

Princeton University

COS 217: Introduction to Programming Systems

Assembler Output for the "detecta" Program

Symbol Table

Label	Section	Byte Offset	Local/Global	Type	Label Sequence #
cMessage	rodata	0	local	?	0
main	text	0	global	function	1
skip	text	26	local	?	2
getchar	?	?	global	?	3
printf	?	?	global	?	4

Relocation Records

Section	Byte Offset	Relocation Type	Label Sequence #
text	4	R_386_PC32 (32 bit displacement)	3
text	14	R_386_32 (32 bit absolute)	0
text	19	R_386_PC32 (32 bit displacement)	4

Rodata Section

Byte Offset	Contents (hex)	Explanation
0	54	'T'
1	68	'h'
2	65	'e'
3	20	' '
4	63	'c'
5	68	'h'
6	61	'a'
7	72	'r'
8	20	' '
9	69	'i'
10	73	's'
11	20	' '
12	41	'A'
13	3E	'.'
14	0A	'\n'
15	00	'\0'

Text Section

Byte Offset	Contents (hex)	Explanation
0	55	pushl %ebp 01010101 This is a "pushl %ebp" instruction
1-2	89 E5	movl %esp, %ebp 10001001 11 100 101 This is a "movl" instruction whose source operand is a register The M field designates a register The source register is %esp The destination register is %ebp
3-7	E8????????	call getchar 11101000 ????????????????????????????????????? This is a "call" instruction with a 4 byte immediate operand This is the displacement to the instruction to be called.
8-10	83F841	cmpl \$'A', %eax 10000011 11 111 000 01000001 This is some "l" instruction that has a 1 byte immediate operand The M field designates a register This is a "cmp" instruction The destination register is %eax The immediate operand is 'A'
11-12	750D	jne skip 01110101 00001101 This is a jne instruction that has a 1 byte immediate operand The displacement from next inst. to destination instr. is 13 (26 - 13 = 13)
13-17	68????????	pushl \$cMessage 11101000 ????????????????????????????????????? This is a "pushl" instruction with a 4 byte immediate operand This is the data to be pushed
18-22	E8????????	call printf 11101000 ????????????????????????????????????? This is a "call" instruction with a 4 byte immediate operand This is the displacement to the instruction to be called.
23-25	83C404	addl \$4, %esp 10000011 11 000 100 00000100 This is some "l" instruction that has a 1 byte immediate operand The M field designates a register This is an "add" instruction The destination register is %esp The immediate operand is 4
26-30	B800000000	movl \$0, %eax 10111000 00000000000000000000000000000000 This is an instruction of the form "movl 4-byte-immediate, %eax" The immediate operand is 0
31-32	89EC	movl %ebp, %esp 10001001 11 101 100 This is a "movl" instruction whose source operand is a register The M field designates a register The source register is %ebp The destination register is %esp
33	5D	popl %ebp 01011101 This is a "popl %ebp" instruction
34	C3	ret 11000011 This is a "ret" instruction