CS143A
Principles on Operating Systems
Discussion 05:

Instructor: Prof. Anton Burtsev

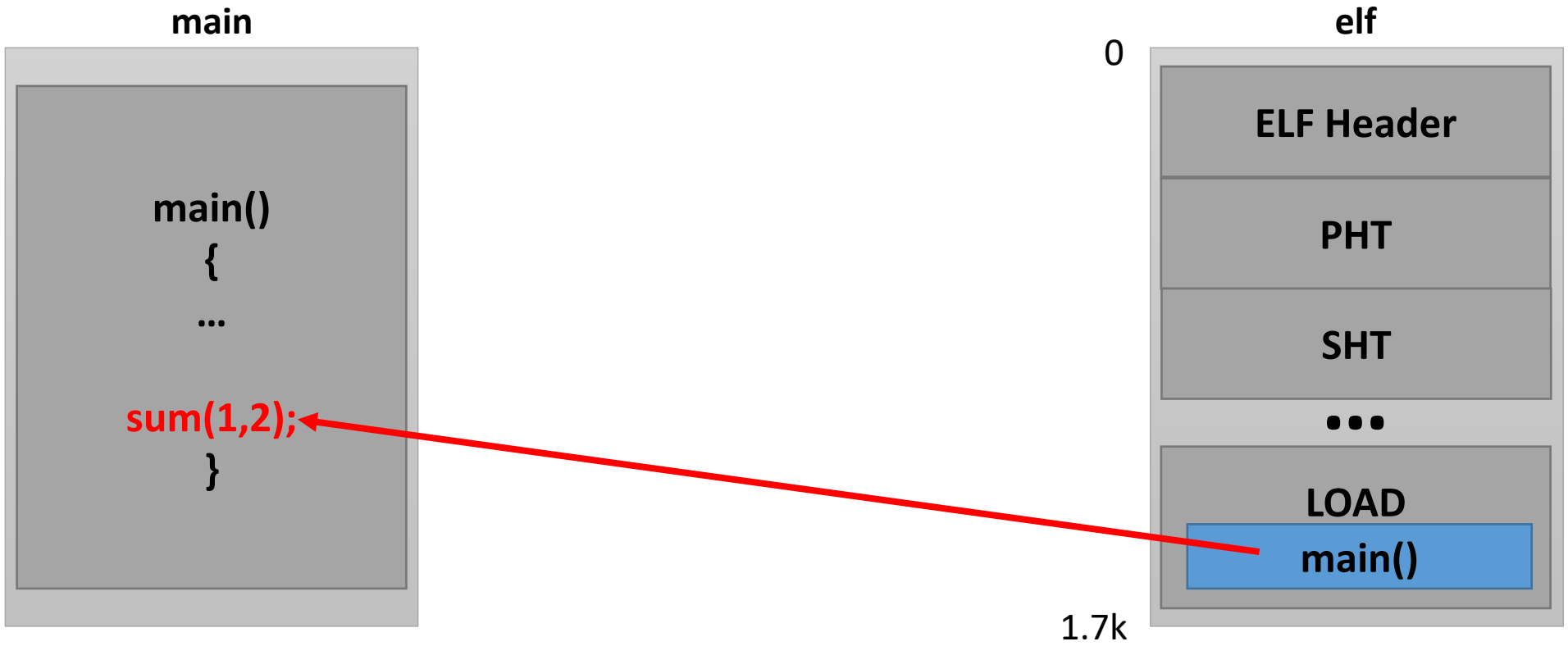
TA: Saehanseul Yi (Hans)

