

OSNOVI PROGRAMIRANJA

(II)

PREDSTAVLJANJE PODATAKA U RAČUNARU

mr Dražen Brđanin

Elektrotehnički fakultet Banja Luka

2007.

2.1. Predstavljanje u računar

INFORMACIONE
JEDINICE ?

Osnovne informacione jedinice

bit (b) nosilac najmanje količine informacije
nosilac elementarne (binarne) informacije

1

0

veće jedinice:

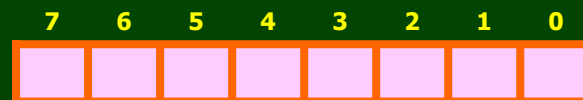
kilobit - kb = 2^{10} b = 1024 bita

megabit - Mb = 2^{20} b = 1024 kb = 1.048.576 bita

gigabit - Gb = 2^{30} b = 1024 Mb = 1.073.741.824 bita

bajt (B)

1 B = 8 b



veće jedinice:

kilobajt - kB = 2^{10} B = 1024 bajta

megabajt - MB = 2^{20} B = 1024 kB = 1.048.576 bajta

gigabajt - GB = 2^{30} B = 1024 MB = 1.073.741.824 bajta

2.1. Predstavljanje u računar

**CJELOBROJNI
PODACI ?**

Cjelobrojni podaci (INTEGER)

neoznačeni cijeli brojevi (unsigned integer)

cijeli brojevi bez predznaka (pozitivni + nula)

označeni cijeli brojevi (signed integer)

cijeli brojevi sa predznakom (pozitivni + negativni + nula)

U računar se koriste sljedeći formati

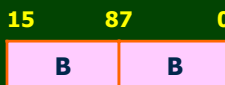
bajt

(byte) - B



riječ

(word) – W=2B



dvostruka riječ

(doubleword) – D=4B



čtetvorostruka riječ

(quadword) – Q=8B



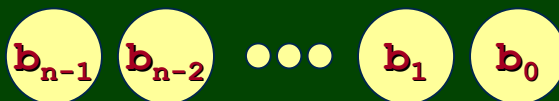
2.2. Neoznačeni cijeli brojevi

UNSIGNED
INTEGER ?

Neoznačeni cjelobrojni podaci (unsigned integer)

cijeli brojevi bez predznaka (pozitivni + nula)

niz od 8, 16, 32 ili 64 bita



vrijednost

$$V = b_{n-1} \cdot 2^{n-1} + \dots + b_1 \cdot 2^1 + b_0 \cdot 2^0$$

primjer:

bajt

7	6	5	4	3	2	1	0
1	0	0	0	0	1	0	0

$$2^7 + 2^2 = 128 + 4 = 132$$

riječ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1

$$2^{10} + 2^2 + 2^0 = 1024 + 4 + 1 = 1029$$

2.2. Neoznačeni cijeli brojevi

UNSIGNED
INTEGER ?

Neoznačeni cjelobrojni podaci (unsigned integer)

byte

minimalna vrijednost

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0

$$V_{\min}=0$$

maksimalna vrijednost

7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1

$$V_{\max} = 255 = 256 - 1 = 2^8 - 1$$

word

minimalna vrijednost

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$$V_{\min}=0$$

maksimalna vrijednost

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

$$V_{\max} = 65535 = 65536 - 1 = 2^{16} - 1$$

$$0 \leq v \leq 2^n - 1$$

2.2. Neoznačeni cijeli brojevi

UNSIGNED
INTEGER ?

Opseg vrijednosti neoznačenih cjelobrojnih podataka

$$0 \leq V \leq 2^{n-1}$$

format		opseg vrijednosti	
byte	(B = 8b)	$0 - 2^8 - 1$	0 - 255
word	(W = 16b)	$0 - 2^{16} - 1$	0 - 65535
doubleword*	(D = 32b)	$0 - 2^{32} - 1$	0 - 4294967295
quadword	(Q = 64b)	$0 - 2^{64} - 1$	0 - ???

* doubleword = longword

2.2. Neoznačeni cijeli brojevi

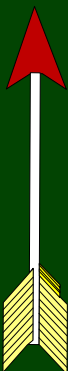
UNSIGNED
INTEGER ?

Primjer:

Predstaviti broj 19 kao neoznačeni cjelobrojni podatak.

