca mize, medtem ko s spreminjanjem merila grafikona poudarimo ali zmanjšamo razliko med vrednostmi.



Slika 16.18: Prikazovanje ali skrivanje mrežnih črt grafikona.

	Osnovno Vs	tavljanje Post	avitev	strani Formule Podati	i Pregled Ogled	Načrt 2	Postavitev Oblika	🕡 🗕 📼 🗙
bm lz	očje grafikona bor oblike	- 6	₽		genda • 3			T
P(onastavi na ujema	anje sloga	avi	Naslov Naslovi	datkovna tabela 🔻	Osi Mrežn	e Ozadje Analiza Las	tnosti
-	Trenutni izbo	r		Oznake		Primarna	a vodoravna os 🕨	
	Grafikon 8	- (9	f,	e	4	Primarna	a <u>n</u> avpična os 🕨	×
Α	В	С	D	E	F	G	Brez	
						iye.	Ne prikaži osi	
		Strošek					Pokaži privzeto os	
	Kategorija	(EUR/leto))				Prikaži os s privzetim vr oznakami	stnim redom in
	Surovine	2688815	1			1000	Pokaži os v obliki tisoč	ĩc
	Pogonska	30684598	ß				Prikaži os s številkami, obliki tisočic	predstavljenimi v
-	Zaposleni	5179000		1 St	trošek (FUR/	Le 1900000	Pokaži os v obliki milij	onov
	Laboratorii	800000					Prikaži os s številkami,	predstavljenimi v
	Ostalo	15000000		35000000	_	109	Pokaži os v obliki miliji	ard
	•	*		25000000		<u>l din</u>	Prikaži os s številkami, obliki milijard	predstavljenimi v
				20000000		LOG	Pokaži os z logaritems	kim merilom
				15000000			Prikaži os z uporabo m	erila na osnovi
			:	10000000		5 Ve	č možnosti za primarno	navnično os
				0				
				e s	6 6 6	10		
				urowith redst	poste orato ost	ġ.		
				5°	120, 20			
				4080.				
	L, _			_				
	► ► List1 / L	ist2 / List3 /	°.	/				
pra	vijen							🕈
	O	blikovanje (osi					
	6	Možnosti osi		Možnosti osi				
		Številka	N	lajmanjše: 🔿 Samo	deino 💿 Nespreme	nljivo 0.0	7	
		Poloilo		aivečie: O Samo	doine (Nespreme		· · · · · · · · · · · · · · · · · · ·	
		POINIO		Samo	dejno 🕑 Nespreme	nij <u>i</u> vo 3,5		
		Barva črt	V	ecja enota: 🔘 Sa <u>m</u> o	dejno 💿 Nesp <u>r</u> eme	nljivo 5,0	E6	
		Slog črte	M	lanjša enota: 🔘 Samo	dejno 💿 Nespreme	nlji <u>v</u> o 1,0	E6	
		Senčeno		Vrednosti v o <u>b</u> ratnem	vrstnem redu			
		3D-oblika		Logaritmično merilo	osnova: 10			
		JU-JUIIKā	F	note prikaza: Brez	*			
		Poravnava		Pokaži oznako enot pri	kaza na grafikopu			
			ľ	rsta pomozne crtice: B				
					oleg osi 🚩			
			K	rizci na vodoravni osi: Samo <u>d</u> ejno				
			0	Vrednost osi: 0.0				
			1	Naivečia vrednost osi				
				<u>najvecja vrednost Osl</u>				

Spreminjanje merila na osi vrednosti (osi Y):

- Kliknemo grafikon, ki ga oblikujemo.
- Kliknemo jeziček Postavitev (Layout).
- 3. Kliknemo Osi (Axes).
- Kliknemo Primarna navpična os.
- Kliknemo Več možnosti za primarno navpično os.
- Kliknemo Možnosti osi.
- 7. Vtipkamo:

- najnižjo vrednost na navpični osi

- najvišjo vrednost na navpični osi

 vrednosti za večje enote za na navpični osi

 vrednosti za manjše enote za na navpični osi.

8. Kliknemo Zapri (Close).

Slika 16.19: Spreminjanje merila na osi vrednosti.



Spreminjanje merila na osi kategorij

- Kliknemo grafikon, ki ga oblikujemo.
- Kliknemo Postavitev (Layout).
- 3. Kliknemo Osi
- 4. Kliknemo Primarna vodoravna os.
- Kliknemo Več možnosti za primarno vodoravno os.
- Kliknemo Možnosti osi.
- 7. Vnesemo želeni interval med črticami.
- 8. Izberemo gumb Določi enoto intervala.
- 9. Odkljukamo druge možnosti odstranimo kljukice pred
- 10. Kliknemo Zapri (Close).

Slika 16.20: Spreminjanje merila na osi kategorij.

16.3.5 Prilagajanje podatkov grafikona

Pri izdelovanju grafikona moramo vedno predvideti možnost, da se bodo podatki, prikazani v grafikonu spremenili. Če te spremembe zrcalijo nadaljnjo prodajo, posodobljene vrednosti, ki predstavljajo vrnjene izdelke in spremembe zaloge, ali spremembo ocene naložbe, ki je usklajena z novimi razmerami na trgu, grafikon posodobimo s prepoznavanjem novega vira podatkov.



Slika 16.21: Spreminjanje izvornih podatkov grafikona.



Slika 16.22: Dodajanje novega niza.

- Kliknemo grafikon, ki ga spreminjamo.
- 2. Kliknemo jeziček Načrt (Design).
- Kliknemo Izberi podatke (Select Data).
- Kliknemo Dodaj (Add).
- 5. Vtipkamo ime za niz.
- 6. Kliknemo v polje Vrednosti serij.
- Označimo celice,ki jih želimo dodati.
- Kliknemo V redu, da zapremo okno Urejanje nizov.
- Kliknemo v redu, da zapremo okno Izbira vira podatkov.

		- L) - (L -) =	Izrač	un 4.1 -	Microsoft	Excel		Orodj	a za grat	fikone		-	- 🗆 ×	Bri	sanie niza (slika
0	2	Snovno Vstav	janje Po	ostavitev strani	Formul	le Podatki	Pregled	Ogled	2 Načrt	t Po:	stavitev	Oblika	0 -	. 🗝 X	16	.23):
Spi	emen Irafiko	i vrsto Shrani ko predlogo Vrsta G13	ot Pr vrstic	reklopi Iz co/stolpec poc Podatki	beri latke	Hitra postavite ostavitve gr	v • afikona	Hitri slogi Slogi graf	ikona	Premakr grafikor Mesto	ni			~	1.	Kliknemo grafikon, ki ga spreminjamo.
	А	В		С	D		E		F	-		G		н	_	
1															2.	Kliknemo jeziček
2		Kategorija	S (El	trošek JR/leto)	350						1					Načrt (Design).
3		Surovine	20	688815	250	000000 -									2	Kliknomo Izbori
4		Pogonska sredstva	30	684598	200	000000 -									э.	podatke (Select
5		Zaposleni	5	179000	100	000000 +					Stroš	ek (EUR/let	o)			Data).
6		Laboratorij	8	00000	5	000000 +					Prihr	anek				,
7		Ostalo	15	000000		0 +									Л	Kliknomo imo
8							~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	ín.	8	~0					4.	
			Pr	ihranek	lzbira	vira poda	ikov							21		niza, ki ga
9		Kategorija	(El	JR/leto)	Obse	g podatkov v	grafikonu									želimo izbrisati.
10		Surovine	2	80000	Obseg p	podatkov je p	- oreobsežer	n, zato ga	ni mogo	če prikaz	ati. Če iz	berete nov o	bseg, b	0		
		Pogonska			zamenja	al vse nize na	plošči »Ni	izi«.							5	Kliknemo
11		sredstva	30	000000			(Pr	eklop <u>i</u> vr	rstico/sto	olpec				0.	odetroni
12		Zaposieni		50000	Vnosila	aende (nizi)		-			Ozna	ke vodoravn	ne osi (K	ategorija)		oustrani.
13		Laboratorij	4	700000		yende (<u>liizi</u>)	likoinnin		latrani					ategorijaj		
14		Ustaio	1.	/00000			Urejanje		Istrani			U <u>r</u> ejanje			6.	Kliknemo V
15					Stroše	k (EUR/leto)					1					redu.
10					Prihran	iek <mark>4</mark>					2					
10											3					
10											4					
20											5					
21					Skrite	in prazne ce	lice				6	Viredu		rekliči		
14		Liet1 Lie	·2 /1i+	2 / 27 /	Baue	in prozne Ce					, C	Teuu				
Prir	ravlie	n LISUI / LISU	.∠ ∕ LISL							nm	100%		1			
	. ange											0				

Slika 16.23: Brisanje niza.

Primer 16.2: Kreiranje vrelnega diagrama zmesi metanol/voda.

Na osnovi eksperimentalnih podatkov kreirajmo vrelni diagram za zmes metanol/voda⁵. Slika 16.24 prikazuje eksperimentalne podatke zmesi (celice A9:C25).

Potek kreiranja vrelnega diagrama:

Označimo stolpca, ki predstavljata x in y os (x_1 in T stolpca oziroma A9:C25), ter z raztresenim grafom (scater) narišemo **vrelno krivuljo** (rdeča krivulja na sliki 16.24).

Nato dodamo še **kondenzacijsko krivuljo** (vrednosti za y₁ oziroma stolpec C9:C25) po naslednjem postopku:

- kliknemo na zavihek Design in izberemo opcijo Select Data,
- odpre se okno Select Data Source,
- pri opciji Chart data range se pojavi informacija o podatkih, ki so zajeti v krivulji, ki že obstaja,
- če želimo dodati novo krivuljo v obstoječ graf, kliknemo Add,
- odpre se okno Edit Series v katerem:
 - > zapišemo ime krivulje (Series name),

⁵ Gmehling J., Onken U., Vapor-Liquid Equilibrium Data Collection, Chemistry Data Series Vol. I, Part 1, DECHEMA, Deutsche Gesellschaft f
ür Chemisches Apparatewesen, Frankfurt, 1977, str. 43.

- pri Series X values označimo celice, ki predstavljajo podatke za x os, tj. sestavo parne faze y₁ (C9 do C25),
- pri Series Y values označimo celice, ki predstavljajo podatke za y os, tj. temperaturo T (B9 do B25).
- kliknemo OK.



Slika 16.24: Kreiranje vrelne krivulje.



Slika 16.25: Dodajanje kondenzacijske krivulje.

Graf nato primerno opremimo (označimo osi, napišemo naslov, spremenimo barve krivulj ipd.), kot prikazuje slika 16.26.



Slika 16.26: Vrelna krivulja za zmes metanol/voda pri tlaku p = 1,013 bar.

16.3.6 Kreiranje logaritemskih in semilogaritemskih skal

Kadar je pri reševanju nekega problema primerneje prikazati odvisnost spremenljivk v logaritemskem koordinatnem sistemu, lahko v Excelu definiramo logaritemske skale.

Primer 16.3: Kreiranje logaritemskih skal.

Narišite graf $C_o = f(A)$, ki predstavlja odvisnost osnovne cene cevno/plaščnih toplotnih prenosnikov (TP) od površine prenosnikov in sicer v logaritemskem koordinatnem sistemu, tj. x in y os naj predstavljata logaritemski skali. Podatki so naslednji:

A/m ²	C _o /USD
20	3330
50	6040
100	9480
150	12335
200	14870
250	17190
300	19360
350	21400
400	23340
450	25190

Osnovno krivuljo prikazuje slika 16.27. Za kreiranje grafa smo izbrali graf Scater.



Slika 16.27: Odvisnost osnovne cene TP od površine.

Če spremenimo horizontalno in vertikalno osi v logaritemski skali, dobimo graf prikazan na sliki 16.28. Krivulja podaja sedaj linearno odvisnost med spremenljivkami. Tak način izračuna in prikaza osi je primeren, ko želimo določiti formulo za odvisnost dveh spremenljivk. Matematično lahko zapišemo: Y = A X + B in iz grafa določimo naklon premice, A in odsek na Y osi, B. Izberemo dve točki:

$$A = \frac{\log 12335 - \log 9480}{\log 150 - \log 100} = \frac{4,09114 - 3,97681}{2,176 - 2} = 0,65$$

 $B = Y - AX = 4,09114 - 0,65 \cdot 2,176 = 2,67674$

Sedaj zapišemo enačbo premice: $Y = 0.65 \cdot X + 2.67674$.

Če enačbo antilogaritmiramo, dobimo enačbo, iz katere lahko pri različnih površinah A izračunamo osnovno ceno TP:







Kako kreiramo graf z logaritemskimi skalami?

V zavihku **Insert** izberemo primeren graf (Scater) za prikaz podatkov. Nato v zavihku **Layout** kliknemo gumb **Axes** in izberemo opcijo **Primary Horizontal Axis**, če želimo definirati x os. Zatem izberemo **More Primary Horizontal Axis Options**. Odpre se okno za oblikovanje x osi (slika 16.29). Izberemo **Axis Options**.

Format Axis	? ×						
Axis Options	Axis Options						
Number	Minimum: <u>Auto</u> C <u>Fixed</u> 1,0						
Fill	Maximum: Auto C Fixed 1000,0						
Line Color	Major unit: Auto C Fixed 10,0						
Line Style	Minor unit: Auto C Fixed 10,0						
Shadow	Values in reverse order						
3-D Format	Logarithmic scale Base: 10						
Alignment	Display units: None						
	Major tick mark type: Outside 💌						
	Minor tick mark type: Inside						
	Axis labels: Next to Axis						
	Vertical axis crosses: O Aut <u>o</u> matic						
	Axis valu <u>e</u> : 1,0						
	O Maximum axis value						
	Close						

Slika 16.29: Okno za definiranje x osi.

Definiramo lahko poljubno merilo. Če želimo, da je x os predstavljena z logaritemsko skalo, to označimo pri zapisu **Logaritmic scale**, pri **Base** pa definiramo osnovo logaritma. Nato pri tekstu **Major tick mark type** izberemo npr. **Outside** (glavne črte merila bodo prikazane pod osjo) in pri **Minor tick mark type** npr. **Inside** (pomožne črte merila bodo prikazane nad osjo). Kliknemo **Close** in zapustimo okno.

Za bolj nazoren prikaz naj bo graf opremljen z logaritemsko mrežo. Če želimo narisati vertikalne črte mreže kliknemo na gumb **Gridlines**, izberemo **Primary Vertical Gridlines** in nato **Minor Gridlines**.

Na podoben način sedaj definiramo y os in horizontalno logaritemsko mrežo. V grafu lahko nato poimenujemo x in y osi, definiramo ime grafa ipd., kar smo se že naučili.

16.3.7 Prileganje krivulj (Curve Fitting), regresija (Regression)

Prileganje krivulj je proces v katerem želimo najti krivuljo (in je predstavljena z modelno enačbo), ki najbolje predstavlja podatke, ki so na voljo, oziroma bolj specifično poda zvezo med odvisnimi in neodvisnimi spremenljivkami v nizu podatkov.

a) Linearno prileganje (Linear Curve Fitting)

<u>Primer 16.4</u>: Regresija premice⁴. Podatki umerjanja rotametra so naslednji:

Odčitek na rotametru, <i>R</i> o	<i>q</i> ∨ (L/min)
10	22,0
30	55,1
50	81,6
70	118,3
90	154,0

Na osnovi podatkov generirajmo najboljšo ravno črto in pri tem uporabimo:

- trendno črto,
- funkcije SLOPE, INTERCEPT in RSQ.

Uporaba trendne črte

Po vnosu podatkov v Excel narišemo graf $q_v = f(R_o)$. Pri tem uporabimo graf **Scater**. Po kreiranju grafa v zavihku **Layout** kliknemo gumb **Trendline** in izberemo **More Trendline Options**. Izberemo linearno trendno črto ter opciji **Display Equation on Chart** in **Display R-squared value on chart**. S prvo opcijo dobimo najboljšo enačbo za naš graf, druga opcija pa prikaže kvadrate korelacijskega koeficienta R, tj. kazalnika, s katerim ocenimo kvaliteto prileganja krivulje k točkam (slika 16.30).

Format Trendline	<u> </u>
	Trendline Options
Line Color	inend/kegression rype
Line Style	C Exponential
515500	🗾 💿 Linear
	C Logarithmic
	C Polynomial Order: 2
	O Power
	C Moving Average Period: 2
	Trendline Name
	<u>A</u> utomatic : Linear (Series1)
	C <u>C</u> ustom:
	- Forecast
	Eorward: 0,0 periods
	Backward: 0,0 periods
	Set Intercept = 0,0
	Display Equation on chart
	Display <u>R</u> -squared value on chart
	Close

Slika 16.30: Definiranje opcij pri linearnem prileganju krivulje.

⁴ Felder R., Rousseau W., Elementary Principles of Chemical Processes, third edition, John Wiley & Sons, Inc., New York, 2000.

Bliže je vrednost R² vrednosti 1, bolj se krivulja prilega točkam in enačba odgovarja podatkom (slika 16.31).



Slika 16.31: Prikaz linearne, prilegajoče krivulje k eksperimentalnim podatkom.

Enačba, ki podaja odvisnost $q_v(R_o)$ je torej: $q_v = 1,636 R_o + 4,4$ in jo lahko uporabljamo v podanem merilnem območju.

Uporaba funkcij SLOPE, INTERCEPT in RSQ

Za izračun naklona premice in odseka na y osi ter kvadrata korelacijskega koeficienta R lahko uporabimo tudi naslednje funkcije:

SLOPE – funkcija izračuna naklon regresijske premice.

Primer: V poljubno celico zapišemo:

=SLOPE(B2:B6;A2:A6), kjer prvi niz podatkov predstavlja podatke odvisne spremenljivke in drugi niz podatke neodvisne spremenljivke.

- INTERCEPT funkcija izračuna točko v kateri seka premica y os. **Primer:** V poljubno celico zapišemo: =INTERCEPT(B2:B6;A2:A6)
- RSQ funkcija izračuna kvadrat korelacijskega koeficienta R, tj. R². **Primer:** V poljubno celico zapišemo: =RSQ(B2:B6;A2:A6)

L		B15	-	f_{x}	fx =SLOPE(B2:B6;A2:A6)				
5		А	В	С	D		E	F	
7	13								
	14	Uporaba f	unkcij SLO	PE, INTERC					
7	15	naklon	1,636						
б. •	16	odsek	4,4						
	17	R ²	0,9971						
-	18								

Slika 16.32: Uporaba funkcij SLOPE, INTERCEPT in RSQ.

Problemi:

16.1 V 13. poglavju smo se naučili vnašati podatke v program Excel. Sedaj narišite različne grafikone iz podatkov sestave mineralne vode Donat Mg!

Sestava ioni	Koncentracija c/(mg/L)
Mg ²⁺	1030
Na⁺	1500
Ca ²⁺	180
HCO ₃ ⁻	7700
SO4 ²⁻	2400
Cl	59
CO_2	3500

Ugotovite, kateri tip grafikona je najprimernejši za uporabo v tem primeru!

