20-1)

main.o:

6: $\underbrace{\text{e8}}_{\text{call opcode}}$ $\underbrace{\text{fc ff ff ff}}_{\to\ -4}$ call   7

relocation table entry : <7, R_386-PC32, printf>

```
foreach section s {                          (suppose .text is
  foreach relocation r {                       0x80483b4 )
    refptr = s+r.offset;                     ( 0x80483bb )
    if (r.type == R_386-PC32)
                  ~~ADDR(r.symbol)~~
     refaddr = s + r.offset
     *refptr = ADDR(r.symbol) + *refptr - refaddr
             = ADDR(printf) + (-4) - 0x80483bb
             = 0x80483c8    - 0x80483bf
             = 0xc8         - 0xbf
             = 200          - 191
             = 9   = 0x9
```

80483ba: $\underbrace{\text{e8}}_{\text{call}}$ $\underbrace{\text{09 00 00 00}}_{9}$

PC = 80483bf

1. push PC onto stack
2. PC ← PC + 0x9 = 80483bf + 9 = 80483c8 = print

int *start = &array[0]   int *middle = &array[8]

.data : 0 : 00 00 00 00   < 0, R_386_32, array >

```
    if (r.type == R-386-32)
        *refptr = ADDR(r.symbol) + *refptr
```

4 : 08 00 00 00   < 4, R-386-32, array >

20-2/ ELF Executable

○ ELF HEader

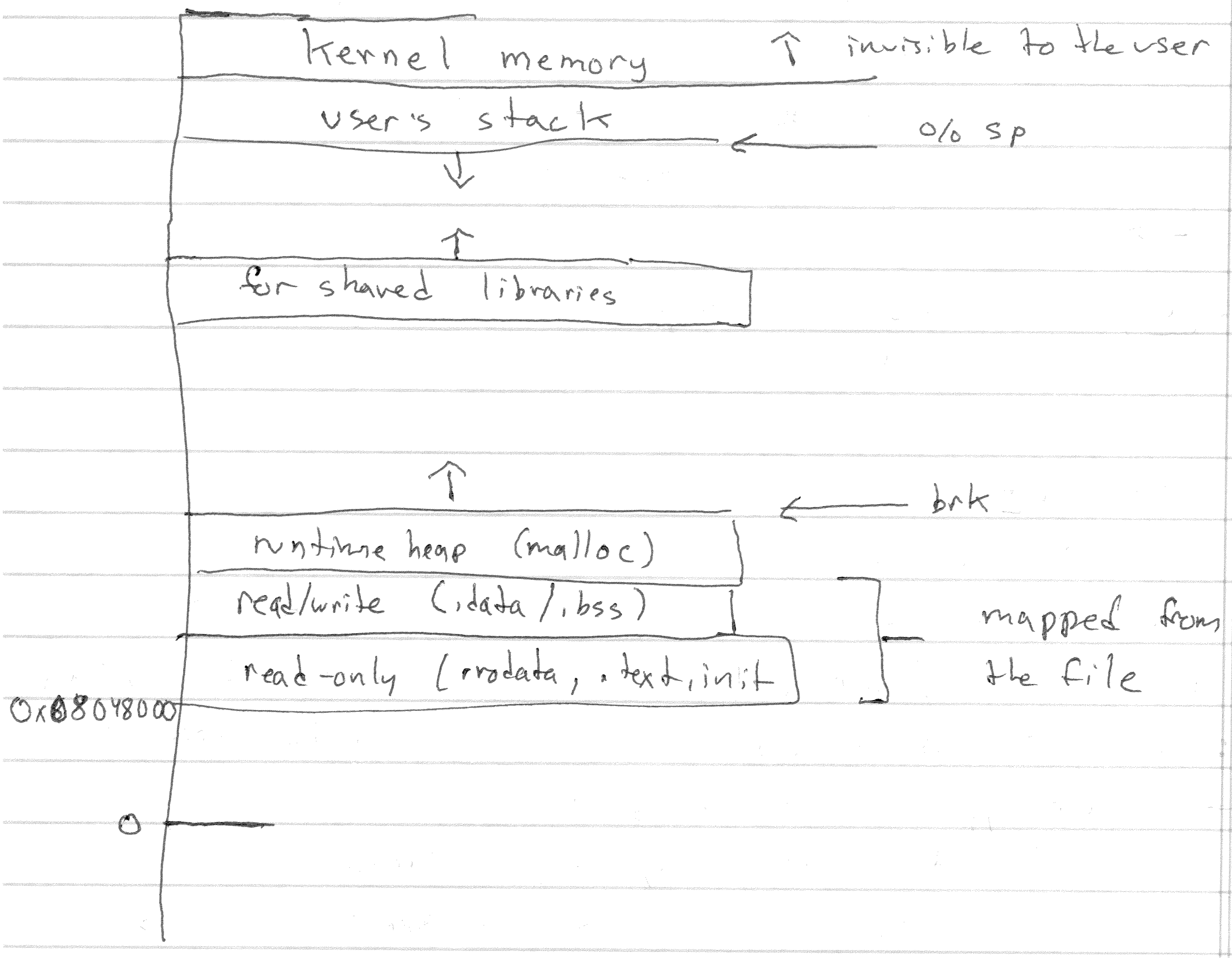
Segment header table — 'this' region of file goes to memory of type x