Nov 1, 2019 **1 PM**

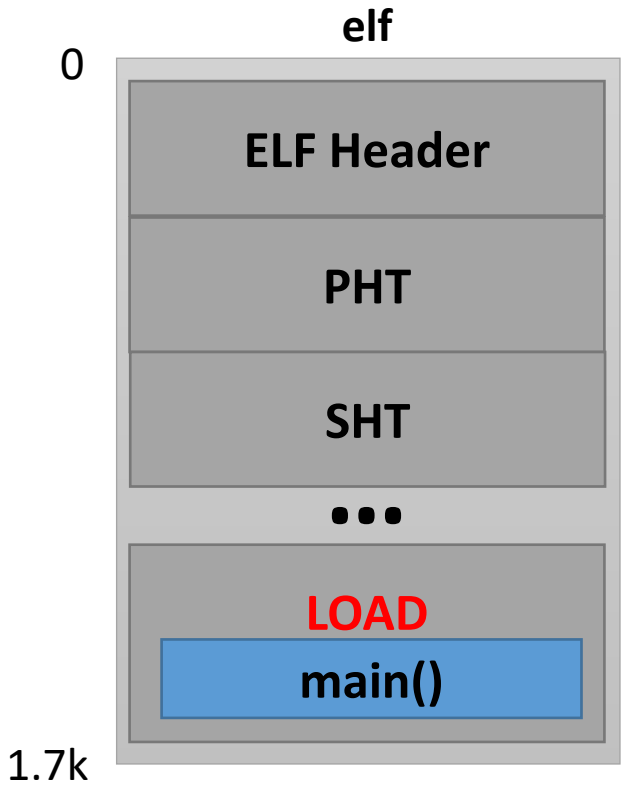
Agenda

- HW2 Walk-through
- Debugging seg fault

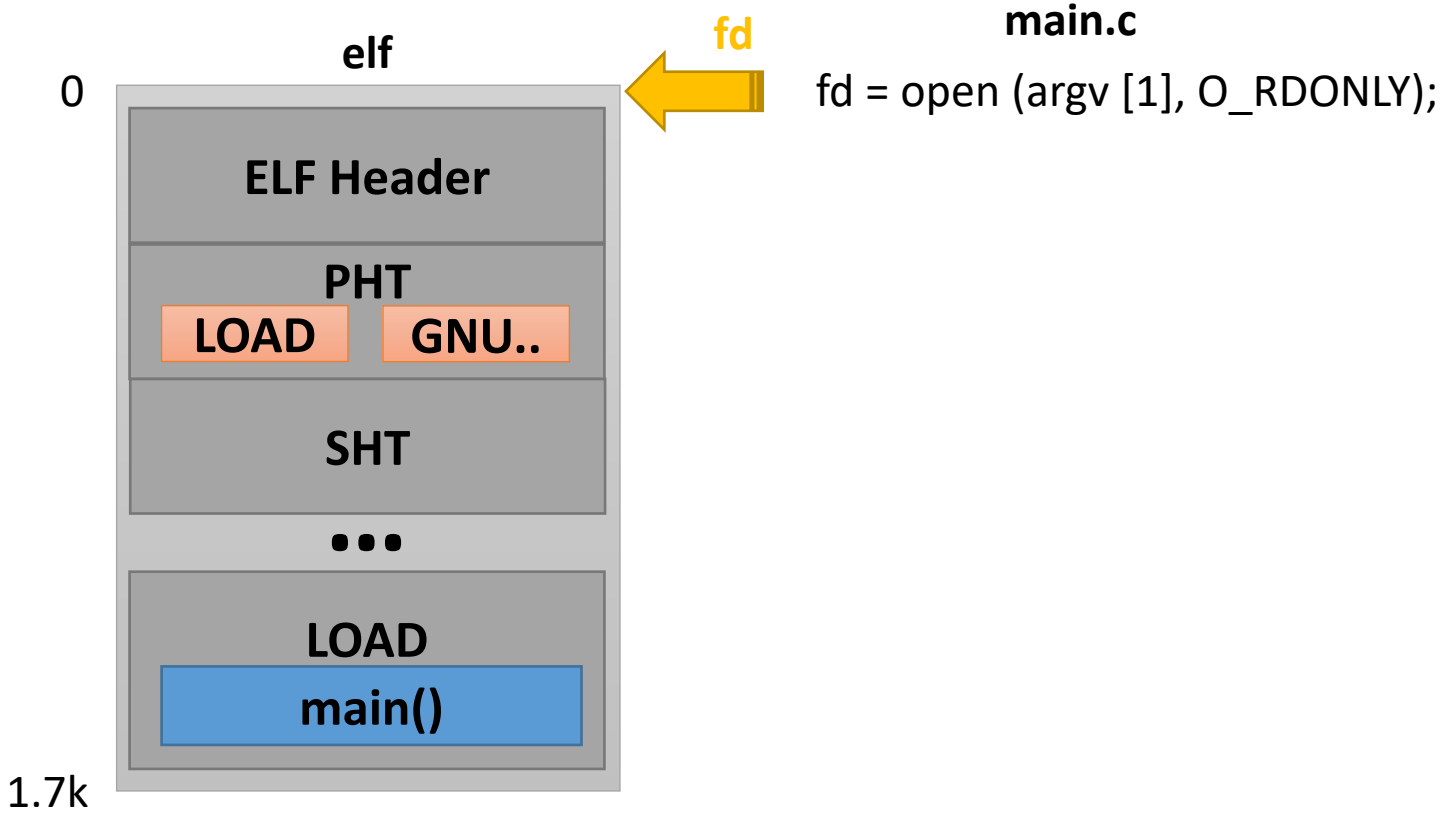
HW2-ELF



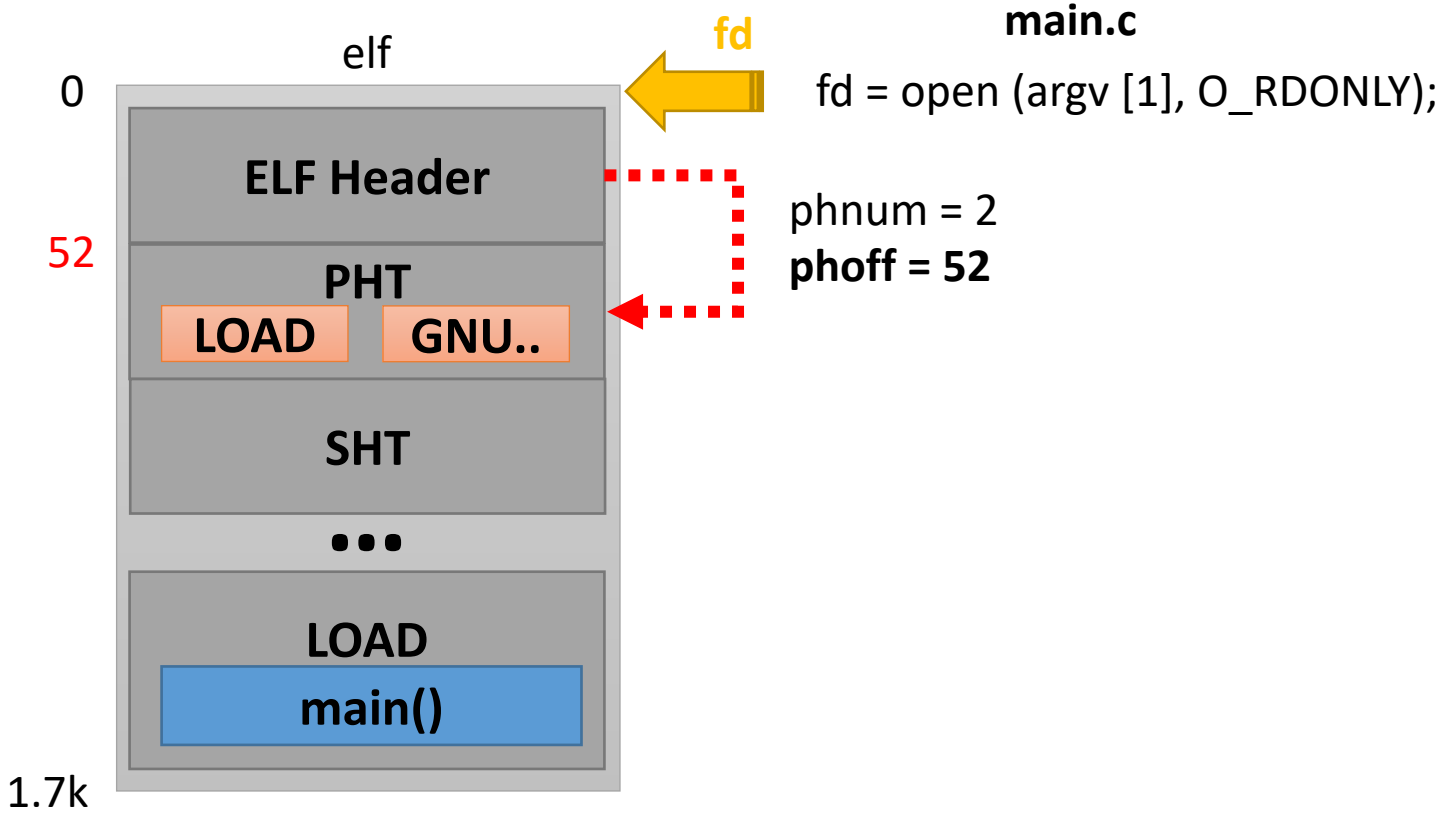
