

3¹⁰ Асемблер који је произвео наредна два предметна програма ради тако што секције ређа једну за другом почев од нуле, код РС релативног адресирања одмах уграђује део који се односи на изворишну адресу, а разрешавање на местима коришћења глобалних симбола оставља за што каснији тренутак, осим уколико садржај локације може у потпуности да разреши (без потребе за каснијом релокацијом). Приказана два фајла се доводе на улаз линкера.

#tabela simbola						#tabela simbola					
#ime	sek	vr	vel	vid	atr	#ime	sek	vr	vel	vid	atr
1 text	1	0	C	1	RXP	1 text	1	0	A	1	RXP
2 bss	2	C	12	1	RW	2 data	2	A	8	1	RWP
3 c	UND	0		g		3 bss	3	12	16	1	RW
4 a	1	6		g		4 a	UND	0		g	
5 d	2	10		g		5 b	2	E		g	
#rel.text						#rel.text					
#adr	tip				simbol	#adr	tip				simbol
2	R_386_32			4	a	4	R_386_PC32			6	d
7	R_386_PC32			3	c	#rel.data					
#sadrzaji prisutnih sekcija						#adr	tip				simbol
87 10	00 00 00 00	FC F5	FF FF FF FF			C	R_386_32			3	
						#sadrzaji prisutnih sekcija					
						5A FF FF FF	FA FF FF FF				
						11 12 13	14 15 16 17				

Излаз линкера треба да буде предметни програм намењен за даље повезивање (формиран по истим правилима по којима асемблер генерише излаз). Програми се обрађују слева надесно (на основу тога треба да буду поређане секције, док се симболи после секција испisuју по алфавитном поретку).

- 2) Приказати табелу симбола излазног предметног програма.
4) Приказати табеле релокација излазног предметног програма.
4) Ако би се излазни предметни програм учитао у меморију тако да се секције сместе једна за другом по редоследу из табеле симбола, тако да прва секција буде на адреси 0x400, приказати како би изгледао садржај меморије.

ЕЛФ табела симбола

Индекс	Вредност	Лок/Глоб	Секција	Назив
0	0	L	UND	
1	0, 16	L	.text	.text
2	16, 8	L	.data	.data
3	16, 28	L	.bss	.bss
4	0	2	.text	a
5	10	2	.data	c
6	38	2	.bss	c
7	22	2	.bss	d

ЕЛФ релокациони записи .text секције

Офсет	Тип	Симбол
2	ABS 32	a
7	PC 32	c
10	PC 32	d

ЕЛФ релокациони записи .data секције

Офсет	Тип	Симбол
18	ABS 32	.bss

$$56$$

$$3 \cdot 16 = 48$$

$$8$$

$$38$$

$$10$$

$$16$$

$$A$$

$$10$$

$$8$$

$$34$$

$$22$$

$$430$$

$$C$$

$$12$$

$$C$$

$$1$$

$$12$$

$$18$$

$$12$$

$$42$$

$$2 \cdot 16 + 10$$

$$20$$

$$0101$$

$$1010$$

$$ff \ ff \ ff \ ff$$

$$00 \ 00 \ 00 \ 00 + 1$$

$$-6$$

$$6$$

$$438$$

$$-406$$

$$2d$$

$$422$$

$$-406$$

$$1c$$

$$-6$$

$$00 \ 00 \ 00 \ 05 + 1$$

$$1010$$

$$101$$

$$ff \ ff \ ff \ ff$$

$$00 \ 00 \ 00 \ 05 + 1$$

102

фору $\delta = |T|$

06 04 00 00 2 00 00 00

400:

87 10 [00 00 00 00] FC [F5 FF FF FF] FF

1st .text

40c:

5A FF FF FF [FA FF FF FF] FF [?]

2nd .text

43 18 15 16

41b:

11 12 [13 14 15 16] 17 [?]

2nd .data

.text .bss

41e:

[00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00] [00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00] ~ 2nd .bss

16 15 14 13

4 30

16 15 18 43