16.2 Narišite graf, ki bo podajal odvisnost parnega tlaka vode in etanola od temperature v območju od t = 0 °C do t = 100 °C. Eksperimentalni podatki so naslednji⁶:

t/°C	p ^{nas} /	/kPa
	voda	etanol
0	0,610	1,63
10	1,23	3,15
20	2,34	5,85
30	4,24	10,5
40	7,38	18,1
50	12,35	29,6
60	19,92	47,0
70	31,16	72,3
80	47,35	108,3
90	70,10	
100	101,32	

Pri vnosu podatkov za temperaturo uporabite opcijo **Fill**, ki smo jo spoznali v primeru 15.4.

16.3 Pri t = 40 °C razpada H₂O₂ z določeno hitrostjo. Sledenje hitrosti reakcije med razpadanjem je razvidno iz naslednjih podatkov:

<i>t</i> /h	0	6	12	18	24
c (H ₂ O)/(mol/L)	1	0,5	0,25	0,125	0,0625

Narišite krivuljo reakcijske hitrosti, ki bo prikazovala zvezo med koncentracijo H_2O_2 in časom razpadanja.

⁶ Božič Sodja J., Fizikalna kemija, Učbenik za predmet Fizikalna kemija, Tehniška založba Slovenije, Ljubljana, 2005.

16.4 Določite aktivacijsko energijo reakcije, kjer se ciklopropan presnuje v propen. Določitev izvedite po Arrheniusu.

Logaritmirana oblika enačbe je:

$$\ln k = \ln k_0 - \frac{E_a}{R} \cdot \frac{1}{T}$$

Izračunajte:

- nove koordinate $X\left(\frac{1}{T}\right)$ in Y (ln k), ki dajo novo funkcijo (premico) z naklonom $-\frac{E_a}{R}$,
- na osnovi izračunanih koordinat z Excelom narišite funkcijo Y = f(X),
- določite linearno trendno črto (linearna regresija),
- iz izračunanega naklona premice določite E_a .

Podatki so naslednji:

<i>t</i> /⁰C	<i>k/</i> s ⁻¹
750	1,8 × 10 ⁻⁴
800	$2,7 \times 10^{-3}$
850	$3,0 \times 10^{-2}$
900	0,26

Vprašanja za preverjanje znanja:

- 1. Kaj je grafikon?
- 2. Zakaj je primerneje prikazati rezultate z grafikonom?
- 3. Zakaj rečemo, da so grafikoni "dinamični"?
- 4. Kateri grafikon je najenostavnejši in zakaj?
- 5. Kako naj bo grafikon opremljen?
- 6. Kateri so osnovni tipi grafikonov?
- 7. Kdaj je primerna uporaba stolpčnega grafikona?
- 8. Kdaj je primerna uporaba paličnega grafikona?
- 9. Kdaj je primerna uporaba črtnega grafikona?
- 10. V katerih primerih uporabljamo tortni grafikon?
- 11. Kdaj je primerna uporaba ploščinskega grafikona?
- 12. Kaj je trendna črta?
- 13. Kaj definirajo funkcije SLOPE, INTERCEPT in RSQ ?

17. POGOJNI STAVKI V MS EXCELU

Pogojni stavki IF se uporabljajo tudi v Excelu. Ker smo njihovo delovanje in uporabo spoznali že v prvem delu predmeta pri programiranju v Fortranu, bomo lažje razumeli njihovo delovanje v Excelu.

17.1 Uporaba pogojnega stavka IF v Excelu

Pogojni stavek zapišemo z besedo IF. V vsakdanjem življenju velikokrat uporabljamo pogojne stavke npr.: »Če se bom učil, bom dobil lepo oceno.« Uporabljen je logični pogoj, ki mu sledi posledica: če (IF) bom naredil to, potem (THEN) se bo zgodilo to in to. V pogojnih stavkih uporabljamo naslednje primerjalne operatorje:

Primerjalni operator	Pomen	Primer
= (enačaj)	Enako	A1=A2
> (znak večji od)	Večji od	A1>A2
< (znak manjši od)	Manjši od	A1 <a2< td=""></a2<>
>= (znak večji od in enačaj)	Večji kot ali enak	A1>=A2
<= (znak manjši od in enačaj)	Manjši kot ali enak	A1<=A2
<> (neenačaj)	Ni enako	A1<>A2

Splošni zapis pogojnega stavka IF v Excelu:

IF (logični pogoj; zapis za TRUE; zapis za FALSE)

Z logičnim pogojem preverimo nek izraz ali vrednost v določeni celici.

Zapis za TRUE

Na tem mestu vnesemo izraz, ki pove, kaj naj se izvrši, če je pogoj izpolnjen (TRUE). To je lahko poljuben tekst ali enačba.

Zapis za FALSE

Na tem mestu vnesemo izraz, ki se izvrši, če pogoj ni izpolnjen (FALSE).

Poglejmo uporabo na primerih.

Primer 17.1: Uporaba stavka IF oziroma funkcije IF.

V celico D16 bomo zapisali število 10. V celico D17 bomo vnesli funkcijo IF, s katero bomo preverjali, če je vrednost v celici D16 večja ali manjša od 8. Če je večja, bo izpolnjen nek pogoj, če je manjša pa drugi pogoj. Spomnite se v Fortranu delovanja **IF THEN ELSE** strukture.

Z logičnim pogojem preverimo vrednost v celici D16. V našem primeru preverimo, če je vrednost v celici D16 večja ali manjša od 8. V Excelu uporabimo simbol > za večje kot.

Zapis za TRUE

Na tem mestu vnesemo izraz, ki pove, kaj naj se izvrši, če je pogoj izpolnjen (TRUE), v našem primeru **Večje od 8.**

Zapis za FALSE

Na tem mestu vnesemo izraz, ki se izvrši, če pogoj ni izpolnjen (FALSE), v našem primeru **Manjše od 8.**

Primer prikazuje slika 17.1.

	D17	- (0	f_{x}	=IF(D16>8;"Večje	e od 8.";"M	anjše od 8.")
	А	В	С	D	E	F
15						
16				10		
17				Večje od 8.	t.	
18					Rezultat	1
19					Število 10	je večje
20					od 8.	
21						

Slika 17.1: Prikaz zapisa pogojnega stavka in dobljen rezultat.

Slika prikazuje zapis funkcije IF, ki je naslednji: **=IF(D16>8;"Večje od 8.";"Manjše od 8.")**. Zapis se prične z **=**, nato zapišemo besedo **IF**. Posamezne dele v oklepaju ločimo s podpičjem, tekst, ki se izvrši, če je pogoj TRUE ali FALSE pa zapišemo med narekovaji ". Ker je število 10 večje od 8 je rezultat **Večje od 8**.

V naslednjem primeru zapišimo in izvedimo z Excelom primer 8.4 (stran 47), ki smo ga programirali v Fortranu.

Primer 17.2: Uporaba stavka IF v nizu podatkov.

Glede na doseženo oceno pri izpitu, razvrstite študente na tiste, ki so dosegli pozitivno oceno (60 ali več), ter tiste z negativno oceno (manj kot 60). Pri študentu, ki je opravil izpit, se naj izpiše tekst **Opravil. Dober študent.**, pri študentu, ki ni opravil izpita, pa se naj izpiše **Ni opravil.**

	F1	- ()	<i>f</i> _≪ =IF(E1>=60;"Oprav	il. Dober š	tudent.";"I	Ni opravil.")	
	D	E	F	G	Н	I.	
1	Ana	87	Opravil. Dober študent.				
2	lvo	55					
3	Jana	75					
4	Maja	60					
5	Luka	45					

Slika 17.2: Zapis IF funkcije ter rezultat za prvega študenta.

Slika 17.2 prikazuje imena študentov (D1-D5) ter vhodne podatke, tj. dosežene izpitne ocene teh študentov (E1 - E5). V celici F1 je zapis funkcije IF:

=IF(E1>=60;"Opravil. Dober študent.";"Ni opravil.")

Po zagonu se izpiše rezultat **Opravil. Dober študent.**, saj je študent dosegel oceno 87, ki je višja od 60.

Postopek ponovimo še za ostale ocene. V aktivni celici F1 desnem robu spodaj aktiviramo kvadratek in ga povlečemo navzdol do celice F5. Rezultat prikazuje slika 17.3.

	D	E	F
1	Ana	87	Opravil. Dober študent.
2	lvo	55	Ni opravil.
3	Jana	75	Opravil. Dober študent.
4	Maja	60	Opravil. Dober študent.
5	Luka	45	Ni opravil.

Slika 17.3: Izpitni rezultati opremljeni s tekstom.

Iz rezultatov razberemo, da so trije študenti izpolnili pogoj TRUE, tj. opravili izpit, dva študenta pa nista opravila izpita, torej sta izpolnila pogoj FALSE.

17.1.1 Barvni efekti v pogojnih stavkih

Sliki 17.4 in 17.5 prikazujeta, kako lahko rezultat opremimo z barvami in pri tem uporabimo pogojni stavek IF. V našem primeru bomo barvno označili pozitivne ocene. Postopek je naslednji:

- označimo stolpec z vhodnimi podatki (od E1 do E5),
- kliknemo jeziček Home,
- kliknemo gumb Conditional Formating,
- izberemo New Rule.

Odpre se okno prikazano na sliki 17.4.

ew Formatti	ng Rule	? ×
<u>S</u> elect a Rule T	'ype:	
► Format all	cells based on their values	
Format on	ly cells that contain	
Format on	ly top or bottom ranked values	
Format on	ly values that are above or below average	
Format on	ly unique or duplicate values	
► Use a form	nula to determine which cells to format	
Edit the Rule D	escription: y cells with: greater than or equal to 60	
Preview:	No Format Set <u>F</u> ormat	
, 	ОК	Cancel

Slika 17.4: Podajanje rezultata v barvah.

V oknu **New Formatting Rule** izberemo:

- Format only cells that contain,
- v prvem padajočem meniju Cell Value,
- v drugem padajočem meniju greater than or equal to,
- v zadnje okence zapišemo vrednost 60.

Ker želimo rezultat primerno obarvati, kliknemo gumb **Format**. Odpre se okno **Format Cells**. Izberemo **Fill** in želeno barvo (slika 17.5). Rezultat je prikazan na sliki 17.6.

Format Cells	? ×
Number Font Border Fill	
Background Golor:	Pattern Color: Automatic Pattern Style:
	Clear_

Slika 17.5: Izbira barve za prikaz pozitivnih ocen.

	D	E	F
1	Ana	87	Opravil. Dober študent.
2	lvo	55	Ni opravil.
3	Jana	75	Opravil. Dober študent.
4	Maja	60	Opravil. Dober študent.
5	Luka	45	Ni opravil.

Slika 17.6: Barvni prikaz pozitivnih ocen.

17.1.2 Uporaba funkcije COUNTIF

Uporabili bomo še funkcijo COUNTIF in z njo prešteli število študentov s pozitivno oceno. Splošni zapis uporabe funkcije COUNTIF je:

COUNTIF(območje;pogoj)

Postopek je naslednji:

- kliknemo celico E7,
- zapišemo formulo za klic funkcije: = COUNTIF(E1:E5;">=60")
- kliknemo tipko ENTER,
- pojavi se rezultat 3.

Rezultat prikazuje slika 17.7.

	E7	• (•	<i>f</i> _≪ =COUNTIF(E1:E5;">=60")	
	D	E	F G	
1	Ana	87	Opravil. Dober študent.	
2	Ivo	55	Ni opravil.	
3	Jana	75	Opravil. Dober študent.	
4	Maja	60	Opravil. Dober študent.	
5	Luka	45	Ni opravil.	
6				
7	Število pozitivnih:	3		
8			Funkcija COUNTIF	
9			prešteje pozitivne	
10			ocene.	

Slika 17.7: Uporaba funkcije COUNTIF.

17.1.3 Enačbe v pogojnem stavku

V pogojnem stavku lahko pri zapisu TRUE in FALSE vnesemo tudi enačbo oziroma formulo, ki jo uporabimo za reševanje problema, če je pogoj izpolnjen oziroma ni izpolnjen. Uporabo prikazuje primer 17.3.

Primer 17.3: Izračun osnovne cene toplotnih prenosnikov. Osnovno ceno toplotnih prenosnikov (TP) lahko izračunamo po formuli⁷:

 $C_0 = 313 \cdot A^{0,08}$, ko je površina TP manjša od 10 m²,

 $C_0 = 475 \cdot A^{0.65}$, ko je površina TP večja od 10 m².

Izračunajte osnovno ceno naslednjih TP, če so površine:

A/m ² 2 4 6 8 10 50 100 150 200 250 300 350 400 450
--

Rešitev:

Slika 17.8 prikazuje rešitev primera. V stolpcu od A7 do A20 so podane površine TP. Celica B7 prikazuje klic IF funkcije, ki je naslednji:

=IF(A7<=10;\$E\$7*A7^\$E\$8;\$E\$11*A7^\$E\$12)

Opomba: Enačb za računanje ne pišemo med narekovaji "!

Ko dobimo prvi rezultat dobimo ostale tako, da aktiviramo kvadratek desno spodaj v aktivni celici in ga povlečemo navzdol do celice B20. Za bolj zanimiv in pregleden prikaz rezultatov so površine TP, ki so manjše od 10 m², obarvane. Uporabljen je ustrezen pogoj. Kako ga uporabimo? Obnovite postopek podpoglavja 17.1.1.

⁷ Guthrie K. M., Capital Cost Estimating, Chem. Eng., 76 (6) 1969.

	B7	- (f_x	=IF(A7<:	=10;\$E\$7	/*A7^\$E\$8;	\$E\$11*	A7^\$E	\$12)
	А	В	С	D	E	F	G	Н	
1									
2	Izračun os	novne cene TP po	o Guthrie	eju:					
3	Formula z	a izračun osnovne	e cene TF	P, če je A	<10 m²: (C _o = 313·A ⁰	,08		
4	Formula z	a izračun osnovne	e cene TR	P, če je A	>10 m²: (C _o = 475·A ⁰	,65		
5									
6	A (m²)	C ₀ (USD)		Vhodr	ni podat	ki:			
7	2	331		a=	313	za povr	šine ma	injše	
8	4	350		b=	0,08	od 10 m	12.		
9	6	361							
10	8	370							
11	10	376		a=	475	za pov	ršine ve	ečje	
12	50	6040		b=	0,65	00 101	112.		
13	100	9477							
14	150	12335							
15	200	14872		Splošr	na formu	ıla za izrač	un pov	ršine T	P:
16	250	17193							
17	300	19356		C _o = a	Ap				
18	350	21396							
19	400	23336							
20	450	25193							

Slika 17.8: Izračun osnovne cene TP.

17.1.4 Uporaba logičnih operatorjev AND, OR in NOT

Spomnimo se programiranja v Fortranu. Pogoj lahko predstavlja logični izraz ali še en pogoj, torej je lahko sestavljen iz več pogojev. V takih primerih uporabljamo logične operatorje AND, OR in NOT. Enaka logika se uporablja tudi pri uporabi v Excelu. Poglejmo primer.

Primer 17.4: Uporaba pogojnega stavka s TRUE in FALSE.

V stolpec vnesimo podatke: **35, 10, 30, vroče, hladno**. Nato jih primerjajmo med seboj. Pri izpolnjenem pogoju se bo izpisala beseda TRUE, če pogoj ne bo izpolnjen pa beseda FALSE. Kaj se bo izpisalo kot rezultat v primerih podanega pogoja :

- a) 35 večje od 10 in manjše od 30? Uporabite primerjalni operator AND !
- b) 35 večje od 10 in manjše od 30? Uporabite primerjalni operator OR !
- c) 35 plus 10 je enako 45? Uporabite primerjalni operator NOT !
- d) A5 je enako vroče ? Uporabite primerjalni operator NOT !
- e) A5 ni enako vroče ali A6 je enako hladno? Uporabite primerjalni operator OR !

Rezultat prikazuje slika 17.9.

	A8	•	(•	f_{x}	=AND)(A2>A3;A2	2 <a4)< th=""></a4)<>
	А	В	С		D	E	F
1	Podatki						
2	35						
3	10						
4	30						
5	vroče						
6	hladno						
7							
8	FALSE						
9	TRUE	Upor	raba operat	orja			
10	FALSE	AND.					
11	FALSE						
12	TRUE						

Slika 17.9: Uporaba operatorjev AND, OR in NOT.

Zapis pogojnih stavkov je naslednji:

a)	=AND(A2>A3;A2 <a4)< th=""><th>Rezultat: FALSE (oba pogoja morata biti izpolnjena)</th></a4)<>	Rezultat: FALSE (oba pogoja morata biti izpolnjena)
b)	=OR(A2>A3;A2 <a4)< td=""><td>Rezultat: TRUE (en pogoj mora biti izpolnjen)</td></a4)<>	Rezultat: TRUE (en pogoj mora biti izpolnjen)
c)	=NOT(A2+A3=45)	Rezultat: FALSE (izpolnjeno mora biti nasprotje podanega pogoja)
d)	=NOT(A5="vroče")	Rezultat: FALSE (izpolnjeno mora biti nasprotje podanega pogoja)
e)	=OR(A5<>"vroče";A6="hladno")	Rezultat: TRUE (en pogoj mora biti izpolnjen)

Primer 17.4 lahko rešimo tudi na drug način (slika 17.10) in sicer tako, da pogojni stavek pričnemo vedno z besedo IF, nato pa v oklepaju uporabimo ustrezni primerjalni operator. Vhodni podatki so v stolpcu G2 do G6.

Kaj se bo izpisalo kot rezultat, če bo pogojni stavek zapisan kot:

a) =IF(G2=35;"Velja.";"Ne velja.")	Rezultat: Velja.
b) =IF(G2<>35;"Res je.";"Ni res.")	Rezultat: Ni res.
c) =IF(NOT(G2<=35;"Tako je.";"Ni tako.")	Rezultat: Ni tako.
d) =IF(G5<>"Vroče.";"V redu.";"Ni v redu.")	Rezultat: Ni v redu.
e) =IF(AND(G2>G3;G2 <g4);"velja.";"ne td="" velja.")<=""><td>Rezultat: Ne velja.</td></g4);"velja.";"ne>	Rezultat: Ne velja.
f) =IF(AND(G2<>G3;G2<>G4);"Tako je.";"Ni tako.")	Rezultat: Tako je.
g) =IF(OR(G2>G3;G2 <g4);"drži.";"ne drži.")<="" td=""><td>Rezultat: Drži.</td></g4);"drži.";"ne>	Rezultat: Drži.

Rezultat prikazuje slika 17.10.