HW2-ELF



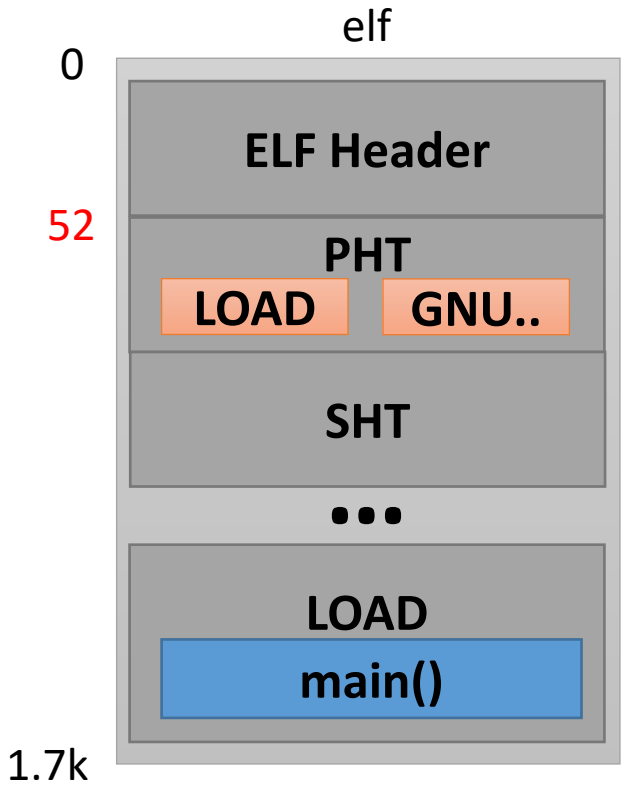
HW2-ELF


elf
fd
main.c
0
fd = open (argv [1], O_RDONLY);
ELF Header
PHT
LOAD
GNU..
SHT
LOAD
main()
1.7k

HW2-ELF


elf
0
52
1.7k
ELF Header
