

# Vježbe IX

## Kontrola procesorske jedinice

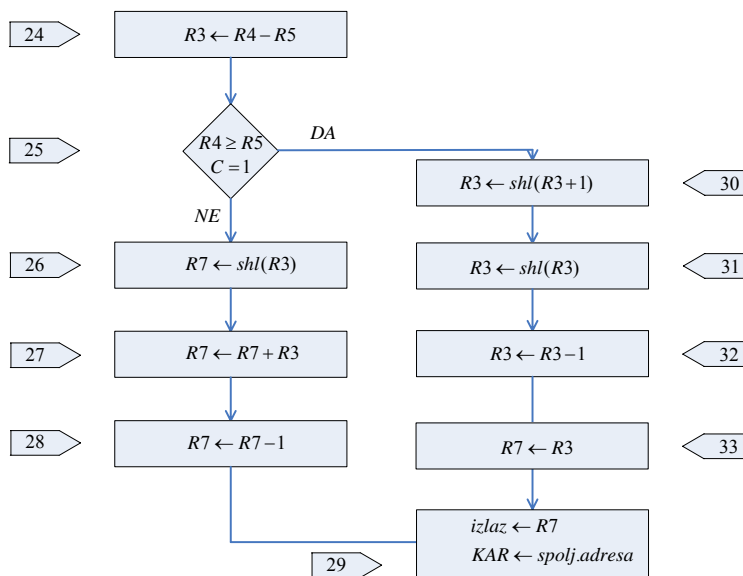
1. Napisati mikroprogram u simboličkom i mnemoničkom obliku kojim se realizuje sljedeća funkcija  $R7 \leftarrow f(R4 - R5)$ , pri čemu je:

$$f(x) = \begin{cases} 4x + 3, & x \geq 0 \\ 3x - 1, & x < 0 \end{cases}$$

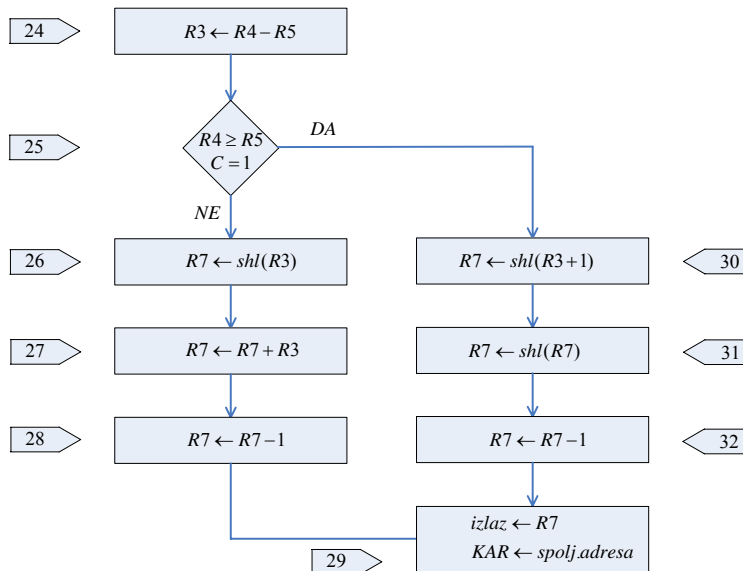
Početna mikroinstrukcija se nalazi na lokaciji 24.

Imajući u vidu da nam je za ovaj slučaj  $x = R4 - R5$ , funkciju možemo zapisati:

$$R7 \leftarrow \begin{cases} 4(R4 - R5) + 3, & R4 - R5 \geq 0, R4 \geq R5 \\ 3(R4 - R5) - 1, & R4 - R5 < 0, R4 < R5 \end{cases}$$



Slika 1. Prvi način rješavanja zadatka



Slika 2. Drugi način rješavanja zadatka

Zadatak je moguće riješiti na više načina, zavisno od odabranog načina realizacije funkcije. Na slikama 1 i 2 su dati algoritmi dva moguća načina. Odabraćemo rješavanje sa slike 2 i za njega je dato objašnjenje u nastavku.