	G13	•		<i>f</i> _∞ =IF(A	ND(G2>G3	;G2 <g4);"v< th=""><th>elja.";"Ne</th><th>velja.")</th></g4);"v<>	elja.";"Ne	velja.")
	E	F	G	Н	1	J	K	L
1			Podatki					
2			35					
3			10					
4			30					
5			vroče					
6			hladno					
7								
8		Razultati	uporabe Al	ND, OR in M	IOT:			
9			Velja.					
10			Ni res.		Uporaba			
11			Ni tako.		operatorja	AND.		
12			Ni v redu.					
13			Ne velja.					
14			Tako je.					
15			Drži.					

Slika 17.10: Uporaba operatorjev AND, OR in NOT.

Problemi:

- **17.1** Rešite primer 8.1 iz 8. poglavja z Excelom in uporabite ustrezne primerjalne operatorje!
- **17.2** Rešite primer 8.2 iz 8. poglavja z Excelom in uporabite logične operatorje AND, OR in NOT!
- 17.3 Rešite problem 8.3 z Excelom!
- 17.4 Rešite problem 8.4 z Excelom!
- 17.5 Rešite problem 8.5 z Excelom!

Vprašanja za preverjanje znanja:

- 1. Kako zapišemo različne primerjalne operatorje v pogojnem stavku IF, če uporabimo Excel?
- 2. S čim ločimo logični pogoj, zapis za TRUE in zapis za FALSE v pogojnem stavku IF?
- 3. Kdaj uporabimo funkcijo COUNTIF? Kako se glasi klic te funkcije?
- 4. Ali enačbe, s katerimi bomo reševali problem, napišemo, v pogojnem stavku, med narekovaji?
- 5. Kako deluje pogojni stavek IF, ko uporabimo:
 - logični operator AND,
 - logični operator OR,
 - logični operator NOT?
 - Zapišite primere!

18. REŠEVANJE ALGEBRSKIH ENAČB Z ENO NEZNANKO

Kadar želimo najti rešitev nelinearne algebrske enačbe z eno neznanko, moramo uporabiti primerno numerično metodo. Obstajajo različni matematični paketi (Polymath, Mathcad) z numeričnimi metodami, s katerimi lahko rešimo takšne probleme. V Excelu lahko uporabimo orodji Goal Seek in Solver.

18.1 Graf funkcije

Kadar ne vemo točno, kje so rešitve enačbe, narišemo najprej graf, iz katerega odčitamo približne rešitve. Funkcijo pripravimo v obliko f(x) = 0. Nato izračunamo vrednosti f(x) pri izbranih vrednostih x. Ves postopek izvedemo s programom Excel.

Primer 18.1: Kreiranje grafa funkcije. Z Excelom poiščimo rešitev naslednje enačbe:

 $x^3 + 2x^2 - x + 1 = 0$

Slika 18.1 prikazuje vnos vrednosti za x v celice od A59 navzdol (x se spreminja od vrednosti -3 do 2 po 0,2). Prikazana je aktivna celica B59 ter zapis izraza funkcije f(x) za to celico. Ostale vrednosti f(x) dobimo tako, da povlečemo desni spodnji kvadrat na aktivni celici navzdol po koloni B.

	B59	-	0	$f_{\mathcal{K}}$	=A59	^3 + 2*A59	^2 - A59 + 1
	А	В	С		D	E	F
54							
55	y=x^3 + 2x	^2 - x + 1					
56							
57							
58	х	γ					
59	-3	-5					
60	-2,8	-2,472	-				
61	-2,6	-0,456					
62	-2,4	1,096					
63	-2,2	2,232					
64	-2	3					
65	-1,8	3,448					
66	-1,6	3,624					
67	-1,4	3,576					
68	-1,2	3,352					
69	-1	3					
70	-0,8	2,568					
71	-0,6	2,104					
72	-0,4	1,656					
73	-0,2	1,272					
74	0	1					
75	0,2	0,888					
76	0,4	0,984					
77	0,6	1,336					
78	0,8	1,992					
79	1	3					
80	1,2	4,408					
14 4	→ > She	et1 Sheet	2 / Sheet3	1		1	

Slika 18.1: Prikaz vnosa podatkov v celice.

Po izračunanih koordinatah x in y, narišemo graf, kot ga prikazuje slika 18.2 z eno od rešitev funkcije, tj. pri x \approx – 2,5. Graf je narisan z **Scater** grafom.



Slika 18.2: Graf funkcije z rešitvijo.

18.2 Orodje Goal Seek

Goal Seek je enostavno orodje, ki je že vgrajeno v Excelu in ga ni težko uporabljati. Poglejmo njegovo uporabo na prejšnjem primeru.

Najprej pripravimo enačbo v primerno obliko tako, da bo veljalo f(x) = 0. Če želimo rešiti enačbo z orodjem Goal Seek, moramo definirati celico za spremenljivko, ki jo želimo spreminjati in celico za funkcijo, katere rešitev iščemo (slika 18.3).

C	2	Home	Ins	ert Pa	ge Layou	ut Fo	rmulas	Data	Review	١	√iew	Acrob	at									
*			*				6	Disconnection Connection	ctions ties	↠[AZA	Y	🖗 Clear	bly	÷		ő			2	•	
Fro	om ess	From Web	From Text	From Othe Sources	r Ex Coni	isting nections	Refresh All *	🖙 Edit Lin	nks	Z↓	Sort	Filter	V Advan	iced	Text to Columns	Remove Duplicates	Data Validation	Consolidate	Wha Analy	at-If ∕sis ▼	Group	Ungroup S
	Get External Data				C	Connections So			Sort & Filter Data Tools			ls		<u>S</u> cen	ario Mana	ager						
		B2		- (•	f.	⊊ =A2^	3 + 2*A2	^2 - A2 + 1	L											<u>G</u> oal	Seek	
		Α	В	(:	D	E	F		G		н	1		J	К	L	М		Data	<u>T</u> able	
1	Х	1	f(X)																_			
2		1		3																		
2																						

Slika 18.3: Definiranje celic in izbira orodja Goal Seek.

V našem primeru smo v celico A1 vnesli ime neodvisne spremenljivke x, v celico B1 pa ime funkcije f(x). V celico A2 vnesemo ocenjeno vrednost za x, v celico B2 pa zapišemo funkcijski izraz. V našem primeru se po zapisu enačbe prikaže rezultat za f(x) in je enak vrednosti 3, kar je res, če je x=1. Vendar to ni rešitev, saj mora veljati f(x)=0. Če kliknemo zavihek **DATA** in **What if Analysis?**, se nam odpre seznam, na katerem izberemo orodje Goal Seek (slika 18.3), s katerim poiščemo rešitev. Po izbiri orodja Goal Seek se odpre okno, v katerem definiramo:

- ciljno celico B2 v kateri bo rezultat (Set cell),
- želen rezultat, 0, (To value) in
- celico A2 v kateri se bo spreminjala vrednost x (By changing cell).

Vnešene podatke prikazuje slika 18.4.

	B2	-		<i>f</i> _x =A2^	3 + 2*A2^2	- A2 + 1				
	А	В	С	D	E	F	G	Н	1	J
1	х	f(X)					Goal Seek		?	XI
2	1	3						_		_
3							S <u>e</u> t cell:	B2		<u>.</u>
4							To <u>v</u> alue:	0		
5							By <u>c</u> hangin	ig cell: A2		S
6								OK I	Canad	
7								UK	Cancel	
2										

Slika 18.4: Definiranje ciljne celice želenega rezultata in celice z neodvisno spremenljivko.

Nato kliknemo **OK**. Program prične iskati rešitev in izvaja iteracije. Ko je rešitev dosežena, se v oknu A2 izpiše vrednost neodvisne spremenljivke x, v oknu B2 pa vrednost funkcije f(x). V našem primeru je rešitev x = -2,54682. Hkrati se pojavi okno orodja Goal Seek s sporočilom o doseženi rešitvi (slika 18.5).

	B2	•	0	<i>f</i> _x =A2^	3 + 2*A2^2	- A2 + 1			
	А	В	С	D	E	F	G	Н	I.
1	х	f(X)				Goal Se	ek Status		? ×
2	-2,54682	-7,7E-06				Goal Se	ekina with Cel	B2	Ghan
3						found a	solution.	_	breh
4						Target	value: 0		Pause
5						Current	value: -7,68	- 8498E-06	
6							· · · · · ·	OF 1	Capital
7							<u>.</u>		Cancel
Q									

Slika 18.5: Rešitev enačbe z Goal Seek orodjem.

18.3 Orodje Solver

Navodilo za instalacijo

Solver je program za MS Excel, ki ga uporabljamo za iterativne izračune, optimizacijo in simulacijo tehničnih in ekonomskih problemov. Če program Solver uporabljate prvič, ga morate aktivirati. Postopek je naslednji.

- v osnovnem Excelovem oknu kliknite na Office gumb skrajno levo zgoraj,
- v oknu, ki se odpre, kliknite Excel Options,
- v tem oknu izberite Add-ins kategorijo,
- v seznamu Manage izberite Excel Add-ins in kliknite Go,
- odpre se okno z raznimi programi v katerem izberete program Solver Add-in in kliknete OK.

Če odprete zavihek **Data**, boste v skupini **Analysis** našli program **Solver**. Uporabo orodja Solver prikazuje naslednji primer.

Primer 18.2: Uporaba Solverja.

Z orodjem Solver rešimo kubično enačbo iz prejšnjega poglavja. Najprej vnesemo ime neodvisne spremenljivke x (v našem primeru v celico A1) in ime funkcije f(x), katere rešitev iščemo (v našem primeru v celico B1). V celici A2 je ocenjena vrednost za x, v celici B2 je zapisan funkcijski izraz. Nato kliknemo na ikono za Solver. Odpre se okno (slika 18.6) v katerem definiramo:

- ciljno celico B2 (Set target cell),
- želen rezultat, 0, (**Equal to** \rightarrow **Value of**) in
- celico A2 v kateri se bo spreminjala vrednost x (By changing cell). Nato kliknemo na tipko Solve.

	B2	- (0	f_x	=A2^3 + 2	*A2^2 - A2	+1				
	А	В	С	D	E	F	G	Н	1	J
1	х	f(X)								
2	-2,5	<mark>0,37</mark> 5								
3										
4										
5			Solve	r Paramete	rs					×
6			Set T	arget Cell:	\$B\$2	1			Solve	
7			Equa	To: O	мах Ом	in 💽 Val	ue of: 0			
8			-Ву С	hanging Cells	:	<u>11</u> - <u>1</u> 0	40 on je		Close	
9			\$44	12				Guess		
10				-			_	<u>a</u> aoso		
11			-5 <u>u</u> b)	ect to the Co	nstraints:			,	Option	is
12			_					<u>A</u> dd		
13			_					Change		
14									<u>R</u> eset /	41
15							—	Delete	Help	
16									<u>_</u>	
17										

Slika 18.6: Osnovni podatki v Solverju.

Velikokrat z ocenjeno vrednostjo ne dobimo rešitve. V tem primeru lahko z opcijo **Options** spremenimo določene parametre kot so: čas izračunavanja iteracij, število iteracij, konvergenčni kriterij ipd. (slika 18.7), druga začetno vrednost.

	B2	• (0	<i>f</i> _x =A2^3	+ 2*A2^2 -	A2+1		
1	testni primer	SOLVER.xlsx						
	А	В	С	D	E	F	G	Н
1	х	f(X)	5	luor Options				
2	-2,5	0,375		nver options				
3			P	4ax Time:	100 :	seconds	C	ж
4			ī	terations:	100	1	Cal	ncel
5								
6			F	Precision:	0,000001	1	Load M	1odel
7			1	olerance:	5	%	<u>S</u> ave M	1odel
8				onvergence:	0.0001		He	eln l
10					1.,			
11				Assume Line	ear <u>M</u> odel	<u>U</u> se	e Automatic S	caling
12				Assume Nor	n-Negative	💌 Sha	w Iteration <u>P</u>	lesults
13				C Tangent	C	Forward	Newl	ton
14				Ouadratic		Central	C Coni	ugate
15			L	7000.000		<u>T</u>		
16								

Slika 18.7: Nastavitev parametrov.

Pomen opcij:

Precision – natančnost napake (omejitve). Manjše je to število, večja je natančnost. Specificirana je kot decimalno število med 0 in 1. Privzeta vrednost je $1,0 \times 10^{-6}$.

Max Time – maksimalni čas za dosego rezultata, ki ga lahko spreminjamo. Standardna omejitev je 100 s, in je primerna za manjše probleme. Maksimalni čas je 32 767 s (\approx 9 h).
Iterations – število iteracij predvidenih za dosego rezultata. 100 iteracij je standardna omejitev, število lahko spreminjate.

Convergence – V vsaki iteraciji Solver primerja nove vrednosti s prejšnjimi. Če je sprememba (razlika) manjša od podanega konvergenčnega kriterija, je rešitev dosežena. Če zmanjšamo vrednost konvergenčnega kriterija, je za dosego rešitve potreben daljši čas.

Tolerance – Uporablja se za določitev celoštevilčnih pogojev. Privzeta vrednost je 5 %.

Assume Linear Model – Če rešujemo linearni model enačb, lahko uporabimo Simplex metodo namesto gradientne metode. Če izberemo to opcijo, Solver preveri, ali je model linearen glede na kriterije.

Assume Non-Negative – to opcijo izberemo, če želimo, da so vse spremenljivke večje od nič.

Show Iteration Results – opcijo izberemo, če želimo zasledovati iteracije. V tem primeru se Solver večkrat ustavi. Trenutni rezultati se pojavijo na aktivnem delovnem listu. Opcija je primerna za uporabo, če želimo vpogled v vsako iteracijo.

Estimates – Z opcijo specificiramo oceno začetne vrednosti neodvisne spremenljivke. S tangentno metodo se izvede linearna ekstrapolacija, Quadratic uporablja kvadratno ekstrapolacijo, ki pospeši konvergenco nelinearnih problemov.

Derivatives – Solver uporablja končne diference pri izračunu gradientov. Izberete lahko med Forward in Central Metodo. Central metoda je bolj natančna.

Search – Pri tej opciji lahko uporabimo Newtonovo in Conjugate gradientno metodo. Uporabljata se za iskanje smeri med vsako iteracijo.

Podrobna navodila za uporabo vsake opcije najdete, če kliknete na gumb Help.

Po definiranju opcij prične program z izračunavanjem iteracij. Ko je rešitev dosežena, se pojavi okno (slika 18.8), v katerem izberete opcijo **Answer** za prikaz rezultatov, ki jih nato najdete v delovnem listu.

		-	(•	<i>f</i> _∞ =A2^	3 + 2*A2^2	- A2 + 1								
4	А	В	С	D	E	F	G	Н						
1	Х	f(X)												
2	-2,54682	4,06E-07												
3		Colu	av Doculto											
4		SUIV	er Results						5					
5		Solv	Solver found a solution. All constraints and optimality											
6		con	uluuris are sa	usheu.			Reports							
7) Keep Solver	r Solution			Sensitivity							
8			[•] <u>R</u> ectore Ori	gipal Values			Limits	_						
9			Rescore On	ginar values			1							
10			ОК	Cance	el	ave Scenario		Help						
11														



Slika 18.9 prikazuje izpis vhodnih podatkov in rezultata kubične enačbe.

	A1		- (•)	🕼 Microsoft	Excel 12.0	Answer Re	port							
	A B	С	D	E	F	G								
1	Microsoft Excel 12.0 Answer Report													
2	Worksheet: [testni primer SOLVER.xlsx]Sheet1													
3	Report Created: 3.11.2010 14:13:20													
4														
5	1													
6	Target Ce	ell (Valu	e Of)											
7	Cell	Name	Original Value	Final Value										
8	\$B\$2	f(X)	0,375	4,06333E-07										
9														
10]													
11	Adjustab	le Cells												
12	Cell	Name	Original Value	Final Value										
13	\$A\$2	Х	-2,5	-2,546818228										
14														
15														
16	Constrair	nts												
17	NONE													

Slika 18.9: Prikaz rezultatov.

Problemi:

18.1 Poiščite rešitve naslednje enačbe:

f(x)=1 + x [3 + x (3 x - 7)] = 0

Narišite graf funkcije, v katerem odčitajte rešitve. Nato poiščite rešitve še z orodjema: a) Goal seek in

b) Solver.

Rešite poiščite pri različnih ocenjenih vrednostih x. V prvem primeru naj bo ocenjena vrednost x = 0.5 v drugem primeru pa x = 1.5.

(**Rešitev:**x = 1,549)

18.2 Poiščite rešitev naslednje nelinearne enačbe:

 $f(x) = x^2 - \sin x = 0$

Narišite graf funkcije, v katerem odčitajte rešitev. Nato poiščite rešitev še z orodjema: a) Goal seek in

b) Solver.

Ocenjena vrednost naj bo x = 0.6.

(**Rešitev**: x = 0,87673)

18.3 Poiščite rešitev naslednje nelinearne enačbe⁴:

 $x = e^{-x}$

Za reševanje uporabite orodje Solver. Ocenjena vrednost x = 0.9.

(**Rešitev**: x = 0,56714)

Vprašanja za preverjanje znanja:

- 1. Katera orodja za reševanje algebrskih nelinearnih enačb ima na voljo Excel?
- 2. V kakšno obliko pripravimo funkcijo, ki jo rešujemo s temi orodji?
- 3. Ali dobimo pravilen rezultat že z ocenjeno vrednostjo neodvisne spremenljivke?
- 4. Opišite princip delovanja orodji Goal Seek in Solver!

⁴ R. M. Felder, R. W. Rousseau, Elementary Principles of chemical Processes, Third Edition, John Wiley & Sons, Inc., 2000, str. 612.

19. REŠEVANJE SISTEMOV LINEARNIH ENAČB

Omejili se bomo na sisteme enačb, ki imajo število enačb enako številu neznank in jih lahko rešujemo z matrično algebro. V programu Excel lahko izberemo različna orodja. V prvem primeru bo prikazan postopek z inverzno matriko, v drugem pa uporaba orodja Solver.

19.1 Reševanje z inverzno matriko⁸

Kot smo že povedali, se bomo omejili na reševanje sistemov m linearnih enačb z m neznankami. Ti sistemi imajo natanko eno rešitev, če je matrika sistema regularna. Če je determinanta matrike različna od nič, je matrika regularna.

