

Princeton University
COS 217: Introduction to Programming Systems
ELF Relocation Information

Assembly Language Program:

```
!-----  
.section ".data"  
  
pcGreeting:  
    .asciz "Hello\n"  
  
!-----  
  
.section ".text"  
  
    .global main  
main:  
    save  %sp, -96, %sp  
    sethi %hi(pcGreeting), %o0  
    or    %o0, %lo(pcGreeting), %o0  
    call  printf  
    nop  
    mov   0, %i0  
    ret  
    restore
```

Symbol Table:

Label	Section	Offset	Local/Global	Sequence #
pcGreeting	data	0	local	0
main	text	0	global	1
printf	?	?	global	2

Data Section:

Offset	Contents	Explanation
0	01001000	'H'
1	01100101	'e'
2	01101100	'l'
3	01101100	'l'
4	01101111	'o'
5	00001010	'\n'
6	00000000	'\0'

Text Section:

Offset	Contents	Explanation
0-3	10 01110 111100 01110 1 111110100000	save %sp, -96, %sp
4-7	00 01000 100 ??????????????????????????	sethi %hi(pcGreeting), %o0
8-11	10 01000 000010 01000 1 ??????????????????	or %o0, %lo(pcGreeting), %o0
12-15	01 ??????????????????????????????????????	call printf
16-19	00 00000 100 000000000000000000000000	nop
20-23	10 11000 000010 00000 1 0000000000000000	mov 0, %i0 Same as: or %g0, 0, %i0
24-27	10 00000 111000 11111 1 0000000001000	ret Same as: jmpl %i7+8, %g0
28-31	10 00000 111101 00000 0 00000000 00000	restore Same as: restore %g0, %g0, %g0

Relocation Entries for Data Section:

(None)

Relocation Entries for Text Section:

Offset	Relocation Type	Label	Sequence #	Addend
4	R_SPARC_HI22	0	0	0
8	R_SPARC_LO10	0	0	0
12	R_SPARC_WDISP30	2	0	0

(Relocation type, R_SPARC_WDISP22 is used to mark **branch** instructions for resolution)