19 : 2

9	1
4	1
2	0
1	0
0	1



$$19_{10} = 10011_2$$

byte

7	6	5	4	3	2	1	0
0	0	0	1	0	0	1	1

word

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

2.2. Neoznačeni cijeli brojevi

UNSIGNED
INTEGER ?

Primjer:

Sljedeći niz cijelih brojeva 1, 19, 128, 255, -10 smjestiti u memoriju kao neoznačene cjelobrojne podatke tipa byte počevši od lokacije 10000.

-10

255

128

19

1

Memorija
(binarni sadržaj)

7 6 5 4 3 2 1 0

1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0
0	0	0	1	0	0	1	1
0	0	0	0	0	0	0	1

Memorija
(heks. sadržaj)

10006

10005

10004

10003

10002

10001

10000

09999

F	F
8	0
1	3
0	1

2.2. Neoznačeni cijeli brojevi

UNSIGNED
INTEGER ?

Primjer:

Sljedeći niz cijelih brojeva 1, 255, 1024 smjestiti u memoriju kao neoznačene cjelobrojne podatke tipa word počevši od lokacije 1000.

Memorija
(binarni sadržaj)

	7	6	5	4	3	2	1	0
1006								
1005	0	0	0	0	0	1	0	0
1004	0	0	0	0	0	0	0	0
1003	0	0	0	0	0	0	0	0
1002	1	1	1	1	1	1	1	1
1001	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	1
0999								

1024

0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0

255

0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1

1

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

2.2. Neoznačeni cijeli brojevi

ZADACI ZA VJEŽBU...

1. Broj 123 predstaviti kao neoznačeni cjelobrojni podatak tipa:
a) byte, b) word, c) doubleword, d) quadword.
2. Na slici je prikazan sadržaj jednog segmenta memorije. Odrediti dekadsku vrijednost neoznačenog cjelobrojnog podatka upisanog na lokaciji 0F000h, ako se on posmatra kao:
a) byte, b) word.

0	0	1	1	1	0	0	0	0F002h
1	0	1	0	0	0	0	1	0F001h
0	0	0	0	0	0	0	0	0F000h
1	0	0	0	0	0	0	0	0EFFFh

3. Odrediti koji se niz neoznačenih cjelobrojnih jednobajtnih podataka nalazi u memoriji, ako je sadržaj tog segmenta prikazan na prethodnoj slici.
4. Zadat je niz cjelobrojnih podataka koje treba smjestiti u memoriju kao neoznačene podatke tipa word počevši od lokacije 00100h redom: 32, 5, 1024, -1, 256. Prikazati sadržaj memorije binarno, odnosno heksadecimalno.

2.3. Označeni cijeli brojevi

SIGNED INTEGER ?

Označeni cjelobrojni podaci (signed integer)

cijeli brojevi sa predznakom (pozitivni + nula + negativni)

niz od 8, 16, 32 ili 64 bita

bit znaka
0 = pozitivan
1 = negativan

b_{n-1} b_{n-2} ... b_1 b_0

vrijednost

$$v = b_0 \cdot 2^0 + \dots + b_{n-2} \cdot 2^{n-2} - b_{n-1} \cdot 2^{n-1}$$

primjer:

bajt

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1

$$2^0 = 1$$

bajt

7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	1

$$2^0 - 2^7 = 1 - 128 = -127$$

2.3. Označeni cijeli brojevi

SIGNED INTEGER ?

Označeni cjelobrojni podaci (signed integer) - byte

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0

$V = 0$

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1

$V = 1$

7	6	5	4	3	2	1	0
0	1	1	1	1	1	1	1

$V = 127$

7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0

$V = -128$

7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	1

$V = 1 - 128 = -127$

7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1

$V = 127 - 128 = -1$

$-128 \leq V \leq 127$

2.3. Označeni cijeli brojevi

SIGNED INTEGER ?

Označeni cjelobrojni podaci (signed integer) - word

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

V = 0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

V = 1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

V = 32767

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

V = -32768

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

V = -1

$$-32768 \leq V \leq 32767$$

2.3. Označeni cijeli brojevi

SIGNED INTEGER ?