PHT
LOAD
GNU..
SHT
LOAD
main()
fd
main.c
fd = open (argv [1], O_RDONLY);
phnum = 2
phoff = 52

HW2-ELF



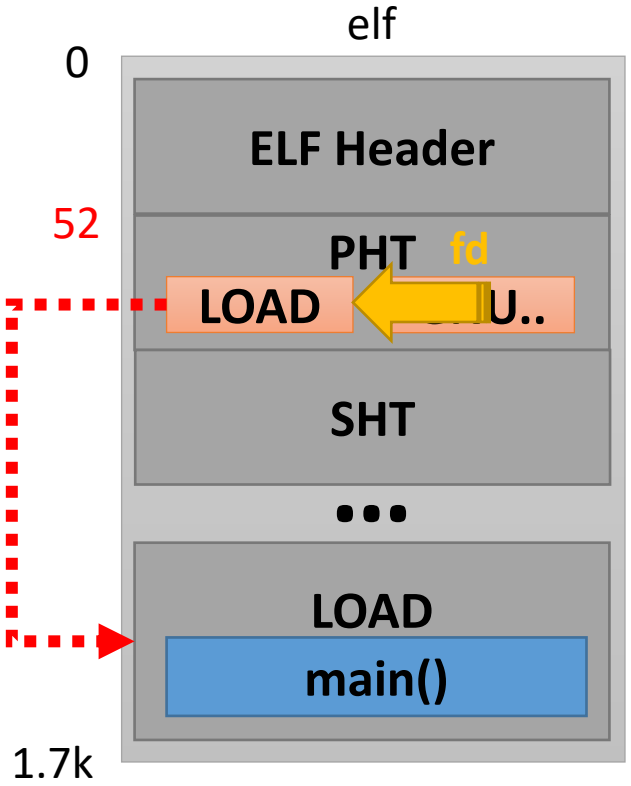
main.c

```
lseek(fd, elf.phoff, SEEK_SET);
```