Funkcija zavisi od promjenljive  $x$ , pri čemu je zadatkom dato da je  $x = R4 - R5$ . Smjestimo promjenljivu u pomoćni registar  $R3$ ,  $R3 \leftarrow R4 - R5$ . Da bi crtali algoritam, neophodno je da ispitamo vrijednost promjenljive. Ukoliko je promjenljiva veća ili jednaka nuli ( $\geq$ ), funkcija uzima vrijednost  $4(R4 - R5) + 3$ ; ako je promjenljiva manja od nule ( $<$ ) funkcija je jednaka  $3(R4 - R5) - 1$ . Ispitivanje vrijednosti promjenljive se vrši korišćenjem bita  $C$ . Ukoliko je  $C=1$ , tada je  $R4 - R5 \geq 0$ , odnosno  $R4 \geq R5$ . Ako je, pak,  $C=0$  važi  $R4 - R5 < 0$ , odnosno  $R4 < R5$ .

Posmatrajmo „DA“ granu ( $C=1$ ). Funkcija je, u tom slučaju,  $4(R4 - R5) + 3$ . Imajući u vidu da je množenje sa 2 operacija *shift left* (*shl*), i da možemo istovremeno sabirati samo po jednu jedinicu (inkrementiranje), željenu vrijednost dobijamo tako što:

1. U izlazni registar  $R7$  smjestimo  $shl(R3+1)$ , što predstavlja  $2((R4 - R5) + 1)$ .
2. U izlazni registar  $R7$  smjestimo  $shl(R7)$ , što predstavlja  $2R7 = 2 * 2((R4 - R5) + 1) = 4(R4 - R5) + 4$ .
3. Kako nam je potrebna vrijednost  $4(R4 - R5) + 3$ , a u registru  $R7$  imamo  $4(R4 - R5) + 4$ , to je potrebno izvršiti oduzimanje jedinice, odnosno  $R7 \leftarrow R7 - 1$ , što predstavlja  $4(R4 - R5) + 4 - 1 = 4(R4 - R5) + 3$ .

Kad smo u registru  $R7$  dobili željenu vrijednost funkcije, prosljedimo je na izlaz,  $izlaz \leftarrow R7$ , a za  $KAR$  uzmemo spoljašnju adresu.

Posmatrajmo „NE“ granu (nije ispunjen uslov  $R4 - R5 \geq 0$ , što automatski znači da je  $R4 - R5 < 0$ ). Funkcija je u tom slučaju  $3(R4 - R5) - 1$ , i dobijamo je tako što:

1. U izlazni registar  $R7$  smjestimo  $shl(R3)$ , što predstavlja  $2(R4 - R5)$ .
2. U izlazni registar  $R7$  smjestimo  $R7 + R3$ , što je  $2(R4 - R5) + (R4 - R5) = 3(R4 - R5)$ .
3. Kako nam je potrebna vrijednost  $3(R4 - R5) - 1$ , a u registru  $R7$  imamo  $3(R4 - R5)$ , to je potrebno izvršiti oduzimanje jedinice, odnosno  $R7 \leftarrow R7 - 1$ .

U registru  $R7$  smo, opet, dobili željenu vrijednost funkcije, koju prosljeđujemo na izlaz,  $izlaz \leftarrow R7$ , a za  $KAR$  uzmemo spoljašnju adresu.

Numerisanje lokacija se vrši redom, pri čemu se početna adresa (25 u ovom slučaju) dodjeljuje prvoj operaciji. Kad se naiđe na grananje u algoritmu, prvo se vrši numeracija „NE“ grane do kraja, nakon čega se vraćamo na „DA“ granu.

Mnemonički oblik programa se dobija prepisivanjem operacija iz algoritma. Pri tome se vodi računa o lokacijama, odnosno o tome da li se operacije izvršavaju uzastopno ( $KAR \leftarrow KAR + 1$ ) ili se prelazi na neku, tačno specificiranu lokaciju (npr.  $KAR \leftarrow 31$ ). Potrebno je voditi računa o operacijama koje uključuju ispitivanje vrijednosti određenih bita (bita  $C$  u ovom slučaju). Tada se odgovarajuća mikroinstrukcija piše u obliku **if** petlje, na slijedeći način:

**if** (uslov) **then** (adresa na koju se prelazi ako je uslov zadovoljen) **else** (adresa na koju se prelazi ako uslov nije zadovoljen)