Opseg vrijednosti označenih cjelobrojnih podataka

$$-2^{n-1} \leq V \leq 2^{n-1}-1$$

format		opseg vrijednosti	
byte	(B = 8b)	$-2^7 \dots 2^7-1$	-128 .. 127
word	(W = 16b)	$-2^{15} \dots 2^{15}-1$	-32768 .. 32767
doubleword	(D = 32b)	$-2^{31} \dots 2^{31}-1$	-2147483648 .. 2147483647
quadword	(Q = 64b)	$-2^{63} \dots 2^{63}-1$???

2.3. Označeni cijeli brojevi

DRUGI
KOMPLEMENT ?

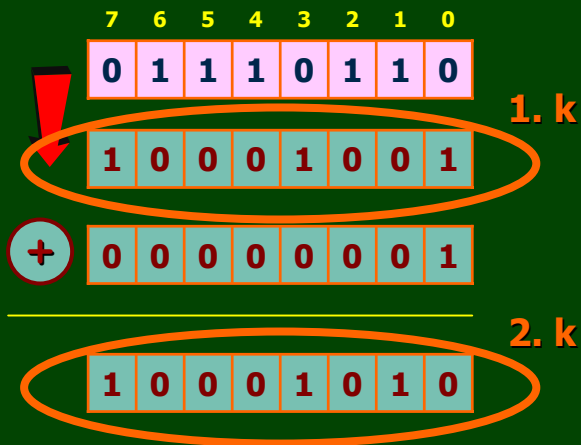
Predstavljanje negativnih cijelih brojeva

Koristi se tehnika komplementiranja za predstavljanje negativnih brojeva

1. Nepotpuno komplementiranje / prvi komplement /



2. Potpuno komplementiranje / drugi komplement /



2.3. Označeni cijeli brojevi

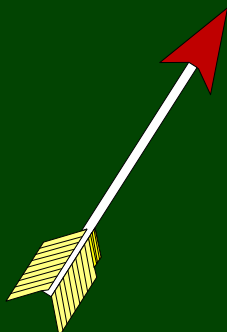
**DRUGI
KOMPLEMENT ?**

Primjer:

Predstaviti broj -19 kao označeni cjelobrojni podatak tipa byte.

19 : 2

9	1
4	1
2	0
1	0
0	1



byte

7	6	5	4	3	2	1	0	
0	0	0	1	0	0	1	1	19
1	1	1	0	1	1	0	0	1. k
0	0	0	0	0	0	0	1	+1
1	1	1	0	1	1	0	1	-19

2.3. Označeni cijeli brojevi

DRUGI
KOMPLEMENT ?

Primjer:

Predstaviti broj -1 kao označeni cjelobrojni podatak.

byte

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1

1

1	1	1	1	1	1	1	0
---	---	---	---	---	---	---	---

1. k

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

+1

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

-1

word

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

2.3. Označeni cijeli brojevi

DRUGI KOMPLEMENT ?

Primjer:

Sljedeći niz cijelih brojeva 1, -1, 127, -128, 255 smjestiti u memoriju kao označene cjelobrojne podatke tipa byte počevši od lokacije 100h.

255

-128

127

-1

1

**Memorija
(binarni sadržaj)**

7 6 5 4 3 2 1 0

1	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	1

**Memorija
(heks. sadržaj)**

106h

105h

104h

103h

102h

101h

100h

0FFh

8	0
7	F
F	F
0	1

2.3. Označeni cijeli brojevi

DRUGI KOMPLEMENT ?

Primjer:

Sljedeći niz cijelih brojeva 1, 255, -1 smjestiti u memoriju kao označene cjelobrojne podatke tipa word počevši od lokacije 1000.

**Memorija
(binarni sadržaj)**

	7	6	5	4	3	2	1	0
1006								
1005	1	1	1	1	1	1	1	1
1004	1	1	1	1	1	1	1	1
1003	0	0	0	0	0	0	0	0
1002	1	1	1	1	1	1	1	1
1001	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	1
0999								

← -1

← 255

← 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

2.3. Označeni cijeli brojevi

ZADACI ZA VJEŽBU...

1. Brojeve 123 i -123 predstaviti kao označene cjelobrojne podatke tipa:
a) byte, b) word, c) doubleword, d) quadword.
2. Na slici je prikazan sadržaj jednog segmenta memorije. Odrediti dekadsku vrijednost označenog cjelobrojnog podatka upisanog na lokaciji 0F000h, ako se on posmatra kao:
a) byte, b) word.