fd

```
for(i = 0 ; i < elf.phnum; i++)
```

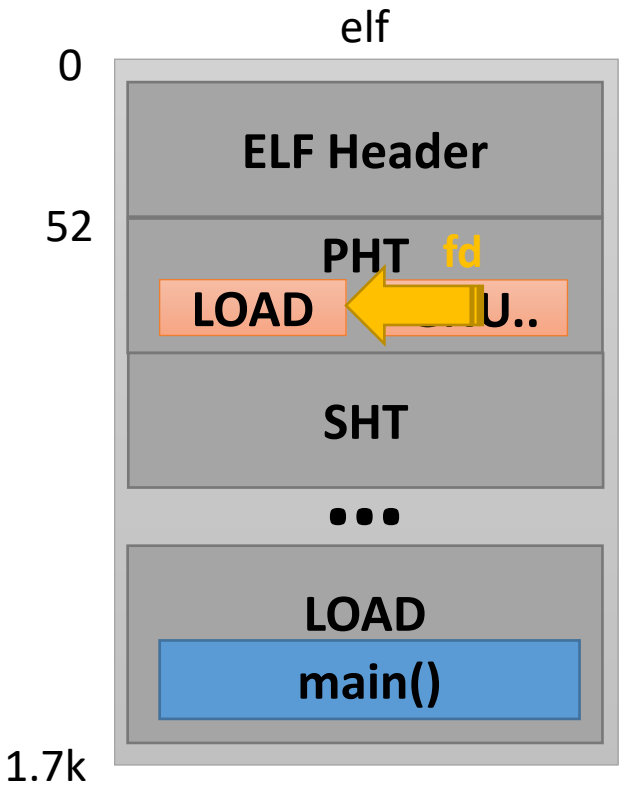
HW2-ELF



main.c

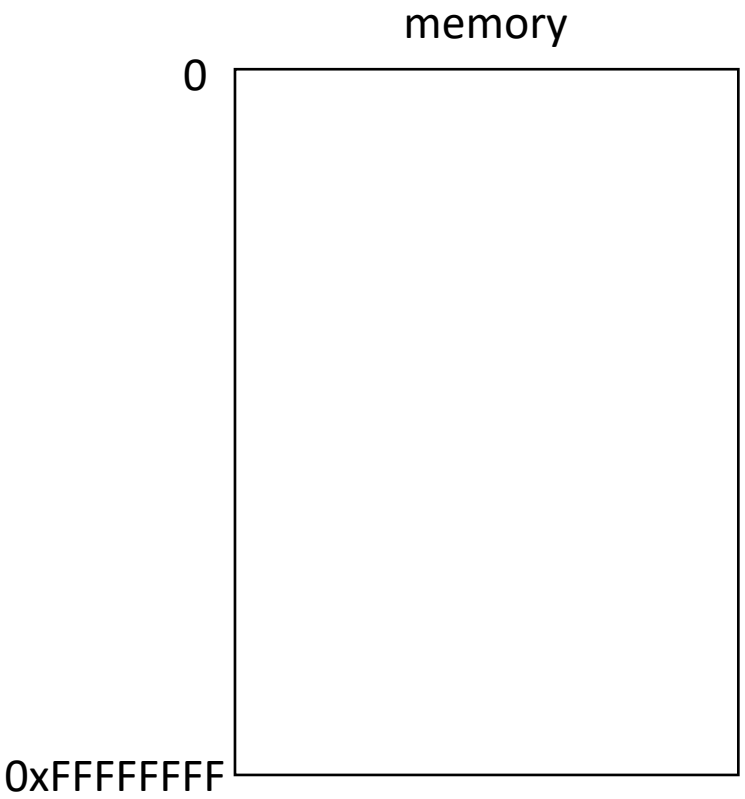
```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)
```