### Mnemonički oblik:

ADS	Mikroinstrukcije
24	$R3 \leftarrow R4 - R5$ ; $KAR \leftarrow KAR + 1$
25	if ( $C=1$ ) then ( $KAR \leftarrow 30$ ) else ( $KAR \leftarrow KAR + 1$ )
26	$R7 \leftarrow shl(R3)$ ; $KAR \leftarrow KAR + 1$
27	$R7 \leftarrow R7 + R3$ ; $KAR \leftarrow KAR + 1$
28	$R7 \leftarrow R7 - 1$ ; $KAR \leftarrow KAR + 1$
29	$izlaz \leftarrow R7$ ; $KAR \leftarrow spolj.adresa$
30	$R7 \leftarrow shl(R3+1)$ ; $KAR \leftarrow KAR + 1$
31	$R7 \leftarrow shl(R7)$ ; $KAR \leftarrow KAR + 1$
32	$R7 \leftarrow R7 - 1$ ; $KAR \leftarrow 29$

Simbolički oblik mikroprograma podrazumijeva definisanje 8 polja:

**A i B:** polja prvog i drugog operanda (registri nad čijim sadržajem se vrše operacije; npr.: R3, R7... ili – ako nema operanda)

**D:** polje rezultata (registar u kojem se bilježi rezultat; npr. R4, R6... ili NONE ako nema rezultata)

**F:** polje aritmetičke funkcije; najčešće aritmetičke funkcije:

Sabiranje (sadržaja dva registra) ADD                      Oduzimanje broja 1 (dekrementiranje) DEC

Oduzimanje (sadržaja dva registra) SUB                      Komplementiranje COM

Sabiranje sa 1 (inkrementiranje) INC                      Transfer (prepisivanje vrijednosti) TSF

**H:** polje pomjeračke funkcije; najčešće pomjeračke funkcije:

Množenje sa 2 (shift left) SHL                      Nema pomjeračke funkcije (no shift) NSH

Dijeljenje sa 2 (shift right) SHR                      Upisivanje nule u registar ZERO

**MUX1:** Interna adresa (poznata tačna lokacija) INT, Spoljašnja adresa EXT, u ostalim slučajevima –

**MUX2:** Prelazak na susjednu lokaciju NEXT, Prelazak pod uslovom da je C=1 LC, Bezuslovni prelazak LAD

**Adresa:** Brojčana vrijednost ako je poznata, u suprotnom –

**Napomena:** Navedene su vrijednosti koje se najčešće srijeću; kompletne tabele za svako od polja se nalaze u knjizi!

### Simbolički oblik:

<u>KAR</u>	<u>A</u>	<u>B</u>	<u>D</u>	<u>F</u>	<u>H</u>	<u>MUX1</u>	<u>MUX2</u>	<u>Adresa</u>
24	R4	R5	R3	SUB	NSH	-	NEXT	-
25	-	-	NONE	TSF	NSH	INT	LC	30
26	R3	-	R7	TSF	SHL	-	NEXT	-
27	R7	R3	R7	ADD	NSH	-	NEXT	-
28	R7	-	R7	DEC	NSH	-	NEXT	-
29	R7	-	NONE	TSF	NSH	EXT	LAD	-
30	R3	-	R7	INC	SHL	-	NEXT	-
31	R7	-	R7	TSF	SHL	-	NEXT	-
32	R7	-	R7	DEC	NSH	INT	LAD	29

**Napomena:** Kada se definiše izlaz iz algoritma i vrijednost KAR-a (lokacija 29), polje D se ravnopravno može popunjavati sa NONE ili izlaz, a Adresa sa "-" ili spoljašnja adresa.

2. Napisati mikroprogram u simboličkom i mnemoničkom obliku, kojim se realizuje sljedeća funkcija  $R1 \leftarrow f(R3 - R5)$ , pri čemu je  $f(x) = |x|$ . Početna mikroinstrukcija je na lokaciji 45.