0	0	1	1	1	0	0	0	0F002h
1	0	1	0	0	0	0	1	0F001h
0	0	0	0	0	0	0	0	0F000h
1	0	0	0	0	0	0	0	0EFFFh

3. Odrediti koji se niz označenih cjelobrojnih jednobajtnih podataka nalazi u memoriji, ako je sadržaj tog segmenta prikazan na prethodnoj slici.
4. Zadat je niz cjelobrojnih podataka koje treba smjestiti u memoriju kao označene podatke tipa word počevši od lokacije 00100h redom: 32, -5, -1024, -1, 256. Prikazati sadržaj memorije binarno, odnosno heksadecimalno.

2.4. Znakovni podaci

ZNAKOVI ?

Predstavljanje znakova (karaktera)

Računar raspolaže odgovarajućim skupom znakova:

- upravljački znakovi – npr. za upravljanje štampačem i sl.
- slova, cifre, znakovi interpunkcije, grafički simboli

Znakovi se kodiraju neoznačenim cjelobrojnim vrijednostima

ako bi se koristio jedan bit

1

A

0

B

moguće kodirati samo 2 znaka (npr. A i B)

ako bi se koristila dva bita moguće kodirati 4 znaka

0 0

A

0 1

B

1 0

C

1 1

D



2.4. Znakovni podaci

ZNAKOVI ?

Predstavljanje znaka (karaktera)

koriste se 6-bitni, 7-bitni, 8-bitni i 16-bitni kodovi

- **6-bitni kodovi**

maksimalno $2^6 = 64$ znaka (npr. 26 slova, 10 cifara i 28 drugih)

- **7-bitni kodovi**

maksimalno $2^7 = 128$ znakova

najpoznatiji ASCII (American Standard Code for Information Interchange)

- **8-bitni kodovi**

maksimalno $2^8 = 256$ znakova

EBCDIC, prošireni ASCII

- **16-bitni kodovi**

maksimalno $2^{16} = 65536$ znakova

UNICODE (Windows)

2.4. Znakovni podaci

ZNAKOVI ?

ASCII kod

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00								♪								
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30		1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

4

3

4

0

0

1

1

0

1

0

0

A

4

1

0

1

0

0

0

0

0

0

1

2.4. Znakovni podaci

ZNAKOVI ?

Primjer:

Prikazati sadržaj memorije ako je, počevši od lokacije 1000, upisan string (niz znakova): **1+Asm**

Memorija
(binarni sadržaj)

	7	6	5	4	3	2	1	0	
1006									
1005									
1004	0	1	1	0	1	1	0	1	← m
1003	0	1	1	1	0	0	1	1	← s
1002	0	1	0	0	0	0	0	1	← A
1001	0	0	1	0	1	0	1	1	← +
1000	0	0	1	1	0	0	0	1	← 1
0999									

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00								♪								
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

2.4. Znakovni podaci

ZADACI ZA VJEŽBU...

1. U memoriji je počev od adrese 0400h upisan niz znakova. Ispisati prvih pet znakova teksta.

2	0	00406h
6	2	00405h
6	F	00404h
4	4	00403h
2	C	00402h
6	C	00401h
5	3	00400h
3	0	003FFh

		00506h
		00505h
		00504h
		00503h
		00502h
		00501h
		00500h
		004FFh

2. Upisati u memoriju (počevši od lokacije 500h) riječ **Doboj**.

2.5. Brojevi u fiksnom zarezu

FIKSNI ZAREZ ?

Predstavljanje brojeva u fiksnom zarezu

Brojevi u fiksnom zarezu imaju najširu primjenu u administraciji

Primjenjuje se cjelobrojna aritmetika, pri čemu se ima u vidu položaj binarne (decimalne) tačke

Najčešće se primjenjuje BCD kodiranje (Binary Coded Decimal)

- nepakovani BCD podaci
- pakovani BCD podaci

Binarno kodirane decimalne cifre

DEC	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

2.5. Brojevi u fiksnom zarezu

NEPAKOVANI BCD ?