Kadar tvorimo sistem enačb v obliki matrične enačbe, lahko uporabimo princip z inverzno matriko. Pri reševanju so na voljo različne funkcije (MINVERSE, MMULT). Če smo zapisali sistem linearnih enačb v obliko:

$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$

lahko neznanke **x** izračunamo na naslednji način:

$\mathbf{x} = \mathbf{A}^{-1} \cdot \mathbf{b}$

Primer 19.1: Reševanje sistema linearnih enačb z inverzno matriko. Z matrično algebro in principom inverzne matrike rešite sistem treh linearnih enačb s tremi neznankami. Z izračunom vrednosti determinante preverite, če ima sistem rešitev! Sistem je naslednji:

9,375 x_1 + 3,042 x_2 - 2,437 x_3 = 9,233 3,042 x_1 + 6,183 x_2 + 1,216 x_3 = 8,205 -2,437 x_1 + 1,216 x_2 + 8,443 x_3 = 3,934

<u>Reševanje:</u>

Najprej tvorimo matriko **A** (celice od C6 do E8) (slika 19.1). Za izračun determinante uporabimo funkcijo MDETERM. Postavimo se v prazno celico (v našem primeru B25) in zapišemo enačbo za klic funkcije:

=MDETERM(C6:E8)

S klikom tipke F2 in zaporedja tipk CTRL+SHIFT+ENTER aktiviramo formula za izračun determinante. Kot rezultat se izpiše vrednost determinante (celica B25). Ker je vrednost determinante različna od nič, ima sistem enačb natanko eno rešitev.

Če želimo dobiti končni rezultat, tvorimo še matriko **b** (celice od C10 do C12) (slika 19.1). Prvi korak do rešitve je izraziti inverzno obliko matrike **A**. Pri izračunu uporabimo funkcijo MINVERSE. Označimo prazno polje (v našem primeru od C14 do E16), v katerega bodo zapisani koeficienti inverzne matrike. Nato zapišemo enačbo za klic funkcije:

⁸ Bourg D. M., Excel Scientific and Engineering Cookbook, O'Reilly, Beijing, 2006.

=MINVERSE(C6:E8)

Upoštevati moramo, da je vnesena matrika v območju definiranih celic. Nato s klikom tipke F2 in zaporedja tipk CTRL+SHIFT+ENTER aktiviramo formulo. Kot rezultat dobimo inverzno matriko (celice C14 do E16).

Naslednji korak je množenje inverzne matrike **A**⁻¹ z matriko **b**. Označimo prazno polje za rezultat (v našem primeru od C19 do C21) in zapišemo formulo za klic funkcije MMULT:

=MMULT(C14:E16;C10:C12)

Končni rezultat je prikazan v celicah od C19 do C21 (slika 19.1).

	B25	- (*	f _x	{=MDETER	M(C6:E8)}
	В	С	D	E	F
1					
2					
3					
4	David M. Bour	g, knjiga			
5					
6		9,375	3,042	-2,437	
7	[A]=	3,042	6,183	1,216	
8		-2,437	1,216	8,443	
9					
10		9,233			
11	[b]=	8,205			
12		3,934			
13					
14		0,148	-0,084	0,055	
15	Inv[A]	-0,084	0,214	-0,055	
16		0,055	-0,055	0,142	
17					
18					
19		0,896			
20	[x]=Inv[A][b]	0,765			
21		0,614			
22					
23					
24	vrednost dete	rminante:			
25	342,662				
26					
H I	→ > Sheet1	Sheet2 🖉	Sheet3 📈 🕈	7	

Slika 19.1: Prikaz vnosa podatkov in rezultat sistema treh linearnih enačb s tremi	
neznankami.	

19.2 Reševanje z orodjem Solver

19.2.1 Definiranje pogojev

Solver smo spoznali že v 18. poglavju. Uporabimo ga lahko tudi za reševanje sistemov linearnih enačb, ki imajo enako število enačb kot neznank. Uporaben je tudi takrat, ko zaradi velikega števila enačb težko rešujemo sisteme s postopkom inverzne matrike.

Slika 19.2 prikazuje podatke za prejšnji primer, ki ga bomo rešili z orodjem Solver. Enačbe sistema imajo obliko $a_1 x_1 + a_2 x_2 + a_3 x_3 = b$.

L	Ц		H4	•	0	<i>f</i> _x =B4*	\$F\$4+C4*\$I	\$6			
			А	В	С	D	E	F	G	Н	
-		1									
		2	Reševanje	e sistema li	nearnih er	načb s Solv	erjem:				
•		3		a1	a2	a3		x	b	a1*x1+a2*x2+a3*x3	
		4	(1)	9,375	3,042	-2,437		0,895	9,233	9,233	
		5	(2)	3,024	6,183	1,216		0,769	8,205	8,205	
•		6	(3)	-2,437	1,216	8,443		0,614	3,934	3,934	

19.2: Reševanje sistema linearnih enačb s Solverjem in definiranimi pogoji.

Najprej zapišemo tabelo z vhodnimi podatki, tj.koeficienti a_i in b_i za vsako enačbo ter začetne vrednosti za x_1 , x_2 in x_3 . Koeficienti **a** so vnešeni v celicah od B4 do D6, koeficienti **b** od G4 do G6 ter ocenjene vrednosti za x_1 , x_2 in x_3 od F4 do F6 (slika 19.2).

Nato definiramo stolpec s celicami od H4 do H6, v katerih bodo izračunane vrednosti **b**jev po formuli: $a_1 x_1 + a_2 x_2 + a_3 x_3$. Po tej formuli izračunamo **b** iz vhodnih podatkov za **a** in ocenjenih **x**. Ker so na začetku vrednosti za **x** ocenjene, se vrednosti podanih **b** in izračunanih razlikujejo. Torej moramo spremeniti vrednosti x, da se bodo vrednosti **b** čim manj razlikovale.

V zavihku DATA kliknemo na gumb Solver (Analysis). Odpre se okno Solver Parameters. Set Target Cell naj bo prazno, pri Equal to možnosti izberemo Min. Value of ni potrebno definirati. V polju By Changing Cells definiramo celice v katerih se bodo spreminjale vrednosti x. Nato v polju Subject to the Constraints definiramo pogoje oziroma omejitve (slika 19.3).

Solver Parameters	×
Set Target Cell:	<u>S</u> olve
Equal To: O Max O Min O Value of: By Changing Cells:	Close
\$F\$4:\$F\$6 Guess	
-Subject to the Constraints:	Options
\$H\$4 = \$g\$4 \$H\$5 = \$g\$5	
\$H\$6 = \$g\$6	Decet All
Delete	

Slika 19.3: Solver z definiranimi pogoji.

Ker želimo spreminjati vrednosti za **x**, na tem mestu postavimo pogoje (definiramo celice) za izračunane in podane **b**-je. Za vnos pogojev kliknemo tipko **Add**. Odpre se okno **Add Constraint** (slika 19.4).

Add Constraint			×
Cell Reference:		Constraint:	
ОК	Cancel	Add	

Slika 19.4: Okno za definiranje pogojev.

V polju **Cell Reference** izberemo calico, v kateri bo izračunana vrednost **b** (npr. H4). V polju **Constraint** izberemo calico, v kateri je podana vrednost **b** (npr. G4). V padajočem meniju na sredini okna izberemo znak = saj želimo, da so vrednosti, ko je rešitev dosežena, enake oziroma se minimalno razlikujejo. Kliknemo gumb **Add** za vnos pogojev. Po istem postopku vnesemo vse pogoje. Na koncu vnosa kliknemo **OK**, da zapremo okno. V oknu **Solver Parameters** so sedaj vidni vsi pogoji (slika 19.3). Sedaj klilknemo gumb **Solve** in Solver reši problem. Rešitev sistema v našem primeru prikazuje stolpec z vrednostmi **x** (celice od F4 do F6) na sliki 19.2.

19.2.2 Minimiranje odstopanj

Za reševanje sistemov linearnih enačb s Solverjem lahko uporabimo tudi alternativni postopek prejšnjemu. V tem primeru minimiramo odstopanja med podanimi in izračunanimi vrednostmi **b**-ja s spreminjanjem vrednosti **x**.

L		B1	5 -	• (•	<i>f</i> _∞ =SUM(B12:B14)			
21		А	В	С	D	E	F	G	Н
	1								
-	2	Reševa	nje sistema l	inearnih en	iačb s Solvei	rjem:			
9	3		a1	a2	a3		x	b	a1*x1+a2*x2+a3*x3
	4	(1)	9,375	3,042	-2,437		0,800	9,233	8,167
	5	(2)	3,024	6,183	1,216		0,700	8,205	7,477
14	6	(3)	-2,437	1,216	8,443		0,600	3,934	3,967
-	7								
• •	8								
15	9								
-	10	Minimi	ranje kvadra	tov odstopa					
	11			Izračun po	enačbi:	1			
-	12		1,136E+00	=(G4-H4)^	2				
	13		5,301E-01						
	14		1,116E-03	Vsota kv	adratov				
	15		1,667E+00	odstopa	nj pri				
-	16			ocenjeni vrednost	h tihx.				
	17								



Slika 19.5 prikazuje primer, v katerem smo v stolpcu od B12 do B14 definirali kvadrate odstopanj v obliki npr. =(G4-H4)^2. Celica B15 prikazuje vsoto kvadratov odstopanj definirano kot =SUM(B12:B14). Dejansko smo sedaj definirali problem najmanjših kvadratov odstopanj, v katerem želimo minimirati vsoto kvadratov odstopanj. Uporabimo Solver s katerim minimiramo to vsoto s spreminjanjem vrednosti x v stolpcu F4 – F6.

Solver Parameters	×
Set Target Cell: B\$15 🔣	<u>S</u> olve
Equal To: C Max Min C Value of: 0	Close
\$F\$4:\$F\$6 Guess	Options
<u>A</u> dd Change	Beret All
Delete	<u>H</u> elp

Slika 19.6: Uporaba Solverja pri minimiranju vsote kvadratov odstopanj.

Slika 19.6 prikazuje, kako definiramo ciljno celico (**Set Target Cell**), tj.celico B15 v našem primeru, v kateri je vrednost, ki jo želimo minimirati. Pri možnosti **Equal to** izberemo **min**. Nato definiramo celice v katerih so vrednosti, ki jih bomo spreminjali, tj. vrednosti **x** v stolpcu F4 – F6 (**By Changing Cells**). Kot vidimo v tem primeru ne definiramo pogojev. Po kliku na gumb Solve dobimo rezultat, ki ga prikazuje slika 19.7.

L		B1	5 +	()	f _∞ =SUM(B12:B14)				
-		А	В	С	D	E	F	G	Н	-
÷	1									
	2	Reševa	nje sistema li	inearnih er	načb s Solve	rjem:				
H	3		a1	a2	a3		x	b	a1*x1+a2*x2+a3*x3	
	4	(1)	9,375	3,042	-2,437		0,895	9,233	9,233	
	5	(2)	3,024	6,183	1,216		0,769	8,205	8,205	
-	6	(3)	-2,437	1,216	8,443		0,614	3,934	3,934	
=	7									
Ŀ	8									
	9									
1	10	Minimi	ranje <mark>kv</mark> adrat	tov odstopa	anj:					
1	11									
•	12		3,086E-14							
9	13		3,397E-14							
-	14		1,047E-14	Minimal	na vsota					
÷	15		7,530E-14	kvadrat	ov					
. 14	16			odstopa	nj.					

Slika 19.7: Rešitev sistema linearnih enačb.

Problemi:

19.1. S principom inverzne matrike in Solverjem rešite naslednji sistem linearnih enačb:

 $x_1 + 2 x_2 + 3 x_3 + 4 x_4 = -2$ $2 x_1 + 3 x_2 + 4 x_3 + x_4 = 2$ $3 x_1 + 4 x_2 + x_3 + 2 x_4 = 2$ $4 x_1 + x_2 + 2 x_3 + 3 x_4 = -2$

(**Rešitev:** $x_1 = 0, x_2 = 1, x_3 = 0, x_4 = -1$)

Vprašanja za preverjanje znanja:

- 1. Na kakšne načine lahko rešujemo sisteme linearnih enačb?
- 2. Kateri funkcijski program uporabimo pri izračunu determinante matrike?
- 3. Katere funkcijske programe, ki so vgrajeni v Excelu, uporabimo pri reševanju sistemov linearnih enačb z inverzno matriko?
- 4. Kakšno obliko imajo enačbe sistema linearnih enačb, ko uporabimo za reševanje orodje Solver?
- 5. Katera dva načina reševanja lahko uporabimo, ko rešujemo sistem linearnih enačb z orodjem Solver?

20. NUMERIČNA INTEGRACIJA

Določene integrale računamo z numeričnimi metodami, če je podintegralska funkcija prezapletena oziroma se ne da elementarno integrirati, ali če jo poznamo samo v diskretnih vrednostih (vozliščih) na nekem intervalu [a,b]. Za numerično integriranje uporabljamo tako imenovane kvadraturne formule. Najbolj znani sta trapezna in Simpsonova formula. Pogledali bomo primer numerične integracije s trapezno formulo⁹.

20.1. Trapezna formula ali trapezno pravilo

Trapezna formula predstavlja interpolacijske polinome prve stopnje, linearne funkcije, ki vežejo dve sosednji vozlišči. Na primer na intervalu $[x_0 \dots x_0+h]$ bo podintegralska funkcija f(x) nadomeščena s premico, ki gre skozi točki (x_0, y_0) in $(x_1, y_1) \ge y_0 = f(x_0)$ in $y_1 = f(x_1)$:

$$\int_{x_o}^{x_{o+h}} f(x) dx \approx \frac{h}{2} (y_o + y_1),$$

kjer je h razlika med x_1 in x_0 .

Če seštejemo prispevke vseh intervalov med dvema sosednjima vozliščema, dobimo trapezno formulo:

$$\int_{a}^{b} f(x) dx \approx h\left(\frac{y_{o}}{2} + y_{1} + y_{2} \dots + y_{n-1} + \frac{y_{n}}{2}\right)$$

Slika 20.1 prikazuje grafično predstavitev trapeznega pravila¹⁵.



Slika 20.1: Grafični prikaz trapeznega pravila¹⁵.

⁹ Bronštejn I. N., Semendjajev K. A., Musiol G., Muehlig H., Matematični priročnik, Tehniška založba Slovenije, 1997.

¹⁵ www.peterstone.name/Maplepgs/numint.html

Interval razdelimo na **n enakih delov**. Ploščina enega segmenta pod krivuljo je izražena kot ploščina trapeza npr:

$$\mathsf{p}_1 = \frac{\mathsf{h}}{2} \big[f(\mathsf{x}_0) + f(\mathsf{x}_1) \big]$$

 $h = x_1 - x_0$

Skupna ploščina pod krivuljo na intervalu od a do b je:

$$p = \sum_{i=1}^{n} p_{i} = p_{1} + p_{2} + \dots + p_{n-1} + p_{n} =$$

$$= \frac{h}{2} [f(x_{o}) + f(x_{1})] + \frac{h}{2} [f(x_{1}) + f(x_{2})] + \dots + \frac{h}{2} [f(x_{n-1}) + f(x_{n})]$$

$$= \frac{h}{2} \cdot f(x_{o}) + h \cdot f(x_{1}) + h \cdot f(x_{2}) + \dots + h \cdot f(x_{n-1}) + \frac{h}{2} \cdot f(x_{n}) =$$

$$= h \cdot \left(\frac{1}{2} \cdot y_{o} + y_{1} + y_{2} + \dots + y_{n-1} + \frac{1}{2} \cdot y_{n}\right)$$

Točnost metode je odvisna od števila trapezov (n), ki jih določimo. Teoretično bi bilo potrebno računati s čim večjim številom trapezov. Na primeru izračuna prostornine cevnega reaktorja uporabimo trapezno formulo.

Primer 20.1: Prostornina cevnega reaktorja¹⁰.

Vodno raztopino reaktanta A presnujemo v produkt v prisotnosti homogenega katalizatorja z določeno koncentracijo. Na hitrost reakcije vpliva samo koncentracija c_A . Znani so naslednji eksperimentalni podatki:

$c_{A}/(mol/L)$	1	2	3	4	5	6	7	8	9	10
-r _A /(mol/(L min)	0,05	0,1	0,15	0,2	0,26	0,33	0,29	0,25	0,19	0,125

Ugotovite prostornino cevnega reaktorja, ki je potrebna za $X_A = 40$ % stopnjo presnove vtoka s prostorninskim tokom $\dot{v} = 10$ L/min in začetno koncentracijo $c_{AO} = 10$ mol/L.

Rešitev:

Najprej vnesemo v Excelovo datoteko eksperimentalne podatke, kot kaže slika 20.2. V stolpcu od A4 do A13 so podatki za koncentracijo reaktanta A, od B4 do B13 pa hitrosti reakcije. Nato v stolpcu C4 do C13 izračunamo recipročne vrednosti hitrosti reakcije $(1/-r_A)$. Enačba, na osnovi katere izračunamo prostornino, je:

$$\tau = \int_{c_{\rm A}}^{c_{\rm Ao}} \frac{\mathrm{d}c_{\rm A}}{-r_{\rm A}}$$

Nato iz zveze:

¹⁰ Krajnc M., Sinteza procesov, zbrano gradivo, Univerza v Mariboru, FKKT, 1999.

 $\tau = \frac{V}{\dot{v}}$

izračunamo prostornino cevnega reaktorja.

Integralno enačbo rešimo s trapeznim pravilom. Določimo h (celica D16), ki je v našem primeru 1 (razlika koncentracij c_A) ter definiramo koeficiente trapezne formule. Vsi koeficienti razen prvega in zadnjega imajo vrednost 1, prvi in zadnji koeficient imata vrednost 0,5 (D9 do D13). Meje integrala smo določili iz podatkov za c_{Ao} = 10 mol/L in končne koncentracije c_A = 6 mol/L (X_A = 40 %). Ploščina pod krivuljo predstavlja bivalni čas τ .

Celica D17 vsebuje trapezno formulo: =D16*SUMPRODUCT(D9:D13;C9:C13), s katero dobimo rezultat. Funkcija SUMPRODUCT izračuna vsoto produktov definiranega polja. Stolpec E9 do E13 prikazuje ploščine posameznih trapezov. Če ploščine seštejemo, dobimo prav tako celotno ploščino trapezov in končni rezultat. Iz podatka o prostorninskem toku (B18) in bivalnem času sedaj izračunamo prostornino reaktorja (B19). Uporabo trapezne metode in rezultat prikazuje slika 20.2.

	D17	• (0	f _x =	D16*SUMP	RODUCT(D	9:D13;C9:C	13)							
	А	В	С	D	E	F	G	Н	1	J	К	L	М	1
1														
2														
		-r _A			ploščina		25							1
3	c₄/(mol/L)	(mol/(Lmin))	1/-r_	koeficient	trapezov		25							
4	1	0,05	20				20							
5	2	0,1	10				20	X I						
6	3	0,15	6,6667					1						
7	4	0,2	5			_	15 -							
8	5	0,26	3,8462											
9	6	0,33	3,0303	0,5	1,515152	Ē.	10 -							
10	7	0,29	3,4483	1	3,448276	3			<u>_</u>					
11	8	0,25	4	1	4	ب» (5 -				-			
12	9	0,19	5,2632	1	5,263158									
13	10	0,125	8	0,5	4		0				1			
14					18,23	t	0	2		e l	。 /	10	12	
15							0	2	4	0	%	10	12	
16			h=	1						c _A /(mol/L)				
17			τ=	18,23	min						_/			
	Prostorninski					Celotna plo	śčina				/			
18	tok:	10	L/min			a apezo m					/			
	Prostornina										(
19	reaktorja je:	182,3	L							/				
20									Označena	ploščina p	od krivuljo	je τ v min		

Slika 20.2: Uporaba trapezne formule.