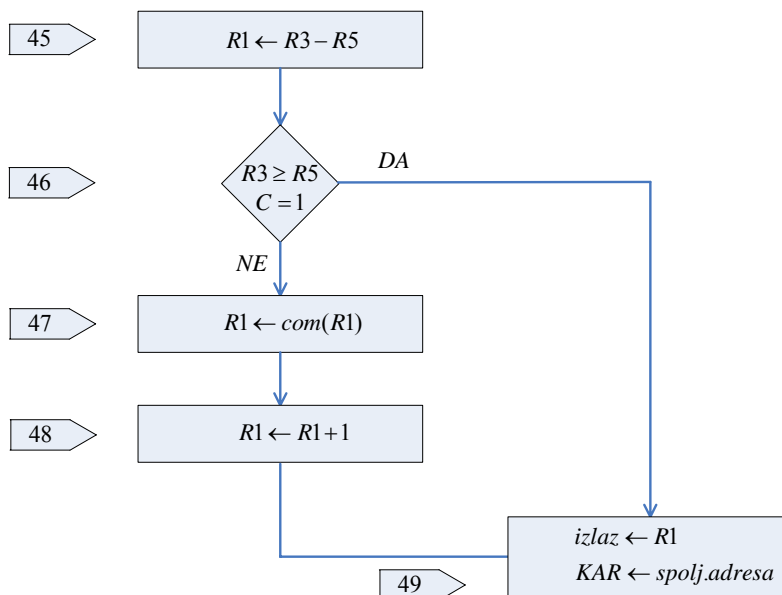
Znajući da je  $f(x) = |x| = \begin{cases} x, & x \geq 0 \\ -x, & x < 0 \end{cases}$ , zaključujemo:  $R1 \leftarrow \begin{cases} (R3 - R5), & R3 - R5 \geq 0, R3 \geq R5 \\ -(R3 - R5), & R3 - R5 < 0, R3 < R5 \end{cases}$

Funkcija zavisi od promjenljive  $x$ , koja je u ovom slučaju  $x = R3 - R5$ . Smjestimo promjenljivu u pomoćni registar  $R1$ ,  $R1 \leftarrow R3 - R5$ .

Ukoliko je promjenljiva veća ili jednaka nuli ( $\geq$ ) funkcija uzima vrijednost  $(R3 - R5)$ ; ako je promjenljiva manja od nule ( $<$ ) funkcija je jednaka  $-(R3 - R5)$ . Ispitivanje vrijednosti promjenljive se vrši korišćenjem bita C. Ukoliko je C=1, tada je  $R3 - R5 \geq 0$ , odnosno  $R3 \geq R5$ . Ako je, pak, C=0 važi  $R3 - R5 < 0$ , odnosno  $R3 < R5$ .

Posmatrajmo „DA“ granu (C=1). Funkcija je u tom slučaju jednaka promjenljivoj, te je potrebno samo prosljediti promjenljivu na izlaz.

Posmatrajmo „NE“ granu (nije ispunjen uslov  $R3 - R5 \geq 0$ , što automatski znači da je  $R3 - R5 < 0$ ). Tada je na izlaz potrebno prosljediti negativnu vrijednost promjenljive. Negativan broj predstavlja zapravo dvojni komplement originalne vrijednosti (jedinični komplement plus 1).



#### Mnemonički oblik:

ADS	Mikroinstrukcije
45	$R1 \leftarrow R3 - R5$ ; $KAR \leftarrow KAR + 1$
46	if (C=1) then ( $KAR \leftarrow 49$ ) else ( $KAR \leftarrow KAR + 1$ )
47	$R1 \leftarrow com(R1)$ ; $KAR \leftarrow KAR + 1$
48	$R1 \leftarrow R1 + 1$ ; $KAR \leftarrow KAR + 1$
49	$izlaz \leftarrow R1$ ; $KAR \leftarrow spolj.adresa$