Nepakovani BCD podaci

Jedna BCD cifra smješta se u jedan bajt

	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	1	0
3	0	0	0	0	0	0	1	1
4	0	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0	1
6	0	0	0	0	0	1	1	0
7	0	0	0	0	0	1	1	1
8	0	0	0	0	1	0	0	0
9	0	0	0	0	1	0	0	1

0	0
0	1
0	2
0	3
0	4
0	5
0	6
0	7
0	8
0	9

7 6 5 4 3 2 1 0

ZONA	BCD
------	-----

3	0
3	1
3	2
3	3
3	4
3	5
3	6
3	7
3	8
3	9

➔
Zonsko
proširenje da bi
se dobile ASCII
cifre

2.5. Brojevi u fiksnom zarezu

NEPAKOVANI BCD ?

Primjer:

Prikazati broj 2345 kao nepakovani BCD podatak, pa ga smjestiti u memoriju počevši od lokacije 1000.

2 3 4 5

0	2	0	3	0	0	0	0	0	0	1	0	5
				0	0	0	0	0	0	1	1	
				0	0	0	0	0	1	0	0	
				0	0	0	0	0	1	0	1	

2

3

4

5

Memorija
(binarni sadržaj)

7	6	5	4	3	2	1	0	
								1006
								1005
								1004
0	0	0	0	0	0	1	0	1003
0	0	0	0	0	0	1	1	1002
0	0	0	0	0	1	0	0	1001
0	0	0	0	0	1	0	1	1000
								0999

2.5. Brojevi u fiksnom zarezu

PAKOVANI BCD ?

Pakovani BCD podaci

Dvije BCD cifre smještaju se (pakuju) u jedan bajt

7	6	5	4	3	2	1	0
BCD				BCD			
"teža" cifra				"lakša" cifra			

Primjer:

Prikazati broj 2345 kao pakovani BCD podatak, pa ga smjestiti u memoriju počevši od lokacije 1000.

2 3 4 5

Memorija
(hex)

2	3	0	0	1	0	0	0	1	2	3
		0	1	0	0	0	1	0	4	5

Memorija
(binarni sadržaj)

7	6	5	4	3	2	1	0	
								1002
0	0	1	0	0	0	1	1	1001
0	1	0	0	0	1	0	1	1000
								0999

2.5. Brojevi u fiksnom zarezu

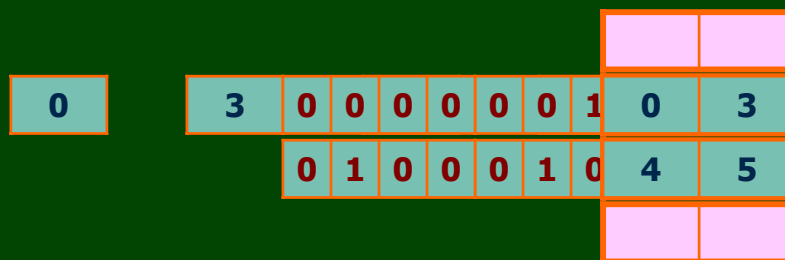
PAKOVANI BCD ?

Primjer:

Prikazati broj 345 kao pakovani BCD podatak, pa ga smjestiti u memoriju počevši od lokacije 1000.

0 3 4 5

Memorija
(hex)




Memorija
(binarni sadržaj)

7	6	5	4	3	2	1	0	
								1002
0	0	0	0	0	0	1	1	1001
0	1	0	0	0	1	0	1	1000
								0999

2.5. Brojevi u fiksnom zarezu

ZADACI ZA VJEŽBU...

- 
1. Broj 457 prikazati kao pakovani BCD, odnosno kao nepakovani BCD podatak, koji se u memoriji nalazi na adresi 0400h. Prikazati sadržaj zauzetih lokacija i u binarnom i u heksadecimalnom obliku.
 2. U memoriju, počev od lokacije 0400h, treba smjestiti sljedeći niz: 1, 12, 211. Prikazati sadržaj zauzetih lokacija, ako se podaci upisuju kao:
 - a) pakovani BCD,
 - b) nepakovani BCD.Lokacije prikazati u binarnom, odnosno u heksadecimalnom obliku.