Po enakem postopku lahko izračunamo določen integral funkcije, ki je podana s formulo.

Problemi:

20.1 Presnova tekočega A teče v prisotnosti homogenega katalizatorja določene koncentracije pri naslednjih hitrostih¹⁰:

¹⁰ Krajnc M., Sinteza procesov, zbrano gradivo, Univerza v Mariboru, FKKT, 1999.

$c_{\rm A}/({\rm mol/L})$	1	2	4	6	7	9	12
$-r_A/(mol/(L\cdot h))$	0,06	0,1	0,25	1,0	2,0	1,0	0,5

Hitrost reakcije določa samo koncentracija c_A . Načrtujemo, da bi izvajali to reakcijo v **diskontinuirnem mešalnem reaktorju** z enako koncentracijo katalizatorja kot pri gornjih poskusih. Ugotovite čas, ki je potreben, da se koncentracija A zniža od $c_{Ao} = 10$ mol/L na $c_A = 2$ mol/L. Uporabite trapezno pravilo.

Namig: Ne pozabite! Interval razdelimo na n enakih delov.

(**Rezultat**: $t \approx 21$ h)

20.2 Tekoči reaktant A pretvorimo v produkt pri naslednjih znanih hitrostih¹⁰:

$c_{A}/(mol/L)$	1	2	4	6	8	10
$-r_A/(\text{mol}/(L\cdot\text{min}))$	0,05	0,1	0,2	0,33	0,25	0,125

Določite prostornino **cevnega pretočnega reaktorja** potrebnega za $X_A = 20$ %, če je prostorninski tok $\dot{v} = 10$ L/min in začetna koncentracija $c_{Ao} = 10$ mol/L. Za reševanje integralne enačbe uporabite trapezno pravilo in Excel.

Namig: Ne pozabite! Interval razdelimo na n enakih delov.

(**Rezultat**: *V* = 115 L)

20.3 Toplotna kapaciteta C_p nekega plina se spreminja s temperaturo¹⁰:

t/°C	20	50	80	110	140	170	200	230
$C_p/(J/mol °C)$	28,95	29,13	29,30	29,48	29,65	29,82	29,99	30,16

Izračunajte toploto, ki jo moramo dovesti 3 mol plina, ki se segreje od 20°C do 230 °C! Za reševanje integralne enačbe uporabite trapezno pravilo in Excel.

(**Rezultat**: $\Delta H = 6208 \text{ J/mol}, Q = 1,86 \times 10^4 \text{ J}$)

Vprašanja za preverjanje znanja:

- 1. Kateri sta najbolj znani formuli za numerično integracijo?
- 2. Kako razdelimo interval [a, b] pod krivuljo pri trapeznem pravilu?
- 3. Od česa je odvisna točnost numerične metode, če je uporabljena trapezna formula?
- 4. Kdaj se bo rezultat numerične integracije dobro ujemal z analitično dobljenim rezultatom?
- 5. Katero funkcijo, ki je vgrajena v Excelovi knjižnici, uporabimo pri numerični integraciji?

¹⁰ Krajnc M., Sinteza procesov, zbrano gradivo, Univerza v Mariboru, FKKT, 1999.

21. NAVADNE DIFERENCIALNE ENAČBE

Obstaja veliko numeričnih metod za reševanje diferencialnih enačb. V tem poglavju se bomo ukvarjali z reševanjem začetnih problemov pri diferencialnih enačbah prvega reda. Za reševanje lahko uporabimo Eulerjevo metodo, Runge-Kutta metodo in še vrsto drugih metod, o katerih je veliko napisanega v literaturi iz tega področja. V nadaljevanju si bomo pogledali delovanje in uporabo Eulerjeve metode.

21.1 Eulerjeva metoda

Eulerjeva metoda ni najboljša metoda za reševanje navadnih diferencialnih enačb (NDE), vendar je enostavna, razumljiva in z usvojenim znanjem o načinu delovanja lažje razumemo delovanje in uporabo drugih numeričnih metod. Pri izračunih je pomembno izbrati primeren korak s katerim se povečuje neodvisna spremenljivka.

Eulerjeva metoda je izpeljana iz Taylorjeve formule za točno rešitev začetnega problema in sicer¹¹:

$$y(t_n+h) = y(t_n) + h \cdot y'(t_n) + \frac{h^2}{2} \cdot y''(\xi_n) \qquad \qquad \xi \in (t_n, t_n+h)$$

Če zanemarimo člen h^2 , dobimo izraz, po katerem iz znane vrednosti y_n izračunamo naslednjo vrednost v zaporedju približkov k rešitvi:

$$y_{n+1} = y_n + h \cdot \frac{dy_n}{dt_n}$$

Geometrijsko formula pomeni, da rešitev enačbe med točkama t_i in t_{i+1} nadomestimo z odsekom premice iz točke t_i v smeri, ki je določena z odvodom $f_i = \frac{dy_i}{dt}$ do točke t_{i+1}.

Postopek izračuna ocene rešitve začetnega problema:

a) Definiramo osnovne podatke, tj. diferencialno enačbo, začetni pogoj in interval:

$$\frac{dy}{dt} = f(t, y)$$
 $y(a) = w$
na intervalu [a,b]

b) Definiramo korak h, ki ga določimo tako, da interval [a,b] razdelimo na n enakih delov.

¹¹ Orel, B., Osnove numerične matematike, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Ljubljana, 2004.

c) Izračunamo:

 $t_i = a + i \cdot h$ $i = 0, 1, 2, \dots, n$

d) Izračunamo približek za w_i :

 $w_i = w_{i-1} + f(t_{i-1}, w_{i-1}) \cdot h$ $i = 1, 2, \dots, n$

Primer 21.1: Usvajanje delovanja Eulerjeve metode⁸. Z Eulerjevo numerično metodo in Excelom rešite navadno diferencialno enačbo:

 $\frac{dy}{dx} = f(x, y) = x + y \qquad \qquad y(0) = 0$

Rešitev najdite na intervalu [0, 1].

Dejansko rešitev dobimo po enačbi: $y = e^x - x - 1$. Primerjajte dobljene rezultate!

Potek reševanja:

Naloga je tipični primer NDE z začetnim problemom. Ker imamo podano dejansko funkcijo ne bi bilo potrebno iskati rešitve z numeričnim reševanjem NDE. Problem je naravnan tako zato, da lahko primerjamo numerične rezultate z dejanskimi in ugotavljamo kateri parametri vplivajo na rešitev.

Izvedli bomo izračuna z dvema različnima prirastkoma za x:

a) h = 0,1 b) h = 0,01

in primerjali dobljene rezultate z dejanskimi.

Potek reševanja z Excelom za prirastek h = 0,1:

a) vnesemo podatek za h (celica B63) in vrednosti za i (A66-A76),

- b) izračunamo vrednosti neodvisne spremenljivke po formuli: $x_i = a + i \cdot h$ (celice B66-B76),
- c) izračunamo približek za y_i po formuli: $y_i = y_{i-1} + f(x_{i-1}, y_{i-1}) \cdot h$ (celice C67-C76),
- d) izračunamo odvod funkcije (celice D66-D76),
- e) izračunamo dejansko vrednost funkcije (celice E66-E76),
- f) izračunamo napako med izračunanim približkom in dejansko vrednostjo funkcije (celice F67-F76).

Slika 21.2 prikazuje rezultate za h = 0,1.

⁸ Bourg, D. M., Excel Scientific and Engineering Cookbook, O'Reilly, CA, 2006.

	E67	•	(j	fx =EXP(B67	=EXP(B67)-B67-1				
	А	В	C D		E	F			
62									
63	h=	0,1							
64									
65	i	x _i	y i	dy _i /dx _i	y _i (dejanski)	napaka			
66	0	0	0,000000	0,00000	0,000000				
67	1	0,1	0,000000	0,10000	0,005171	0,005171			
68	2	0,2	0,010000	0,21000	0,021403	0,011403			
69	3	0,3	0,031000	0,33100	0,049859	0,018859			
70	4	0,4	0,064100	0,46410	0,091825	0,027725			
71	5	0,5	0,110510	0,61051	0,148721	0,038211			
72	6	0,6	0,171561	0,77156	0,222119	0,050558			
73	7	0,7	0,248717	0,94872	0,313753	0,065036			
74	8	0,8	0,343589	1,14359	0,425541	0,081952			
75	9	0,9	0,457948	1,35795	0,559603	0,101655			
76	10	1	0,593742	1,59374	0,718282	0,124539			
77									

Slika 21.2: Prikaz rezultatov pri h = 0,1.

Iz rezultatov je razvidna razlika (stolpec F67 – F76) med dejanskimi (stolpec E67 – E76) in izračunanimi vrednostmi y, dobljenimi z Eulerjevo metodo (stolpec C67 – C76). Kaj to pomeni bo lažje ugotoviti iz grafične predstavitve rezultata.

Slika 21.3 prikazuje del rezultatov pri h = 0,01, slika 21.4 pa grafični prikaz dejanskega rezultata in rezultatov dobljenih z Eulerjevo metodo pri h = 0,1 in h = 0,01 v območju [0; 0,5].

Iz slike 21.4 vidimo, da so rezultati dobljeni z Eulerjevo metodo pri h = 0,01 (modra krivulja) dober približek dejanskim rezultatom (rdeča krivulja). Rezultati pri h = 0,1 so slab približek saj se z večanjem neodvisne spremenljivke x veča tudi razlika v primerjavi z dejanskim rezultatom.

<u>Ugotovitev je naslednja</u>: z nižanjem vrednosti koraka h nižamo razliko med dejanskim in numeričnim rezultatom, a večamo število korakov.

	F8	-	(<i>f</i> ≈ =E8-C8		
	А	В	С	D	E	F
4	Enačba de	janske fun	kcije: y=e [×] -x	-1	h=	0,01
5						
6	i	x _i	y i	dy _i /dx _i	y _i (dejanski)	napaka
7	0	0	0	0	0	
8	1	0,01	0	0,01	5,0167E-05	5,017E-05
9	2	0,02	0,0001	0,0201	0,00020134	0,0001013
10	3	0,03	0,000301	0,030301	0,00045453	0,0001535
11	4	0,04	0,000604	0,04060401	0,00081077	0,0002068
12	5	0,05	0,0010101	0,05101005	0,0012711	0,000261
13	6	0,06	0,0015202	0,061520151	0,00183655	0,0003164
14	7	0,07	0,0021354	0,072135352	0,00250818	0,0003728
15	8	0,08	0,0028567	0,082856706	0,00328707	0,0004304
16	9	0,09	0,0036853	0,093685273	0,00417428	0,000489
17	10	0,1	0,0046221	0,104622125	0,00517092	0,0005488
18	11	0,11	0,0056683	0,115668347	0,00627807	0,0006097
19	12	0,12	0,006825	0,12682503	0,00749685	0,0006718
20	13	0,13	0,0080933	0,13809328	0,00882838	0,0007351
21	14	0,14	0,0094742	0,149474213	0,0102738	0,0007996
22	15	0,15	0,010969	0,160968955	0,01183424	0,0008653
23	16	0,16	0,0125786	0,172578645	0,01351087	0,0009322
24	17	0,17	0,0143044	0,184304431	0,01530485	0,0010004
25	18	0,18	0,0161475	0,196147476	0,01721736	0,0010699
26	19	0,19	0,018109	0,20810895	0,0192496	0,0011406
27	20	0,2	0,02019	0,22019004	0,02140276	0,0012127

Slika 21.3: Delni prikaz rezultatov z Excelom pri h = 0,01.



Slika 21.4: Grafični prikaz dejanskega rezultata in rezultatov dobljenih z Eulerjevo metodo na območju [0; 0,5].

<u>Primer 21.2</u>: Hitrost razpadanja N_2O_5 . Pri temperaturi t = 25 °C se vrši razpad N_2O_5 po reakciji¹²:

 $2 \text{ N}_2\text{O}_5 \rightarrow 4 \text{ NO}_2 + \text{O}_2$

Eksperimentalni podatki so naslednji:

<i>t</i> /min	c _{N2O5} / (mol/L)
0	1,5 × 10⁻²
200	9,6 × 10⁻³
400	6,2 × 10 ⁻³
600	4,0 × 10⁻³
800	2,5 × 10⁻³
1000	1,6 × 10⁻³

Reakcija je prvega reda zato lahko zapišemo: $-\frac{dc_A}{dt} = k \cdot c_A$, kjer je A komponenta N₂O₅. Konstanta reakcijske hitrosti $k = 2,2 \times 10^{-3} \text{ min}^{-1}$.

Z Eulerjevo metodo in programom Excel rešite diferencialno enačbo:

$$-\frac{\mathrm{d}c_{\mathrm{A}}}{\mathrm{d}t} = k \cdot c_{\mathrm{A}}$$
, če je $c_{\mathrm{A}}(0) = 0,015 \frac{\mathrm{mol}}{\mathrm{L}}$

na intervalu [0, 1000].

Pri reševanju uporabite korak h = 50 min. Numerične in eksperimentalne rezultate prikažite grafično in jih primerjajte.

Potek reševanja z Excelom:

- a) vnesemo osnovne podatke: v celici B3 je podatek za korak, v celici B4 je vrednost reakcijske konstante,
- b) v celici B6 je vnesena vrednost za čas t_0 ,
- c) v celici C6 je podatek za začetno vrednost koncentracije c_A pri času t = 0,
- d) v celici D6 je izračunana vrednost $\frac{dc_A}{dt}$ pri t = 0 po enačbi **=-\$B\$4 * C6**.

e) postopek od **b** do **d** se ponovi za vsak prirastek časa *t*.

Celotni izračun prikazuje slika 21.5.

¹² Atkins, P., Jones L., Chemical Principles, The Quest for Insight, W. H. Freeman and Company, New York, 1999.

	C7	-	(J	€ =C6+\$B\$3	*D6			
	А	В	С	D	E	F	G	Н
2	Primer 21.	2 v učben	iku Računaln	ištvo v kemiji	:	A=N ₂ O ₅	B=NO ₂	C=O2
3	h=	50	min			Reakcija: 2	$A \rightarrow 4B + C$	
4	k=	0,0022	min ⁻¹					
5	i	t _i (min)	c _{Ai} (mol/L)	dC _A /dt				
6	0	0	0,015000	-0,000033				
7	1	50	0,013350	-0,00002937				
8	2	100	0,011882	-2,6139E-05				
9	3	150	0,010575	-2,3264E-05				
10	4	200	0,009411	-2,0705E-05				
11	5	250	0,008376	-1,8427E-05				
12	6	300	0,007455	-1,64E-05				
13	7	350	0,006635	-1,4596E-05				
14	8	400	0,005905	-1,2991E-05				
15	9	450	0,005255	-1,1562E-05				
16	10	500	0,004677	-1,029E-05				
17	11	550	0,004163	-9,1581E-06				
18	12	600	0,003705	-8,1507E-06				
19	13	650	0,003297	-7,2541E-06				
20	14	700	0,002935	-6,4562E-06				
21	15	750	0,002612	-5,746E-06				
22	16	800	0,002325	-5,1139E-06				
23	17	850	0,002069	-4,5514E-06				
24	18	900	0,001841	-4,0507E-06				
25	19	950	0,001639	-3,6052E-06				
26	20	1000	0,001458	-3,2086E-06				

Slika 21.5: Prikaz rezultatov hitrosti razpadanja N₂O₅.





Slika 21.6 prikazuje hitrost razpadanja N_2O_5 pri t = 25 °C. Modra krivulja je dobljena na osnovi eksperimentalnih podatkov, rumeni kvadratki prikazujejo rezultate dobljene z Eulerjevo metodo. Ugotovimo lahko, da pri izbranem koraku h = 50 min, numerično reševanje navadne diferencialne enačbe da zadovoljive rezultate.

Problemi:

21.1 Naslednji problem rešite z Excelom in uporabite Eulerjevo metodo¹³:

 $\frac{dy}{dx} = y + x^2 \qquad \qquad y(0) = 1$

na intervalu [0, 2].

Dejansko rešitev dobimo po enačbi: $y = 3 e^{x} - x^{2} - 2 x - 2$.

a) Numerični izračun izvedite pri h = 0,2.

b) Narišite graf in primerjajte rezultate po Eulerju z dejanskimi rezultati!

21.2 Z Eulerjevo metodo in Excelom rešite naslednjo diferencialno enačbo¹⁴:

 $\frac{dy}{dt} = 2 - e^{-4t} - 2y$ y(0) = 1na intervalu [0, 0,5]

Dejansko rešitev dobimo po enačbi : $y(t) = 1 + \frac{1}{2}e^{-4t} - \frac{1}{2}e^{-2t}$.

- a) Uporabite korak h = 0,1.
- b) Izračunajte napako med dejanskimi in numeričnimi rezultati.

Napako si lažje predstavljamo, če jo izrazimo v %:

napaka (%) = <u>|dejanska vrednost - numericna vrednost|</u>.100 dejanska vrednost

c) Narišite graf z dejanskimi in numeričnimi rezultati in komentirajte rezultat.

21.3 Z Excelom in Eulerjevo metodo poiščite približno rešitev začetnega problema¹¹:

 $\frac{dy}{dx} = -y + 1$ y(0) = 2 na intervalu [0, 1]

¹³ Townsend L., Using Excel to solve differential equations, 2003.

¹⁴ Dawkins P. <u>http://tutorial.math.lamar.edu/terms.aspx</u> , 2007.

¹¹ Orel, B., Osnove numerične matematike, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Ljubljana, 2004.

Dejansko rešitev problema dobimo po enačbi: $y = e^{-x} + 1$.

- a) Uporabite korak h = 0,1.
- b) Izračunajte napako med približno (numerično) in dejansko rešitvijo.
- c) Narišite graf z dejanskimi in numeričnimi rezultati in komentirajte rezultat.

Vprašanja za preverjanje znanja:

- 1. Kakšne enačbe lahko rešujemo z opisano Eulerjevo metodo?
- 2. Ali poznate še kakšno numerično metodo za reševanje NDE?
- 3. Na osnovi katere formule je izpeljana Eulerjeva numerična metoda?
- 4. Kako vpliva korak h na končno rešitev NDE?
- 5. Kako izračunamo napako med dejanskimi in numeričnimi rezultati?
- 6. Poiščite probleme, ki jih lahko rešite z Eulerjevo numerično metodo, pri predmetih, ki jih poslušate!

22. TISKANJE

22.1 Predogled tiskanja

Pred tiskanjem preglednice je priporočljivo pogledati, kako bo natisnjeni dokument izgledal. Izgled preglednice dobimo na zaslon z ukazom **Office/Natisni/Predogled tiskanja** (slika 22.1). Če z izgledom dokumenta nismo zadovoljni, dobimo s klikom na ukaz Priprava strani možnost, kjer lahko: določimo usmerjenost lista, spremenimo merilo, nastavimo robove, uredimo glavo in noge strani, določimo poravnavo preglednice glede na stran, vključimo tiskanje mrežnih črt.