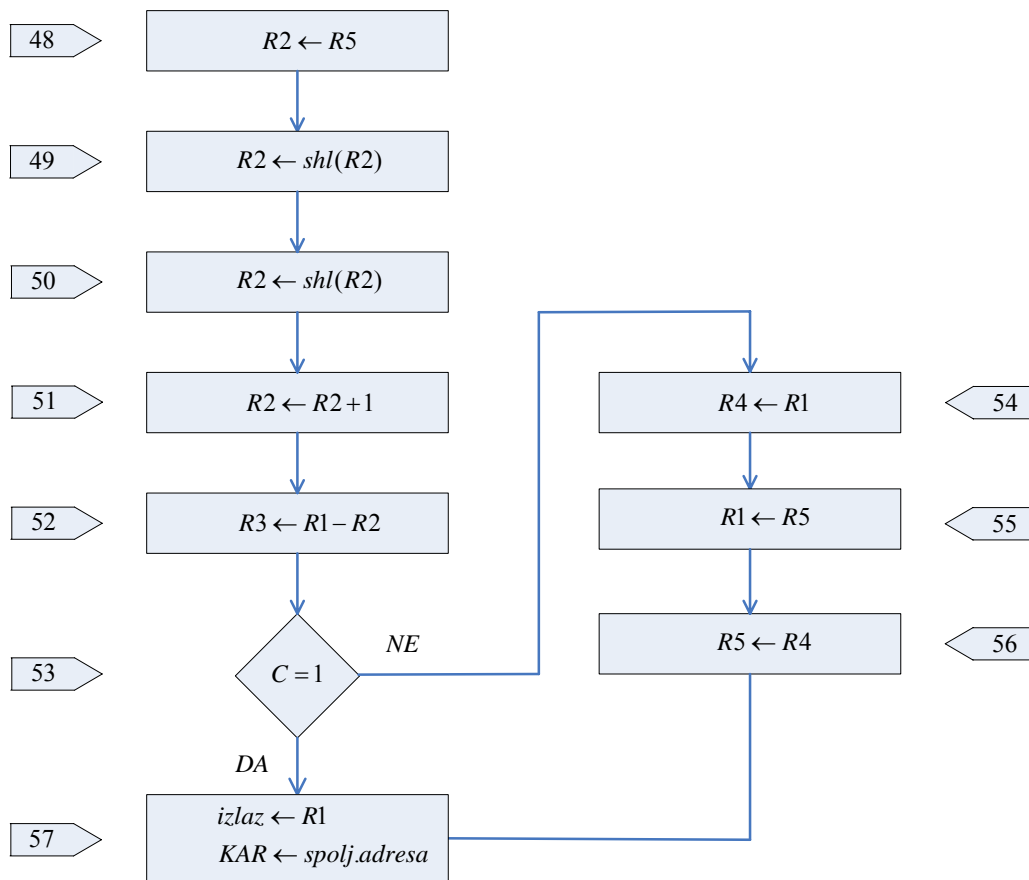
#### Simbolički oblik:

<u>KAR</u>	<u>A</u>	<u>B</u>	<u>D</u>	<u>F</u>	<u>H</u>	<u>MUX1</u>	<u>MUX2</u>	<u>Adresa</u>
45	R3	R5	R1	SUB	NSH	-	NEXT	-
46	-	-	NONE	TSF	NSH	INT	LC	49
47	R1	-	R1	COM	NSH	-	NEXT	-
48	R1	-	R1	INC	NSH	-	NEXT	-
49	R1	-	NONE	TSF	NSH	EXT	LAD	-

3. Napisati program u simboličkom i mnemoničkom obliku kojim se realizuje sljedeća operacija: u registrima R1 i R5 su smješteni cijeli brojevi; ukoliko je ispunjen uslov da je  $R1 \geq 4R5 + 1$ , brojevi ostaju na prvobitnim pozicijama, u suprotnom mijenjaju mjesta. Početna mikroinstrukcija je na lokaciji 48.

Da bismo mogli utvrditi da li je zadovoljena nejednakost  $R1 \geq 4R5 + 1$ , potrebno je najprije vrijednost  $4R5 + 1$  smjestiti u pomoćni registar. To, očigledno, ne možemo obaviti u jednom koraku, već:

1. U pomoćni (proizvoljni) registar R2 smještamo vrijednost R5.
2. U R2 smještamo  $shl(R2)$  što predstavlja  $2R2 = 2R5$ .
3. U R2 smještamo  $shl(R2)$  što predstavlja  $2R2 = 2(2R5) = 4R5$ .
4. U R2 smještamo  $R2 + 1$ , što predstavlja  $R2 + 1 = 4R5 + 1$ .