2.6. Brojevi u pokretnom zarezu

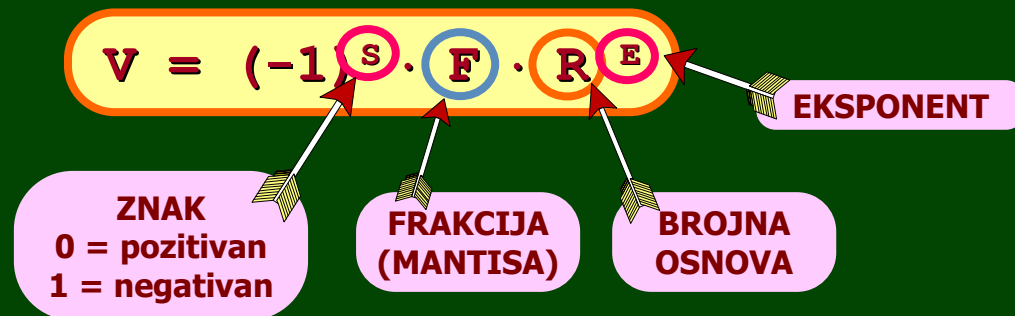
POKRETNİ ZAREZ ?

Predstavljanje brojeva u pokretnom zarezu

Brojevi u pokretnom zarezu služe za predstavljanje realnih brojeva

Često se koriste sinonimi pokretni (plivajući) zarez (tačka)

Opšti oblik broja u pokretnom zarezu (floating point - FP)



Primjer:

$$\begin{aligned} -125.34 &= -12.534 \cdot 10^1 = -1.2534 \cdot 10^2 = -0.12534 \cdot 10^3 \\ -125.34 &= (-1)^1 \cdot 0.12534 \cdot 10^3 \end{aligned}$$

Različiti proizvođači – različiti formati !!!

2.6. Brojevi u pokretnom zarezu

IEEE 754 ?

IEEE 754 FP standard

Najpoznatiji standard (1985)

Najšire primjenjivan u praksi (Intel, Motorola, ...)

Postoje tri formata FP podataka:

obična preciznost (single precision) – 32 bita



dvostruka preciznost (double precision) – 64 bita



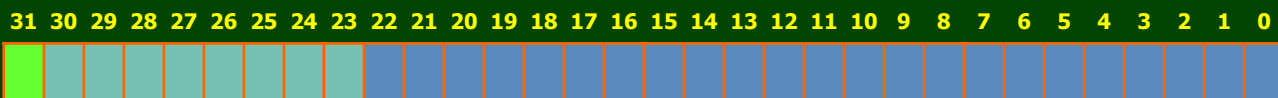
proširena preciznost (extended precision) – 80 bita



2.6. Brojevi u pokretnom zarezu

OBIČNA
PRECIZNOST?

FP podaci u običnoj preciznosti



ZNAK

0 = +

1 = -

MODIFIKOVANI
EKSPONENT
(8 bita)

MANTISA ili FRAKCIJA (23 bita)
podrazumijeva se normalizovana mantisa
1. *frakcija*

0 0 0 0 0 0 0 0

0

za predstavljanje nule i malih vrijednosti

0 0 0 0 0 0 0 1

1

modifikovani eksponent (ME): 1 .. 254

...

...

stvarni eksponent (SE): $SE = ME - 127$

1 1 1 1 1 1 1 0

254

-126 .. +127

1 1 1 1 1 1 1 1

255

za predstavljanje beskonačnosti

2.6. Brojevi u pokretnom zarezu

OBIČNA
PRECIZNOST?

Primjer: Prikazati broj 19.25 kao FP podatak u običnoj preciznosti.

$$19.25_{10} = ?_2$$

$$19 : 2$$

9	1
4	1
2	0
1	0
0	1

$$0.25 * 2$$

0.5	0
1.0	1
0.0	

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

0 1 0 0 0 0 0 1 1 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

ZNAK
0 = +

MODIFIKOVANI
EKSPONENT
 $4 + 127 = 131$
 $131_{10} = 1000011_2$

FRAKCIJA
001101

$$19.25_{10} = 10011.01_2 = +1 \ 001101 \ 2^4$$

$$19_{10} = 10011_2$$

$$0.25_{10} = 0.01_2$$

2.6. Brojevi u pokretnom zarezu

OBIČNA
PRECIZNOST?