22.2 Priprava strani

Na zavihku Stran nastavimo:

- usmerjenost lista,
- merilo ali prilagoditev,
- velikost lista.

A	9 - ((²¹ +	Ŧ	(Zvezek1 - M
	<u>N</u> ovo		Predogled in tiskanje doku	m	Predogled tiskanja
Ê	O <u>d</u> pri		Pred tiskanjem izber možnosti tiskanja.	rit	Natisni Priprava strani Natisni Priprava strani
	<u>S</u> hrani		Hitro tiskanje Pošljite nespremenj privzetemu tiskalnik	er :u	Priprava strani ?X
R	Sh <u>r</u> ani kot	•	Predogled tiskanja Pred tiskanjem si og Pred	ule dic	Stran Robovi Glava/Noga List Usmerjenost
	Na <u>t</u> isni	•			A OLeže <u>č</u> e
Ż	Pripra <u>v</u> i	•			Spreminjanje merila Nastavi na: 100 % navadne velikosti
	P <u>o</u> šlji	×			O Prilagodi na: 1 🗘 stran(i) širine z 1 🗘 višine
	O <u>bj</u> avi	•			Velikost papirja: A4
	Zapr <u>i</u>				Številka prve strani: Samodejno
15			E 6	x	
16					
17					Možnosti
18					
20					V redu Prekliči



Na zavihku **Robovi** nastavimo:

- robove,
- oddaljenost glave in noge od roba lista,
- vodoravno in/ali navpično usredinjenost preglednice.

Na zavihku Glava/Noga uredimo:

- besedilo, ki se izpiše v glavi lista,
- besedilo, ki se izpiše v nogi lista.

Glavo in nogo lista lahko uredimo, če kliknemo na gumb **Glava po meri** oziroma **Noga po meri**.

Na zavihku List nastavimo:

- področje tiskanja s klikom na gumb lahko z miško označimo območje celic za tiskanje,
- območje celic, ki se natisne na vsaki strani npr. naslovna vrstica preglednice,
- tiskanje mrežnih črt.

22.3 Tiskanje

Za tiskanje dokumenta iz okna Predogled tiskanja kliknemo na gumb **Natisni** (slika 22.2). Tiskanje lahko sprožimo tudi s klikom na ukaz Natisni/Natisni v meniju Office vrstici.

V pogovornem oknu določimo:

- obseg strani (listov) vse ali samo določene strani,
- kaj želimo izpisati (izbor, delovni list ali cel zvezek),
- število kopij.



Slika 22.2: Tiskanje Excelovega dokumenta.