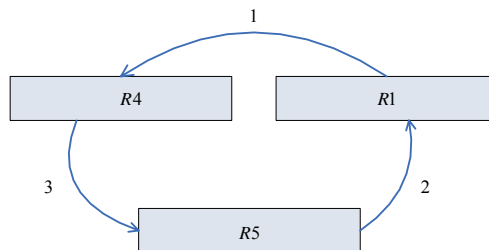


Na taj način smo u registar  $R2$  smjestili vrijednost  $4R5 + 1$ , koju dalje poredimo sa vrijednošću registra  $R1$ .

Ukoliko je zadovoljena nejednakost  $R1 \geq 4R5 + 1$ , odnosno  $C=1$ , vrijednosti registara se ne mijenjaju, te jednostavno prosljeđujemo vrijednost registra  $R1$  (ili  $R5$ , pošto nije precizno navedeno u zadatku) na izlaz.

Ukoliko, pak, nejednakost nije zadovoljena, potrebno je da vrijednosti registara  $R1$  i  $R5$  zamjene mjesta. Ako bismo direktno uputili  $R1 \leftarrow R5$  postigli bismo da se vrijednost registra  $R5$  nađe u registru  $R1$ , ali bismo izgubili informaciju o stvarnoj vrijednosti registra  $R1$  koju treba da smjestimo u registar  $R5$ . Slično bi se dogodilo ukoliko bismo uradili sljedeće:  $R5 \leftarrow R1$ . Zaključujemo da na ovaj način ne možemo izvršiti zamjenu vrijednosti koje se čuvaju u registrima.

Zamjenu ćemo postići upotrebom proizvoljnog pomoćnog registra, na sljedeći način:



1. U pomoćni registar smjestimo vrijednost registra  $R1$ .
2. U registar  $R1$  smjestimo vrijednost registra  $R5$ .
3. U registar  $R5$  smjestimo sadržaj pomoćnog registra koji čuva vrijednost registra  $R1$ .

Na taj način postizemo da se u registru  $R1$  nalazi vrijednost registra  $R5$  i obrnuto.

Sada možemo pisati mnemonički, odnosno simbolički oblik odgovarajućeg mikroprograma:

**Mnemonički oblik:**

<b>ADS</b>	<b>Mikroinstrukcije</b>
48	$R2 \leftarrow R5; KAR \leftarrow KAR + 1$
49	$R2 \leftarrow shl(R2); KAR \leftarrow KAR + 1$
50	$R2 \leftarrow shl(R2); KAR \leftarrow KAR + 1$
51	$R2 \leftarrow R2 + 1; KAR \leftarrow KAR + 1$
52	$R3 \leftarrow R1 - R2; KAR \leftarrow KAR + 1$
53	if (C=1) then ( $KAR \leftarrow 57$ ) else ( $KAR \leftarrow KAR + 1$ )
54	$R4 \leftarrow R1; KAR \leftarrow KAR + 1$
55	$R1 \leftarrow R5; KAR \leftarrow KAR + 1$
56	$R5 \leftarrow R4; KAR \leftarrow KAR + 1$
57	$izlaz \leftarrow R1; KAR \leftarrow spolj.adresa$

**Simbolički oblik:**

<b><u>KAR</u></b>	<b><u>A</u></b>	<b><u>B</u></b>	<b><u>D</u></b>	<b><u>F</u></b>	<b><u>H</u></b>	<b><u>MUX1</u></b>	<b><u>MUX2</u></b>	<b><u>Adresa</u></b>
48	R5	-	R2	TSF	NSH	-	NEXT	-
49	R2	-	R2	TSF	SHL	-	NEXT	-
50	R2	-	R2	TSF	SHL	-	NEXT	-
51	R2	-	R2	INC	NSH	-	NEXT	-
52	R1	R2	R3	SUB	NSH	-	NEXT	-
53	-	-	NONE	TSF	NSH	INT	LC	57
54	R1	-	R4	TSF	NSH	-	NEXT	-
55	R5	-	R1	TSF	NSH	-	NEXT	-
56	R4	-	R5	TSF	NSH	-	NEXT	-
57	R1	-	NONE	TSF	NSH	EXT	LAD	-