Primjer: Prikazati broj -0.3 kao FP podatak u običnoj preciznosti.

$$0.3_{10} = ?_2$$

$$0.3 * 2$$

0.6

0

1.2

1

0.2

0.4

0

0.8

0

1.6

1

0.6

$$-0.3_{10} = -0.010011001..._2 = -1.0011001..._2 \cdot 2^{-2}$$

$$\ominus 1 \text{ } 00110011001100110011001 \cdot 2^{-2}$$

FRAKCIJA

00110011001100110011001

ZNAK

1 = -

MODIFIKOVANI
EKSPONENT

$$-2 + 127 = 125$$

$$125_{10} = 01111101_2$$

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

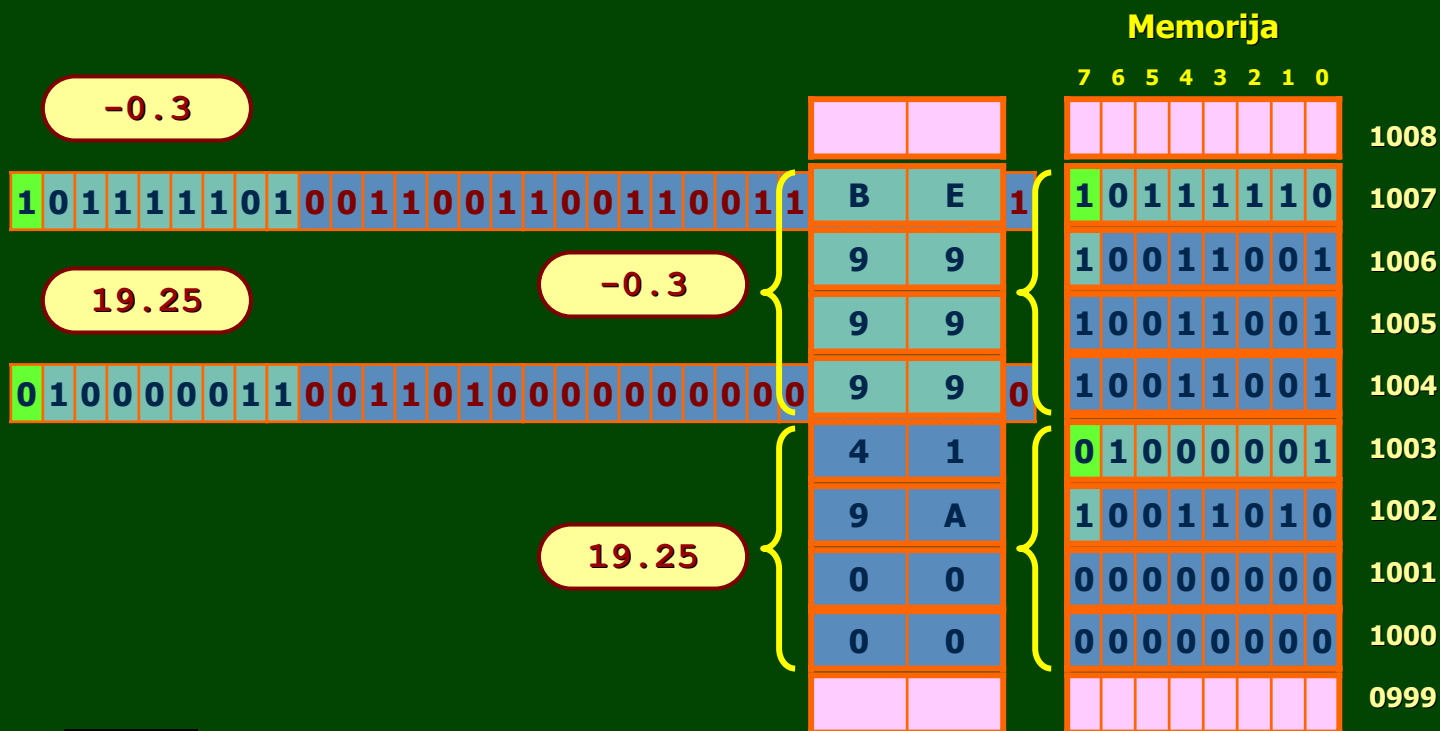
1 0 1 1 1 1 1 0 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1

2.6. Brojevi u pokretnom zarezu

OBIČNA PRECIZNOST?

Primjer:

Sljedeći niz realnih brojeva 19.25, -0.3 smjestiti u memoriju kao podatke u pokretnom zarezu u običnoj preciznosti počevši od lokacije 1000.



2.6. Brojevi u pokretnom zarezu

OBIČNA
PRECIZNOST?