PRILOGE

Priloga A: Ascii kode¹⁶

etamatter key binary etcl dec hex Definition of Use NULL CTR L/A 0.000 000 0	ASCI	Terminal		Equivalent Fo	rms		
NUL CTR L/@ 0 0000 00 0 Filler: Null SOH CTR L/B 0 000 001 002 2 2 Start of Heading: Home Position STX CTR L/D 0 000 100 003 3 End of Text EOT CTR L/D 0 000 100 004 4 End of Text EOT CTR L/D 0 000 100 005 5 Enquiry ACK CTR L/A 0 000 101 005 5 Enquiry ACK CTR L/A 0 001 101 018 8 Backspace; arrow HT CTR L/A 0 001 101 11 18 Vertical Tabulation; arrow VT CTR L/A 0 001 101 115 15 D Carriage Return SO CTR L/A 0 001 101 115 F Shift fout Shift fout SI CTR L/A 0 001 101 122 18 Interedd; l	character	key	binary	oct	dec	hex	Definition of Use
	NUL	CTRL/@	0 000 000	000	0	0	Filler; Null
STX CTRL/B 0 000 010 002 2 2 Start of Text EOT CTRL/D 0 000 100 004 4 4 End of Text EOT CTRL/D 0 000 110 005 5 5 Enquiry ACK CTRL/F 0 000 110 006 6 6 Acknowledge BEL CTRL/H 0 001 010 012 10 A Line Feed; Larrow HT CTRL/J 0 001 010 012 10 A Line Feed; Larrow VT CTRL/A 0 001 101 013 11 B Vertical Tabulation; Larrow FF CTRL/A 0 001 101 015 13 D Carriage Return SO CTRL/A 0 001 100 012 18 12 Device Control 1 DC1 CTRL/A 0 010 010 022 18 Negative Acknowledge SYN CTRL/A 0 010 010 023 11 End of Transmission Block CR CTRL/A <td>SOH</td> <td>CTRL/A</td> <td>0 000 001</td> <td>001</td> <td>1</td> <td>1</td> <td>Start of Heading; Home Position</td>	SOH	CTRL/A	0 000 001	001	1	1	Start of Heading; Home Position
ETX CTRL/C 0 000 011 003 3 3 End of Transmission ENQ CTRL/E 0 000 101 005 5 5 End of Transmission ACK CTRL/E 0 000 110 005 6 6 Acknowledge BEL CTRL/G 0 000 100 010 017 7 Bell BS CTRL/H 0 001 001 012 0 Acknowledge = arrow HT CTRL/A 0 001 010 012 0 Acknowledge = arrow VT CTRL/M 0 001 101 013 11 B Vertical Tabulation: [arrow FF CTRL/M 0 001 101 015 13 D Carriage Return SO CTRL/O 0 001 101 012 17 11 Device Control 2 DC2 CTRL/G 0 100 010 022 18 12 Device Control 2 DC3 CTRL/G 0 101 010 025 21 15 Negative Acknowledge	STX	CTRL/B	0 000 010	002	2	2	Start of Text
EOT CTRL/D 0 000 100 004 4 Enquiry ACK CTRL/F 0 000 110 005 5 5 Enquiry ACK CTRL/F 0 000 110 006 6 6 Acknowledge BEL CTRL/G 0 001 000 011 9 9 Horizontal Tabulation LF CTRL/J 0 001 010 011 13 D Vertical Tabulation; 1 arrow FF CTRL/N 0 001 100 015 14 E Shift Out SO CTRL/N 0 001 110 015 14 E Shift Out SO CTRL/N 0 001 100 021 16 10 Data Link Escape DC1 CTRL/Q 0 010 000 021 17 11 Device Control 1 DC2 CTRL/N 0 010 101 023 12 15 Negative Acknowledge SYN CTRL/V 0 010 101 024 20 14 Device Control 3 DC4 CTRL/V	ETX	CTRL/C	0 000 011	003	3	3	End of Text
ENQ CTRL/F 0 000 101 005 5 Enquiry ACK CTRL/G 0 000 111 007 7 Bell BS CTRL/H 0 001 000 010 8 B Backspace; arrow HT CTRL/H 0 001 010 012 10 A Line Feed; jarrow VT CTRL/M 0 001 010 012 10 A Line Feed; jarrow FF CTRL/N 0 001 100 014 12 C Form Feed; Clear Screen CR CTRL/N 0 001 100 014 12 C Form Feed; Clear Screen SO CTRL/P 0 001 000 020 16 10 Data Link Escape DC1 CTRL/Q 0 010 010 021 17 11 Device Control 2 DC3 CTRL/Y 0 010 010 022 18 12 Device Control 4 NAK CTRL/Y 0 010 100 024 18 Ecord Synchronous Ide SYN CTRL/Y 0 010 100	EOT	CTRL/D	0 000 100	004	4	4	End of Transmission
ACK CTRL/F 0 000 10 006 6 6 Acknowledge BEL CTRL/H 0 000 000 010 8 Backspace;	ENQ	CTRL/E	0 000 101	005	5	5	Enquiry
BEL CTRL/I 0 000 000 010 8 8 Backspace; arrow HT CTRL/I 0 000 010 012 0 Horizontal Tabulation; arrow VT CTRL/K 0 000 010 012 10 A Line Feed; arrow FF CTRL/M 0 000 010 013 11 B Vertical Tabulation; arrow SO CTRL/M 0 000 101 015 13 D Carriage Return SO CTRL/O 0 001 101 016 14 E Shift Out SO CTRL/Q 0 010 000 020 16 10 Data Link Escape DC1 CTRL/V 0 010 000 22 18 12 Device Control 2 DC3 CTRL/V 0 010 010 024 21 15 Negative Acknowledge SYN CTRL/V 0 010 010 031 25 19 End of M	ACK	CTRL/F	0 000 110	006	6	6	Acknowledge
BS CTRL/H 0 000 000 010 8 Backspace;	BEL	CTRL/G	0 000 111	007	7	7	Bell
HT $CTRL/I$ 0 00100101199Horizontal labulationUT $CTRL/K$ 0 00101001311BVertical TabulationFF $CTRL/K$ 0 00110001412CForm Feed; Clear ScreenCR $CTRL/M$ 0 0011000161412CForm Feed; Clear ScreenSO $CTRL/N$ 0 00111001614EShift foutSI $CTRL/Q$ 0 100000201610Data Link EscapeDC1 $CTRL/Q$ 0 1000010211711Device Control 1DC2 $CTRL/R$ 0 1001010231913Device Control 2DC3 $CTRL/S$ 0 1000100252115Negative AcknowledgeSYN $CTRL/V$ 0 1010100252115Negative AcknowledgeSYN $CTRL/V$ 0 101010032261ASubstituteEBC $CTRL/V$ 0 101010032261ASubstituteESC $CTRL/V$ 0 101010032261ASubstituteESC $CTRL/V$ 0 011010032261ASubstituteESC $CTRL/V$ 0 011010032261ASubstituteUS $CTRL/V$ 0 011010033271BEscapeSUB $CTRL/V$ 0 011010033271B	BS	CTRL/H	0 001 000	010	8	8	Backspace; — arrow
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	HT	CTRL/I	0 001 001	011	9	9	Horizontal labulation
V1 C1RL/N 0 001 011 013 11 B Ortheat Iabitation, partow FF CTRL/M 0 001 101 015 13 D Carriage Return SO CTRL/M 0 001 110 016 14 12 C Form Feed; Clear Screen SO CTRL/O 0 001 110 016 14 12 C Form Feed; Clear Screen SO CTRL/Q 0 010 000 020 16 10 Data Link Escape DC1 CTRL/R 0 100 010 022 18 12 Device Control 1 DC3 CTRL/N 0 100 010 025 21 15 Negative Acknowledge SYN CTRL/V 0 010 010 023 24 18 Device Control 4 NAK CTRL/V 0 010 010 023 24 18 Device Control 4 NAK CTRL/Y 0 010 011 023 24 18 Device Control 4 NAK CTRL/Y 0 010 031 25 17 End of Medium S	LF	CTRL/J	0 001 010	012	10	A	Line Feed;] arrow
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	VI	CIKL/K	0 001 011	013	11	D	Form Food: Close Server
CK CIRL/N 0 0 011 012 12 D Canady Kenth SO CTRL/O 0 001 110 017 15 F Shift To DLE CTRL/P 0 000 000 021 17 11 Device Control 1 DC1 CTRL/R 0 000 010 021 17 11 Device Control 1 DC2 CTRL/R 0 000 010 022 18 13 Device Control 3 DC3 CTRL/V 0 001 100 22 18 Negative Acknowledge SYM CTRL/V 0 001 100 25 15 Negative Acknowledge SYM CTRL/V 0 001 003 24 18 Cancel EM CTRL/V 0 011 000 031 25 19 End of Medium SUB CTRL/V 0 011 013 27 18 Escape FS CTRL/I 0 0110 032 20	CP	CTRL/L	0 001 100	014	12	D	Carriage Return
Si CTRL/D 0 000 111 011 15 F Shift In DLE CTRL/P 0 010 000 020 16 10 Data Link Escape DC1 CTRL/P 0 010 001 021 17 11 Device Control 1 DC2 CTRL/R 0 010 010 022 18 12 Device Control 2 DC3 CTRL/T 0 010 100 024 20 14 Device Control 3 DC4 CTRL/T 0 010 101 025 21 15 Negative Acknowledge SYN CTRL/V 0 010 101 026 22 16 Synchronous Idle ETB CTRL/V 0 010 101 026 21 17 End 0 Transmission Block CAN CTRL/V 0 011 001 031 25 19 End of Medium SUB CTRL/Z 0 011 001 032 26 1A Substitute ESC CTRL/Z 0 011 001 032 26 1A Substitute ESC CTRL/I 0 011 100 034 28 1C File Separator GS CTRL/I 0 011 100 035 29 1D Group Separator GS CTRL/I 0 011 110 035 29 1D Group Separator US CTRL/Z 0 011 110 035 29 1D Group Separator US CTRL/Z 0 011 110 035 20 Spece Blank ! ! 0 100 001 041 33 21 Exclamator Mark ! ! 0 100 001 041 33 21 Spece Blank SP SPACE 0 100 000 044 36 24 Dollar Symbol S P SPACE 0 100 010 044 36 24 Dollar Symbol S * 0 100 010 044 36 24 Dollar Symbol S * 0 100 010 044 36 24 Dollar Symbol S * 0 100 010 044 36 26 Ampersand * * * 0 100 011 045 37 27 Single Quote; Apostrophe ((0 010 100 050 40 28 Left Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 051 41 29 Right Parenthesis * * * 0 101 010 054 44 2C Comma 0 101 100 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 45 2D Minus Symbol 0 101 100 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 054 44 2C Comma * * * 0 101 010 055 45 2D Minus Symbol; Hypten 0 101 100 054 44 2C Comma * * * 0 101 100 056 45 36 Digit Five § 8 8 0 111 000 070 56 38 35 Digit Five § 9 9	SO	CTRL/M	0 001 110	016	14	F	Shift Out
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	SI	CTRL/O	0 001 111	017	15	F	Shift In
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	DLE	CTRL/P	0 010 000	020	16	10	Data Link Escape
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	DC1	CTRL/O	0 010 001	021	17	11	Device Control 1
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	DC2	CTRL/R	0 010 010	022	18	12	Device Control 2
DC4 CTRL/T 0 010 100 024 20 14 Device Corrol 4 NAK CTRL/V 0 010 110 025 21 15 Negative Acknowledge SYN CTRL/V 0 010 111 027 23 17 End of Transmission Block CAN CTRL/X 0 011 100 030 24 18 Cancel EM CTRL/Z 0 011 010 031 25 19 End of Medium SUB CTRL/Z 0 011 010 032 26 1A Substitute ESC CTRL/1 0 011 101 033 27 18 Escape FS CTRL/1 0 011 101 036 30 1E Record Separator GS CTRL/1 0 011 111 036 30 1E Record Separator SP SPACE 0 100 001 041 33 21 Exclamation Mark " 0 100 001 041 33 21 Exclamation Mark	DC3	CTRL/S	0 010 011	023	19	13	Device Control 3
NAK CTRL/V 0 010 10 025 21 15 Negative Acknowledge SYN CTRL/V 0 010 110 026 22 16 Synchronous Idle ETB CTRL/X 0 010 010 030 24 18 Cancel CAN CTRL/Y 0 011 001 031 25 19 End of Medium SUB CTRL/Z 0 011 010 032 26 1A Substitute ESC CTRL/Z 0 011 010 033 27 1B Escape FS CTRL/I 0 011 100 034 28 1C File Separator RS CTRL/I 0 011 110 035 29 1D Group Separator RS CTRL/I 0 011 111 037 31 1F Unit Separator SP SPACE 0 100 010 041 33 21 Exclamation Mark " * 0 100 010 041 33 23 Number Symbol \$\$ \$ </td <td>DC4</td> <td>CTRL/T</td> <td>0 010 100</td> <td>024</td> <td>20</td> <td>14</td> <td>Device Control 4</td>	DC4	CTRL/T	0 010 100	024	20	14	Device Control 4
SYN CTRL/V 0 010 111 026 22 16 Synchronous Idle ETB CTRL/X 0 011 000 030 24 18 Cancel EM CTRL/X 0 011 001 031 25 19 End of Transmission Block SUB CTRL/Z 0 011 010 032 26 1A Substitute ESC CTRL/I 0 011 100 032 27 1B Escape FS CTRL/I 0 011 100 034 28 1C File Separator GS CTRL/I 0 011 110 036 30 1E Record Separator US CTRL/I 0 011 010 042 34 22 Double Quate # 0 100 001 043 35 23 Number Symbol \$ 0 100 101 043 35 24 Number Symbol \$ 0 100 101 045 37 25 Percent Symbol \$ 0 100 101 045 37 25<	NAK	CTRL/U	0 010 101	025	21	15	Negative Acknowledge
ETB CTRL/W 0 010 111 027 23 17 End of Transmission Block CAN CTRL/X 0 011 001 030 24 18 Cancel EM CTRL/Z 0 011 001 031 25 19 End of Medium SUB CTRL/Z 0 011 011 010 032 26 1A Subtitute ESC CTRL/I 0 011 010 033 27 1B Escape FS CTRL/I 0 011 100 034 28 1C Filesparator US CTRL/- 0 011 111 037 31 1F Unit Segrator SP SPACE 0 100 000 040 33 21 Exclamation Mark ' ' 0 100 010 041 33 21 Exclamation Mark ' ' 0 100 010 041 33 23 Number Symbol S 0 100 010 044 36 24 Dollar Symbol & <th< td=""><td>SYN</td><td>CTRL/V</td><td>0 010 110</td><td>026</td><td>22</td><td>16</td><td>Synchronous Idle</td></th<>	SYN	CTRL/V	0 010 110	026	22	16	Synchronous Idle
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	ETB	CTRL/W	0 010 111	027	23	17	End of Transmission Block
EM $CTRL/Z$ 0 011 0010312519End of MediumSUB $CTRL/Z$ 0 011 011033271BEscapeFS $CTRL/I$ 0 011 100034281CFile SeparatorGS $CTRL/I$ 0 011 100034281CFile SeparatorRS $CTRL/I$ 0 011 110036301ERecord SeparatorUS $CTRL/I$ 0 011 111037311FUnit SeparatorSPSPACE0 100 0000403220Space; Blank**0 100 0100413321Exclamation Mark**0 100 0100423422Double Quote##0 100 0100433523Number Symbol\$\$0 100 1000443624Dollar Symbol\$\$0 100 1100453725Percent Symbol\$\$0 100 1100463826Ampersand**0 101 0010514129Right Parenthesis**0 101 0010514129Right Parenthesis**0 101 101053432BPlus Symbol**0 101 101054442CComma**0 101 101055452DMinus Symbol; Haypen**0 101 101056462EPeriod; Decimal Point* <t< td=""><td>CAN</td><td>CTRL/X</td><td>0 011 000</td><td>030</td><td>24</td><td>18</td><td>Cancel</td></t<>	CAN	CTRL/X	0 011 000	030	24	18	Cancel
SUBCTRL/Z00002261ASubstituteESCCTRL/I001110033271BEscapeFSCTRL/I001110034281CFile SeparatorGSCTRL/I001110035291DGroup SeparatorRSCTRL/I0001111003601ERecord SeparatorUSCTRL/I000100403220Space: Blank**01000010413321Exclamation Mark**01000100413523Number Symbol\$\$01001010433725Percent Symbol\$\$01001010463826Ampersand''01001010463826Ampersand''0100101051432BPlus Symbol\$\$0101011053432BPlus Symbol\$.0101011053432BPlus Symbol\$.0101011055452DMinus Symbol; Hapten\$0101105452DMinus Symbol; Hapten\$010110<	EM	CTRL/Y	0 011 001	031	25	19	End of Medium
ESCCTRL/10001100023271BEscapeFSCTRL/1000034281CFile SeparatorRSCTRL/10000035291DGroup SeparatorRSCTRL/100000301ERecord SeparatorSPSPACE01000000403220Space; Blank""01000100413321Exclamation Mark""01000100423422Double Quote##01000100433523Number Symbol\$\$01001010433525Percent Symbol\$\$01001010453725Percent Symbol\$\$0100110473927Single Quote; Apostrophe\$\$0100101052422AAsterisk**0101101055452DMinus Symbol; Hyphen\$-0101101055452DMinus Symbol; Hyphen\$-0101105452DMinus Symbol; Hyphen\$-0101055452DMinus Symbol; Siah; Virgule\$0<	SUB	CTRL/Z	0 011 010	032	26	1A	Substitute
PS CIRL/1 0 011 100 0.34 28 IC File Separator GS CTRL/1 0 011 110 0.36 30 1E Record Separator US CTRL/2 0 011 110 0.36 30 1E Record Separator SP SPACE 0 000 000 040 32 20 Space; Blank ! ! 0 100 001 041 33 21 Exclamation Mark " 0 100 011 043 35 23 Number Symbol \$ 9 0 100 011 043 35 23 Number Symbol \$ \$ 0 100 011 043 35 24 Ampersand " 0 100 101 044 36 24 Dollar Symbol \$ 0 101 001 011 051 41 29 Right Parenthesis \$ 0 101 001 051	ESC	CTRL/[0 011 011	033	27	18	Escape
GS CIRL/1 0 011 101 0.35 29 1D Group Separator RS CTRL/- 0 011 111 0.37 31 1F Record Separator SP SPACE 0 100 000 040 32 20 Space; Blank " " 0 100 001 041 33 21 Exclamation Mark " " 0 100 001 043 35 23 Number Symbol \$ \$ 0 100 010 044 36 24 Dollar Symbol \$ \$ 0 100 101 043 35 23 Number Symbol \$ \$ 0 100 101 043 35 24 Dollar Symbol \$ \$ 0 100 101 043 35 24 Dollar Symbol \$ \$ 0 100 101 045 37 25 Percent Symbol \$ 0 101 010	FS	CTRL/\	0 011 100	034	28	IC	File Separator
KSCIRL/ US0 011 1100.36301ERecord SeparatorUSCTRL/ .0 011 111037311FUnit SeparatorSPSPACE0 100 0000403220Space; Blank"10 100 0100413321Exclamation Mark""0 100 0100423422Double Quote##0 100 0110433523Number Symbol\$\$0 100 1000443624Dollar Symbol%%0 100 1100453725Percent Symbol%%0 100 1100463826Ampersand''0 100 1100463826Ampersand''0 100 1100514129Right Parenthesis**0 101 1000514129Right Parenthesis**0 101 101053432BPlus Symbol0 101 101055452DMinus Symbol; Hyphen0 101 101056462EPeriod; Decimal Point//0 101 1000604830Digit Zero0 101 1000614931Digit One220 110 0100645335Digit Three0 110 0100645335Digit Five0 110	GS	CTRL/J	0 011 101	035	29	ID	Group Separator
OS $CIR(J_{})$ 0 0 0 0 31 17 0 0 0 0 31 17 $CIR (2e)$ $Space (2e)$	KS	CTRL/	0 011 110	030	30	IE	Kecord Separator
SrSrACL01000000403220Space main""01000010423422Double Quote##01000110433523Number Symbol\$\$01000443624Dollar Symbol%%01001010453725Percent Symbol&&00011010463826Ampersand''01001010463826Ampersand''01010000504028Left Parenthesis((-01010000504028Left Parenthesis**01010110514129Right Parenthesis**0101010052422AAsterisk++0101101053432BPlus Symbol0101101055452DMinus Symbol; Hyphen0101105452DMinus Symbol; Slash; Virgule0101101055452DMinus Symbol; Slash; Virgule01011065430Digit Zero01010614931Digit Two	SP	SPACE	0 100 000	040	37	20	Space: Blank
0 100 011 042 34 22 Double Quote # 0 100 011 043 35 23 Number Symbol \$ \$ 0 100 101 043 35 23 Number Symbol \$ \$ 0 100 101 043 35 24 Dollar Symbol \$ \$ 0 100 101 044 36 24 Dollar Symbol \$ 0 100 101 044 36 24 Dollar Symbol \$ 0 100 111 047 39 27 Single Quote; Apostrophe ((0 101 0050 40 28 Left Parenthesis)) 0 101 051 41 29 Right Parenthesis . . 0 101 1053 43 28 Plus Symbol . . 0 101 1053 43 20 Comma . . 0	J	J	0 100 000	040	32	20	Space, Dialik Exclamation Mark
# # 0 100 012 53 23 Number Symbol \$ \$ 0 100 100 043 35 23 Number Symbol \$ \$ 0 100 101 045 37 25 Percent Symbol \$ 0 100 101 045 37 25 Percent Symbol \$ 0 100 111 047 39 27 Single Quote; Apostrophe ((0 101 001 051 41 29 Right Parenthesis $* 0 101 001 051 41 29 Right Parenthesis * 0 101 001 051 41 29 Right Parenthesis * 0 101 001 052 42 2A Asterisk + + 0 101 055 45 2D Minus Symbol; Hyphen . . 0 101 056 45 2D Minus Symbol; Slash; Virgule $			0 100 010	042	34	22	Double Quote
\$ \$ 0 100 044 36 24 Dollar Symbol % % 0 100 101 045 37 25 Percent Symbol % % 0 100 110 046 38 26 Ampersand ' ' 0 101 046 38 27 Single Quote; Apostrophe ((0 101 000 050 40 28 Left Parenthesis ((0 101 001 051 41 29 Right Parenthesis * * 0 101 015 41 29 Right Parenthesis * * 0 101 015 42 23 2B Plus Symbol * * 0 101 105 44 2C Comma * * 0 101 105 45 2D Minus Symbol; Slash; Virgule 0 0 101 100 064 48 30 Digit Zero	#	#	0 100 011	043	35	23	Number Symbol
% 0 100 101 045 37 25 Percent Symbol & & 0 100 110 046 38 26 Ampersand ' (0 100 111 047 39 27 Single Quote; Apostrophe ((0 101 001 051 41 29 Right Parenthesis * * 0 101 001 052 42 2A Asterisk + + 0 101 013 43 28 Plus Symbol . . 0 101 055 45 2D Minus Symbol; Hyphen . . 0 101 100 056 46 2E Period; Decimal Point / / 0 101 100 55 45 2D Minus Symbol; Hyphen . . 0 101 100 66 40 2E 2E Divide Symbol; Slash; Virgule 0 0 1010 061 49	ŝ	s	0 100 100	044	36	24	Dollar Symbol
& & 0 100 110 046 38 26 Ampersand ' 0 100 111 047 39 27 Single Quote; Apostrophe ((0 101 000 050 40 28 Left Parenthesis)) 0 101 001 051 41 29 Right Parenthesis * * 0 101 010 052 42 2A Asterisk + + 0 101 010 55 45 2D Minus Symbol , . 0 101 101 055 45 2D Minus Symbol; Hyphen . . 0 101 100 056 46 2E Period; Decimal Point / / 0 101 11 057 47 2F Divide Symbol; Slash; Virgule 0 0 0 1010 061 49 31 Digit Two 1 1 0 100 062 50	%	%	0 100 101	045	37	25	Percent Symbol
' 0 100 111 047 39 27 Single Quote; Apostrophe ((0 101 000 050 40 28 Left Parenthesis)) 0 101 001 051 41 29 Right Parenthesis * 0 101 010 052 42 2A Asterisk + + 0 101 010 53 43 2B Plus Symbol , . 0 101 055 45 2D Minus Symbol; Hyphen , . 0 101 1055 45 2D Pioid: Decimal Point / / 0 101 10 56 66 2E Period: Decimal Point / / 0 101 10 57 47 2F Divide Symbol; Slash; Virgule 0 0 0 101 000 66 48 30 Digit Zero 1 1 0 100 061 49 31 D	&	&	0 100 110	046	38	26	Ampersand
((0 101 000 050 40 28 Left Parenthesis)) 0 101 001 051 41 29 Right Parenthesis + + 0 101 011 052 42 2A Asterisk + + 0 101 011 053 43 2B Plus Symbol , 0 101 100 054 44 2C Comma - 0 101 101 055 45 2D Minus Symbol; Hyphen , . 0 101 101 056 46 2E Period; Decimal Point // / 0 101 100 056 46 2E Period; Decimal Point // / 0 101 001 064 43 Digit Zero 1 1 0 110 001 061 49 31 Digit Two 3 3 0 110 010 062 50 32 Digit Four 5 5 0 110 101 065 53 35 Digit Six 7 7 0 110 111 </td <td>'</td> <td>,</td> <td>0 100 111</td> <td>047</td> <td>39</td> <td>27</td> <td>Single Quote; Apostrophe</td>	'	,	0 100 111	047	39	27	Single Quote; Apostrophe
)) 0 101 001 051 41 29 Right Parenthesis * * 0 101 010 052 42 2A Asterisk + + 0 101 010 052 42 2A Asterisk + + 0 101 010 053 43 2B Plus Symbol . 0 101 101 055 45 2D Minus Symbol; Hyphen . . 0 101 110 056 46 2E Period; Decimal Point / / 0 101 111 057 47 2F Divide Symbol; Slash; Virgule 0 0 0 110 000 66 48 30 Digit Zero 1 1 0 110 001 62 50 32 Digit Two 3 3 0 110 010 62 53 35 Digit Four 5 5 0 110 101 66	((0 101 000	050	40	28	Left Parenthesis
* 0 101 010 052 42 2A Asterisk + + 0 101 011 053 43 2B Plus Symbol , 0 101 100 054 44 2C Comma - 0 101 101 055 45 2D Minus Symbol; Hyphen - 0 101 101 055 46 2E Period; Decimal Point // 0 101 111 057 47 2F Divide Symbol; Slash; Virgule 0 0 0 110 000 060 48 30 Digit Zero 1 1 0 110 001 062 50 32 Digit Two 3 3 0 110 063 51 33 Digit Four 5 5 0 110 106 65 53 35 Digit Six 7 7 0 110 110 066 54 36 Digit Six 7))	0 101 001	051	41	29	Right Parenthesis
++0101011053432BPlus Symbol.0101100054442CComma.0101101055452DMinus Symbol; Hyphen.0101110056462EPeriod; Decimal Point//0101111057472FDivide Symbol; Slash; Virgule0001100000604830Digit Zero1101100010614931Digit One2201100100625032Digit Two3301100100645234Digit Four5501101000665335Digit Six7701101100665436Digit Six7701101100665436Digit Six7701101110675537Digit Seven8801110000705638Digit Fight9901110010715739Digit Nine::0111001072583AColon;:01110075613DEqual Symbol; Left Caret==011110007		*	0 101 010	052	42	2A	Asterisk
. 0 101 100 054 444 2C Comma . 0 101 101 055 45 2D Minus Symbol; Hyphen . 0 101 110 056 46 2E Period; Decimal Point / / 0 101 110 056 46 2E Period; Decimal Point / / 0 101 110 057 47 2F Divide Symbol; Slash; Virgule 0 0 0 1010 000 060 48 30 Digit Zero 1 1 0 110 001 062 50 32 Digit Two 3 3 0 110 010 063 51 33 Digit Four 5 5 0 110 100 064 52 34 Digit Six 7 7 0 110 110 066 54 36 Digit Six 7 7 0 110 010 071 57 39 <td>+</td> <td>+</td> <td>0 101 011</td> <td>053</td> <td>43</td> <td>2B</td> <td>Plus Symbol</td>	+	+	0 101 011	053	43	2B	Plus Symbol
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	•	•	0 101 100	054	44	2C	Comma
.0101110056462EPeriod; Decimal Point/00101111057472FDivide Symbol; Slash; Virgule0001100000604830Digit Zero1101100010614931Digit One2201100100625032Digit Two3301100110635133Digit Four5501101000645234Digit Four5501101010655335Digit Six7701101110675537Digit Seven8801110010715739Digit Nine::0111010072583AColon::0111011073593BSemicolon::0111017613DEqual Symbol; Left Caret==0111101075613DEqual Symbol; Right Caret:::0111107633FQuestion Mark@@111110076623EGreater Than Symbol; Right Caret	-	-	0 101 101	055	45	2D	Minus Symbol; Hyphen
//0101111057472FDivide Symbol; Slash; Virgule0001100000604830Digit Zero1101100010614931Digit One2201100100625032Digit Two3301100110635133Digit Four5501101000645234Digit Four5501101010655335Digit Six7701101110675537Digit Six7701101110675537Digit Seven8801110000705638Digit Eight990111010072583AColon::0111010074603CLess Than Symbol; Left Caret==0111101075613DEqual Symbol>>0111110076623EGreater Than Symbol; Right Caret??0111111077633FQuestion Mark			0 101 110	056	46	2E	Period; Decimal Point
0001100000604830Digit Zero1101100010614931Digit One2201100100625032Digit Two3301100110635133Digit Four4401101000645234Digit Four5501101010655335Digit Six7701101110675537Digit Seven8801110000705638Digit Eight9901110010715739Digit Nine::0111010072583AColon;:0111101073593BSemicolon::0111107603CLess Than Symbol; Left Caret==0111101076623EGreater Than Symbol; Right Caret??0111111077633FQuestion Mark	/	/	0 101 111	057	47	2F	Divide Symbol; Slash; Virgule
1101100010614931Digit One2201100100625032Digit Two3301100110635133Digit Three4401101000645234Digit Four5501101010655335Digit Five6601101100665436Digit Six7701101110675537Digit Seven8801110000705638Digit Eight9901110110715739Digit Nine::0111010072583AColon;:0111010074603CLess Than Symbol; Left Caret==0111101075613DEqual Symbol>>0111110076623EGreater Than Symbol; Right Caret??0111111077633FQuestion Mark	0	0	0 110 000	060	48	30	Digit Zero
2201100100625032Digit Two3301100110635133Digit Three4401101000645234Digit Four5501101010655335Digit Five6601101100665436Digit Six7701101110675537Digit Seven8801110000705638Digit Fine9901110110715739Digit Nine::0111010072583AColon;:0111010074603CLess Than Symbol; Left Caret==0111101075613DEqual Symbol>>0111110076623EGreater Than Symbol; Right Caret??0111111077633FQuestion Mark	1	1	0 110 001	061	49	31	Digit One
3501100110635133Digit Fine4401101000645234Digit Four5501101010655335Digit Five6601101100665436Digit Six7701101110675537Digit Seven8801110000705638Digit Eight9901110100715739Digit Nine::0111010072583AColon;:0111011073593BSemicolon::0111101075613DEqual Symbol; Left Caret==0111101076623EGreater Than Symbol; Right Caret??0111111077633FQuestion Mark	2	2	0 110 010	062	50	32	Digit Two
4 4 0 110 100 004 52 34 Digit Five 5 5 0 110 101 065 53 35 Digit Five 6 6 0 110 110 066 54 36 Digit Six 7 7 0 110 111 067 55 37 Digit Seven 8 8 0 111 000 070 56 38 Digit Five 9 9 0 111 001 071 57 39 Digit Nine : : 0 111 010 072 58 3A Colon ; : 0 111 010 074 60 3C Less Than Symbol; Left Caret = = 0 111 101 075 61 3D Equal Symbol > > 0 111 101 076 62 3E Greater Than Symbol; Right Caret ? : 0 111 110 <t< td=""><td>3</td><td>3</td><td>0 110 011</td><td>063</td><td>52</td><td>33</td><td>Digit Three Digit Four</td></t<>	3	3	0 110 011	063	52	33	Digit Three Digit Four
55610101005555555101106601101100665436Digit Six7701101110675537Digit Seven8801110000705638Digit Eight9901110100715739Digit Nine::0111010072583AColon;:0111011073593BSemicolon<	4	4	0 110 101	065	53	35	Digit Five
0 0 0 110 100 067 55 37 Digit Seven 7 7 0 110 111 067 55 37 Digit Seven 8 8 0 111 000 070 56 38 Digit Eight 9 9 0 111 001 071 57 39 Digit Nine $:$ $:$ 0 111 010 072 58 $3A$ Colon $:$ $:$ 0 111 010 072 58 $3A$ Colon $:$ $:$ 0 111 010 072 58 $3A$ Colon $:$ $:$ 0 111 010 074 60 $3C$ Less Than Symbol; Left Caret $=$ $=$ 0 111 100 076 62 $3E$ Greater Than Symbol; Right Caret $?$ 0 111 110 077 63 $3F$	5	6	0 110 110	066	54	36	Digit Six
8 8 0 111 000 070 56 38 Digit Eight 9 9 0 111 001 071 57 39 Digit Eight 9 9 0 111 001 071 57 39 Digit Nine : : 0 111 010 072 58 3A Colon : : 0 111 010 072 58 3A Colon : : 0 111 011 073 59 3B Semicolon : 0 111 100 074 60 3C Less Than Symbol; Left Caret = = 0 111 101 075 61 3D Equal Symbol > > 0 111 110 076 62 3E Greater Than Symbol; Right Caret ? : 0 111 111 077 63 3F Question Mark @ : : : 0:000	7	7	0 110 111	067	55	37	Digit Seven
9901110010715739Digit Nine9 0 111010072583AColon $:$ $:$ 0 111010072583AColon $:$ $:$ 0 111011073593BSemicolon $<$	8	8	0 111 000	070	56	38	Digit Eight
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	9	9	0 111 001	071	57	39	Digit Nine
;;0111011073593BSemicolon $<$ <	:		0 111 010	072	58	3A	Colon
<	;	;	0 111 011	073	59	3B	Semicolon
= = 0 111 101 075 61 3D Equal Symbol > > 0 111 110 076 62 3E Greater Than Symbol; Right Caret ? ? 0 111 111 077 63 3F Question Mark @ @ 1 000 000 100 64 40 At Sign	<	<	0 111 100	074	60	3C	Less Than Symbol: Left Caret
> > 0 111 110 076 62 3E Greater Than Symbol; Right Caret ? 0 111 111 077 63 3F Question Mark @ @ 1 000 000 100 64 40 At Sign	=		0 111 101	075	61	3D	Equal Symbol
? ? 0 111 111 077 63 3F Question Mark @ @ 1 000 000 100 64 40 At Sign	>	>	0 111 110	076	62	3E	Greater Than Symbol; Right Caret
@ 1 000 000 100 64 40 At Sign	?	?	0 111 111	077	63	3F	Question Mark
	@	@	1 000 000	100	64	40	At Sign

¹⁶ Perry Cole J. W., ANSI Fortran 77, A Structured Problem-Solving Approach, Wm. C. Brown Publishers, Dubuque, Iowa, 1987.