.init — a program put in by C
.text — program
.rodata — data
— read only

.data
.bss
— read/write

.symtab
.debug
.line
.strtab
section header table
— not loaded into memory

call __libc_init
call __os_init
call atexit
call _main ← you wrote
call _exit

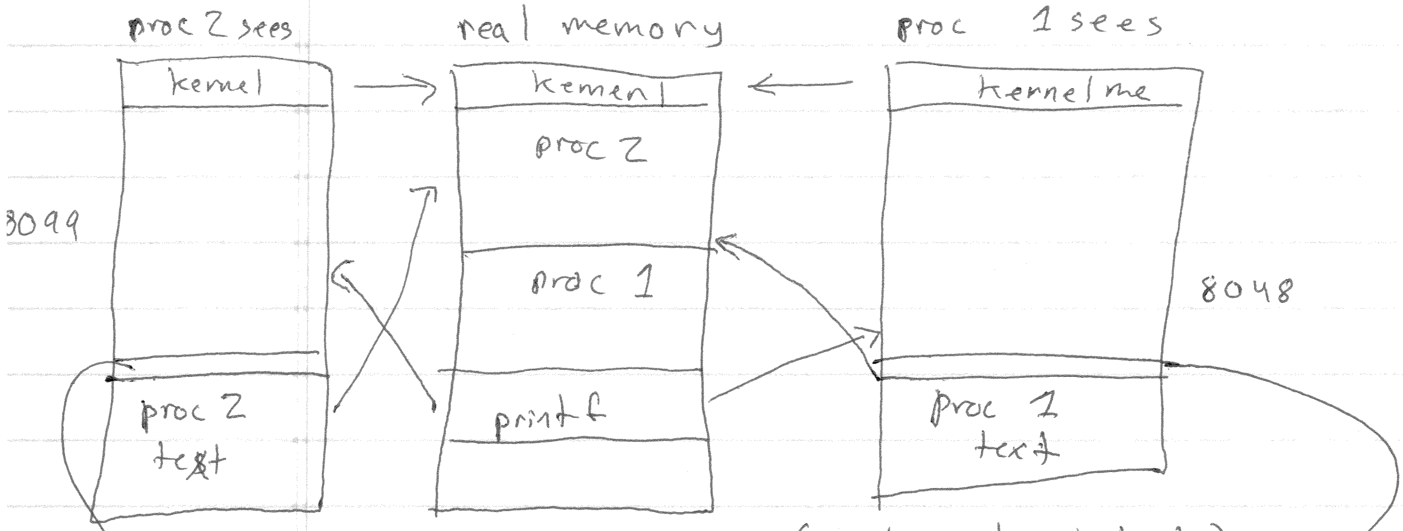


20-3

Shared Libraries

Save space on the disk
- └ the executable has a .interp region
  - └ a list of files (archives)

Save space on memory



printf must be ... PIC (position-independent code)
- uses relative refferences

→ Procedure Linkage Table ←

proc 1 will jump to PLT which jumps to printf

Dynamic Loading

dlopen — path to a shared library
+ returns a handle

dlsym — handle & symbol
+ returns a pointer