Najmanja vrijednost normalizovanog FP podatka



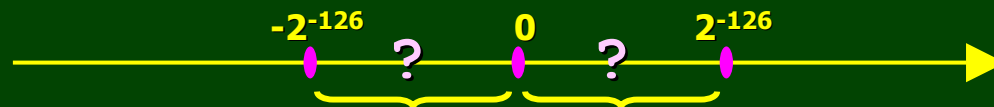
MODIFIKOVANI EKSPONENT
ME = 1

STVARNI EKSPONENT
SE = ME - 127 = -126

NORMALIZOVANA MANTISA
F = 0

$$|V_{\min}| = 1.0 \cdot 2^{-126} = 2^{-126}$$

2^{-126} i -2^{-126} su nuli najbliže vrijednosti koje mogu da se prikažu



PODBAČAJ

(koristi se DENORMALIZOVANA MANTISA)

2.6. Brojevi u pokretnom zarezu

DVOSTRUKA
PRECIZNOST?

FP podaci u dvostrukoj preciznosti



ZNAK

0 = +

1 = -

MODIFIKOVANI
EKSPONENT
(11 bita)

MANTISA ili FRAKCIJA (52 bita)
podrazumijeva se normalizovana mantisa
1. **frakcija**

0 0 0 0 0 0 0 0 0 0 0 0

0

za predstavljanje nule i malih vrijednosti

0 0 0 0 0 0 0 0 0 0 0 1

1

modifikovani eksponent (ME): 1 .. 2046

• • •

• • •

stvarni eksponent (SE): $SE = ME - 1023$
-1022 .. +1023

1 1 1 1 1 1 1 1 1 1 1 0

2046

1 1 1 1 1 1 1 1 1 1 1 1

2047

za predstavljanje beskonačnosti

2.6. Brojevi u pokretnom zarezu

DVOSTRUKA PRECIZNOST?

Primjer: Prikazati broj 19.25 kao FP podatak u dvostrukoj preciznosti.

$$19.25_{10} = 10011.01_2 = +1 \ 001101_2 \cdot 2^4$$

ZNAK

$$0 = +$$

FRAKCIJA

001101000.....0

MODIFIKOVANI EKSPONENT

$$4 + 1023 = 1027$$

$$1027_{10} = 10000000011_2$$

[illegible]

2.6. Brojevi u pokretnom zarezu

DVOSTRUKA PRECIZNOST?

Najmanja vrijednost normalizovanog FP podatka



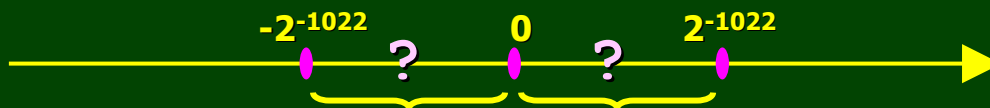
MODIFIKOVANI EKSPONENT

STVARNI EKSPONENT
SE = ME-1023 = -1022

NORMALIZOVANA MANTISA
F = 0

$$|V_{\min}| = 1.0 \cdot 2^{-1022} = 2^{-1022}$$

2^{-1022} i -2^{-1022} su nuli najbliže vrijednosti koje mogu da se prikažu



PODBAČAJ

(koristi se DENORMALIZOVANA MANTISA)

2.6. Brojevi u pokretnom zarezu

IEEE 754 ?

Ostale FP vrijednosti

nula



beskonačnost



nije broj (NaN)



2.6. Brojevi u pokretnom zarezu

ZADACI ZA VJEŽBU...

1. Brojeve 123, -123, 1, 2^{-10} , 2^{-100} , -2^{-1000} predstaviti kao FP podatke:

a) u običnoj preciznosti, b) u dvostrukoj preciznosti.

Date brojeve treba smjestiti u memoriju počevši od lokacije 1000h. Prikazati sadržaj memorije binarno, odnosno heksadecimalno.

2. Na slici je prikazan sadržaj jednog segmenta memorije. Odrediti realan broj upisan kao FP podatak obične precitnosti na lokaciji 0F000h.

0	1	0	0	0	0	0	0	0F003h
1	1	0	0	0	0	0	0	0F002h
0	0	0	0	0	0	0	0	0F001h
0	0	0	0	0	0	0	0	0F000h
1	0	0	0	0	0	0	0	0EFFFh