А	А	1 000 001	101	65	41	Uppercase Letter A
В	В	1 000 010	102	66	42	Uppercase Letter B
C	С	1 000 011	103	67	43	Uppercase Letter C
D	D	1 000 100	104	68	44	Uppercase Letter D
E	E	1 000 101	105	69	45	Uppercase Letter E
F	F	1 000 110	106	70	46	Uppercase Letter F
G	G	1 000 111	107	71	47	Uppercase Letter G
Н	Н	1 001 000	110	72	48	Uppercase Letter H
I	Ι	1 001 001	111	73	49	Uppercase Letter I
J	l	1 001 010	112	74	4A	Uppercase Letter J
K	K	1 001 011	113	75	4B	Uppercase Letter K
L	L	1 001 100	114	76	4C	Uppercase Letter L
M	М	1 001 101	115	77	4D	Uppercase Letter M
N	N	1 001 110	116	78	4E	Uppercase Letter N
0	0	1 001 111	117	79	4F	Uppercase Letter O
Р	Р	1 010 000	120	80	50	Uppercase Letter P
0	0	1 010 001	121	81	51	Uppercase Letter Q
Ř	Ř	1 010 010	122	82	52	Uppercase Letter R
S	S	1 010 011	123	83	53	Uppercase Letter S
Ť	Т	1 010 100	124	84	54	Uppercase Letter T
Û	Û	1 010 101	125	85	55	Uppercase Letter U
v	v	1 010 110	126	86	56	Uppercase Letter V
w	W	1 010 111	127	87	57	Uppercase Letter W
x	x	1 011 000	130	88	58	Uppercase Letter X
Y	Ŷ	1 011 001	131	89	59	Uppercase Letter Y
7	7	1 011 010	132	90	5A	Uppercase Letter Z
ſ	ĩ	1 011 011	133	91	5B	Left Bracket
ł	L \	1 011 100	134	92	5C	Back Slash: Back Slant
ì	ì	1 011 101	135	93	5D	Right Bracket
i	ĭ	1 011 110	136	94	5E	Circumflex: 1 up arrow
		1 011 111	137	95	5F	Underscore: Underline
		1 100 000	140	. 96	60	Back Quote: Grave Accent
	2	1 100 001	141	97	61	Lowercase Letter a
a	h	1 100 010	147	98	62	Lowercase Letter b
0	0	1 100 011	143	99	63	Lowercase Letter c
d	d	1 100 100	144	100	64	Lowercase Letter d
u	e	1 100 101	145	101	65	Lowercase Letter e
ſ	f	1 100 110	146	102	66	Lowercase Letter f
1	1	1 100 111	140	102	67	Lowercase Letter 9
8	B h	1 101 000	150	104	68	Lowercase Letter h
	:	1 101 001	151	105	69	Lowercase Letter i
		1 101 010	152	106	64	Lowercase Letter i
J	J	1 101 011	152	107	6B	Lowercase Letter k
K I	1	1 101 100	154	108	60	Lowercase Letter
	1	1 101 101	155	100	60	Lowercase Letter m
m	m	1 101 101	155	110	6E	Lowercase Letter n
n	n		150	111	6E	Lowercase Letter o
0	0		160	112	70	Lowercase Letter p
р	р	1 110 000	161	112	70	Lowercase Letter o
q	q	1 110 001	167	113	72	Lowercase Letter r
r	I		162	115	72	Lowercase Letter s
S	s	1 110 011	164	116	74	Lowercase Letter t
t	l .	1 110 100	165	117	75	Lowercase Letter u
u	u	1 110 101	165	110	76	Lowercase Letter u
V	v	1 110 110	167	110	77	Lowercase Letter w
w	w		107	119	79	Lowercase Letter v
x	x	1 111 000	170	120	70	Lowercase Letter x
y	У		171	121	74	Lowercase Letter y
Z	Z		172	122	70	Lowercase Letter Z
1			173	123	70	Vertical Part Logical OP
			174	124	70	Pight Broom
1	1		175	125	70	Tilde
	DEI		170	120	76	Delete: Pub cut
DEL	DEL	1 111 111	1//	127	71	Delete, Ruo out

Priloga B: VMS operacijski sistem

VMS operacijski sistem omogoča izvajanje programov in samo delo z različnimi programi. Vedeti moramo:

- kako se priključiti na sistem,
- kako uporabljati sistem,
- kako se izključiti iz sistema,
- kako uporabljati datoteke,
- kako kreirati in popravljati datoteke z editorjem,
- kako brisati in izpisovati datoteke,
- kako vnašati fortranske programe,
- kako vnašati komande za izvajanje fortranskega programa.

B1 Priključitev na univerzitetni računalnik

Najprej se prijavimo v lokalno omrežje:

- vtipkamo uporabniško ime in geslo (dobite ga pri asistentu oziroma predavatelju),
- aktiviramo program Putty (preko katerega se bomo priključili na univerzitetni računalnik),
- v gornje okno (Host name) zapišemo: rcum.uni-mb.si ter med opcijami izberemo Telnet,
- pritisnemo gumb **OPEN**. Na ekranu se pojavi napis:

USERNAME: vtipkamo username in pritisnemo [ENTER]

Pojavi se napis:

PASSWORD: vtipkamo password (se ne vidi na ekranu) in pritisnemo [ENTER].

Pojavi se:

\$

Sedaj je sistem pripravljen za delo. Vnašamo potrebne komande.

B2 Uporaba sistema

Za prikaz časa in datuma vtipkamo:

\$ SHOW TIME [ENTER]

Če med vnašanjem napravimo napako, uporabimo za brisanje tipko [BACKSPACE]. Če želimo uporabiti gornje znake na tipkah tipkovnice, uporabimo tipko [SHIFT].

Včasih uporabimo za izvedbo neke komande dve tipki hkrati, ponavadi tipko [CTRL] in poljubno drugo tipko. Pri takšni izvedbi neke komande držimo tipko [CTRL] nato pa pritisnemo še drugo tipko. Npr. za brisanje cele vtipkane vrstice uporabimo tipki [CTRL] in [U] ali krajše [CTRL/U].

Za prekinitev izvajanja programa uporabimo tipki [CTRL/Y].

Za zaustavitev izpisa na ekranu uporabimo tipki [CTRL/S], za deaktivacijo pa tipki [CTRL/Q].

B3 lzključitev iz sistema

Pri dolarskem znaku vpišemo:

\$ LO [ENTER]

(logging out)

B4 Kreiranje datotek z EDitorjem

a) Datoteka je:

- zbirka podatkov,
- zaporedje nekih informacij, ki jih shranimo (npr. nek fortranski program),
- logični zapis po pravilih.

Poimenujemo jih lahko na različne načine: IME.DAT, IME.FOR, IME.REZ......

b) Editor je neke vrste urejevalnik, ki vsebuje zaporedje komand, ki jih računalnik razpozna. Z editorjem lahko:

- vnašamo podatke v datoteko,
- spreminjamo podatke v datoteki,
- dodajamo podatke v datoteko,
- shranimo datoteko

Mi uporabljamo ED editor. Pokličemo ga na naslednji način:

\$ ED TESTX.DAT [ENTER] ime datoteke tip datoteke

Pri imenu datoteke moramo obvezno navesti tip (zapisan za piko). Po klicu editorja se odpre vsebina datoteke. Na dnu se pojavi napis [EOB], ki pomeni konec delovnega prostora.

B5 Ostale pomembne komande VMS sistema

\$ FOR IME

- prevajanje fortranskega programa.

ime datoteke brez navedbe tipa, če je program napisan na datoteko tipa FOR

- **\$ LINK IME** povezovanje programa.
- **\$ RUN IME** izvajanje programa.

Omenjene komande so pomembne pri zagonu programa. Po prevajanju se tvori datoteka tipa **.OBJ** in po povezovanju tipa **.EXE**.

Priloge

\$ TYPE IME.FOR	- izpis programa na ekranu.
\$ COPY IME.FOR VAJA.FOR	 napravimo kopijo programa. Prvotna datoteka IME.FOR ostane nespremenjena.
\$ REN IME.FOR VAJA1.FOR	- preimenovanje datoteke IME.FOR v VAJA1. FOR.
\$ DIR	- pregledamo seznam datotek na direktoriju.
\$ DELETE IME.FOR;1	- brisanje datoteke.
\$ PURGE IME.*	- brisanje starih datotek razen zadnje.

Priloga C: Funkcijske tipke

Tipke se nahajajo na desni strani tipkovnice in jih uporabljamo pri delu z editorjem.

			Num lock	1	*	_
			7	8	9	
			4	5	6	+
	ſ		1	2	3	ENTER
¢	Ų	⇒	(D	•	



			GOLD	HELP	FNDNXT FIND	DEL L UND L
			PAGE COMMAND	SECT FILL	APPEND REPLACE	DEL C
			ADVANCE BOTTOM	BACKUP TOP	CUT PASTE	UND C
	UP		WORD CHNGCASE	EOL DEL EOL	CHAR SPECINS	ENTER
LEFT	DOWN	RIGHT	LII OPEN	NE I LINE	SELECT RESET	SUBS



Pomen in uporaba nekaterih funkcijskih tipk:

[GOLD]	- uporablja se podobno kot tipka SHIFT na običajni tipkovnici (za uporabo ' spodnjih ' funkcij).
[HELP]	 uporablja se za izpis funkcijskih tipk na ekranu (razlaga pomena tipk).
[FNDNXT] ali [FINDNEXT]	- uporablja se za iskanje naslednje besede, ki smo jo definirali s funkcijo FIND.
[DEL L] ali [DEL LINE]	- brišemo eno vrsto za kurzorjem.
[ADVANCE]	- uporablja se za nadalnji pregled datoteke.
[BACKUP]	- uporablja se za povratni pregled datoteke.
[DEL C] ali [DEL CHAR]	- brišemo znak na katerem stoji kurzor.
[WORD]	 uporablja se premik preko ene besede ali stavka do nasledje praznine: naprej, če smo v stanju ADVANCE, nazaj, če smo v stanju BACKUP.
[EOL]	 kurzor se prestavi na konec najbližje vrstice: naprej, če smo v stanju ADVANCE, nazaj, če smo v stanju BACKUP.
[CHAR]	 kurzor premaknemo za en znak: naprej, če smo v stanju ADVANCE, nazaj, če smo v stanju BACKUP.
[LINE]	 kurzor pomaknemo na začetek naslednje vrstice.
[ENTER]	- uporabljamo ga pri delu z editorjem.
[GOLD]+[FIND]	 uporablja se za iskanje znakov ali besed v programu: naprej, če smo v stanju ADVANCE, nazaj, če smo v stanju BACKUP.
[GOLD]+[UND L]	- uporablja se za povratek nazadnje brisane vrstice.
[GOLD]+[COMMAND]	 uporablja se za izhod iz datoteke oz. editorja. Pri besedi Command: na dnu ekrana vtipkamo: EXIT, če želimo shraniti zadnjo verzijo programa, QUIT, če zadnje verzije ne želimo shraniti.
[GOLD]+[COMMAND]	 se uporablja tudi za vnos drugega programa v naš program (z druge datoteke). Pri besedi Command: vtipkamo INCL ime.tip datoteke. Pri tem moramo paziti, kje v našem programu stoji kurzor, saj se priklicani program izpiše na mestu, kjer stoji kurzor.

Priloge					
[GOLD]+[COMMAND]	 če pri besedi Command: napišemo številko vrstice, se kurzor postavi v željeno vrstico. 				
[GOLD]+[UND W]	- uporablja se za povratek nazadnje brisane besede.				
[GOLD]+[BOTTOM]	- uporablja se za pomik na konec datoteke oz. programa.				
[GOLD]+[TOP]	 uporablja se za pomik kurzorja na začetek datoteke oz. programa. 				
[GOLD]+[PASTE]	 uporablja se za povratek brisanega dela programa na isto ali drugo mesto v datoteki. 				
[GOLD]+[UND C]	- uporablja se za povratek nazadnje brisanega znaka.				
[GOLD]+[DEL EOL]	 brišemo vse znake, ki so v vrstici desno od kurzorja vključno z znakom na katerem je kurzor. 				
[GOLD]+[OPEN LINE]	- uporablja se za odpiranje prazne vrstice.				
Uporaba kombinacije funkcijskih tipk [SELECT]+[LINE]+[CUT]:					
[SELECT]	 s tipko izberemo del programa, ki ga želimo brisati ali prenesti na drugo mesto v datoteki. Postavimo se na začetek prve vrstice tega dela programa in pritisnemo tipko [SELECT]. 				
[LINE]	- s pritiskanjem tipke [LINE] označimo želeni del programa.				

```
[CUT] - s pritiskom tipke [CUT] brišemo označen del programa.
```
Priloga D: Rešitve problemov za 1. del

1. poglavje

- 1.1. $8_{(10)} = 1000_{(2)}$ $15_{(10)} = 1111_{(2)}$
- $24_{(10)} = 11000_{(2)}$
- 1.2. $3,25_{(10)} = 11,01_{(2)}$ $12,4_{(10)} = 1100,0110011....(2)$ $2,475_{(10)} = 10,01111001...(2)$
- 1.3. $110,111_{(2)} = 6,875_{(10)}$

2. poglavje

- 2.1. I, R, I, R R, R, I, I
- 2.2. da ne (realna spremenljivka) da ne (operator v imenu ni dovoljen) ne (prvi znak mora biti črka) ne (samo črke in številke so dovoljene v imenu)
- 2.3. da ne (cela spremenljivka) ne (cela spremenljivka) ne (prvi znak mora biti črka) ne (privzeta FORTRANska beseda) ne (pomišljaj ni dovoljen v imenu)
- 2.4. a) (X+Y)*(U+V)
 - b) 3.0*(X+Y)
 - c) 3.0*X*Y**2.0-2.0*X**2.0*Y
 - d) A+B/C**2
 - e) (A+B)/(C*D)
- 2.5. pravilno nepravilno pravilno nepravilno pravilno pravilno

2.6.	a)	X=18.2	b)	X= -6.0	c)	X=0.0
		L=18		L= -7		L=0

3. poglavje

- 3.1. a) SQRT(5.*X**2+8.*Y**2)
 - b) SIN(X-2.*Y)+EXP(X*Y)-ABS(X**2-Y**2)
 - c) EXP(ABS(A))-B**2/ABS(C)
 - d) LOG((X+Y)**2)
 - e) LOG10((A-B)**2)
- 3.2. Sami napišite program! (Rezultat: r = 5)
- 3.3. Sami napišite program! (Rezultat: $\cos 60^{\circ} = 0.5$; $\sin 30^{\circ} = 0.5$)

4. poglavje

4.1. Vnos podatkov preko terminala (ekrana). Izhod rezultatov na terminal (ekran).

Vnos podatkov preko datoteke RH.DAT Izhod rezultatov na terminal (ekran).

Vnos podatkov preko datoteke RH1.DAT Izhod rezultatov na datoteko PLAC.REZ

- 4.2. STATUS='OLD'
- 4.3. **STATUS='NEW'**
- 4.4. Sami preoblikujte program!

6. poglavje

6.1.

6

DO 10 I=1,10 WRITE (6,*) 'Peter Hren' 10 CONTINUE END

.2.	1.	program	2. program	3. program	4. program
	1	1	3 0	STOLETJE: 1	6 16
	2	3	4 1	STOLETJE: 2	9 19
	3	6	52	STOLETJE: 3	12 22
			63		15 25
			74		18 28
			85	STOLETJE: 20	21 31
					24 34
					175

- 6.3. Sami napišite program!
- 6.4. Sami napišite program!
- 6.5. Sami napišite program!

7. poglavje

7.1.

- a) Za Z ni vejice. Pod Z shranjujemo realno število in ga ne moremo čitati z I15.
- b) Pod J shranjujemo celo število, zato ne moremo uporabiti formatno določilo F.

7.2.

- a) Nekateri prevajalniki pred A ne dopuščajo vejice. B, kot realna spremenljivka, ne more biti izpisan z določilom I.
- b) K, kot cela spremenljivka, ne more biti izpisan z določilom F.

7.3.

- a) Rezerviranih je 10 mest, vendar je izpis v prvih devetih stolpcih desno poravnan. Prvi stolpec ali mesto je rezervirano za izpisovanje v naslednjo vrstico.
- b) Prvi X je rezerviran za izpisovanje v naslednjo vrstico, sledijo tri praznine in 8 stolpcev. Število je desno poravnano.
- c) Prvi X je rezerviran za izpisovanje, sledijo tri praznine in 3 stolpci. Število je desno poravnano, vendar je prostora premalo za izpis danega števila.
- d) Na začetku naslednje strani bo izpisano število 2345. Enka na začetku števila pomeni prehod na novo stran.





7.4. Rezultat prikazujejo slike!



7.5.

- a) Prva praznina je za prehod v novo vrsto, sledi 5 praznin, nato tekst OSEBNA STEVILKA. Sledita dve praznini in v naslednjih osmih stolpcih desno poravnano število.
- b) Prva praznina je za prehod v novo vrstico, sledi 5 praznin, nato tekst TEDENSKA PLACA V USD. V naslednjih osmih stolpcih sledi desno poravnan izpis realnega števila.

7.6.

- 0.123E+03 0.123E+02 0.563E-07 0.347E-03 -0.356E+00 0.3E+14
- 7.7. Sami napišite program!

8. poglavje

8.1. equal to less than or equal to not equal less than

greater than or equal to greater than

8.2. .NOT. .OR. .AND.

- 8.4. X je vecji Dodatek

AND je pravilen OR je pravilen

8 9 10

- 10
- 8.5. Sami napišite program!

9. poglavje

- 9.1. REAL, CHARACTER in INTEGER
- 9.2. 25, cela števila
 50, realna števila
 8, znakovni zapis (po 15 znakov v vsako lokacijo)
- 9.3. 111.0000 110.0000

9.4.	1	1
	2	4
	3	9
	4	16
	5	25
	6	36

9.5.	1	50	70	60
	2	60	80	70
	3	70	90	80

9.6. Sami napišite program!

<u>10. poglavje</u>

10.1.

Podprogram FUNCTION:

30

10

55.00000

10.2.

Podprogram SUBROUTINE:

Z = 156.0600

A = 4.666667

10.3. Sami napišite program.

<u>11. poglavje</u>

- 11.1. F = 3.500000
- 11.2. A = 2.4B = 2.4C = 1.5
- 11.3. Sami napišite program!