HW2-ELF

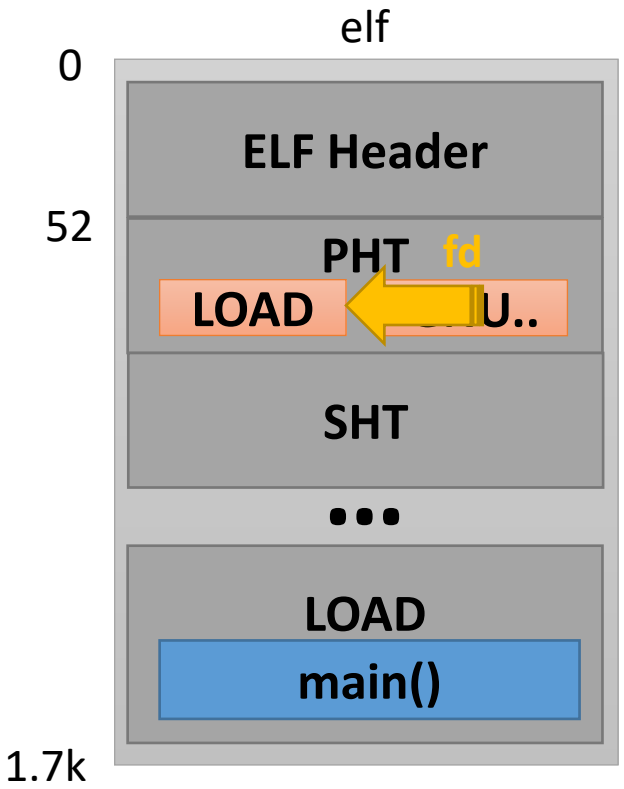


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(...)
```

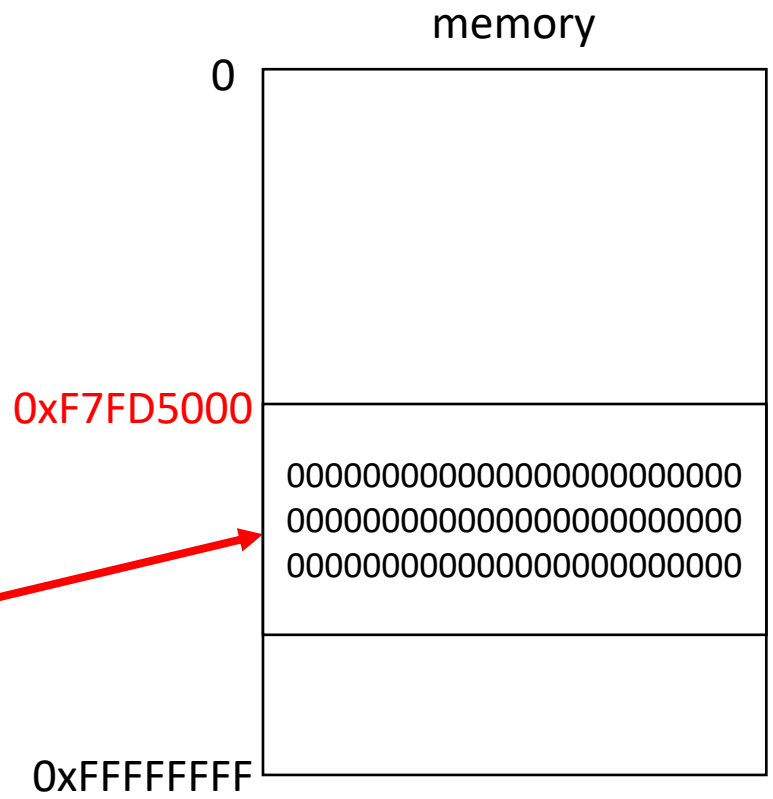


HW2-ELF

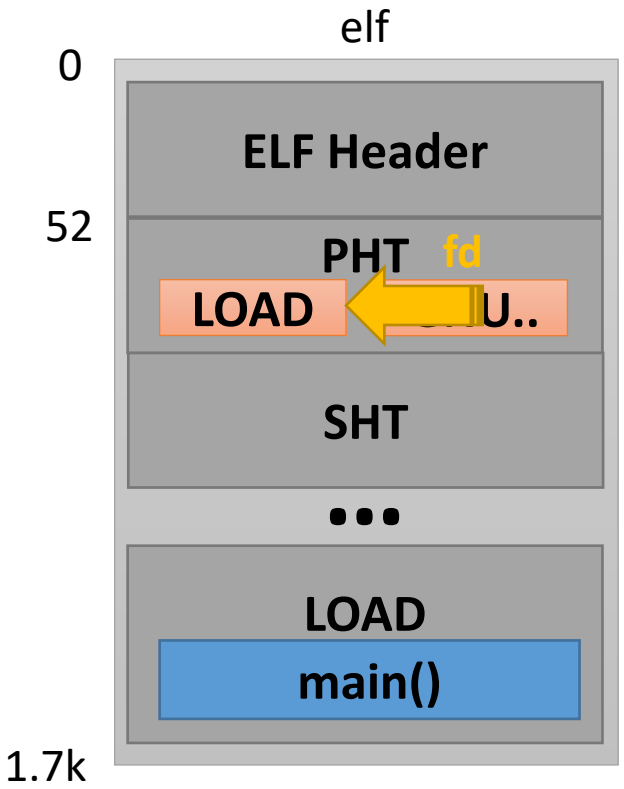


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(...)
```

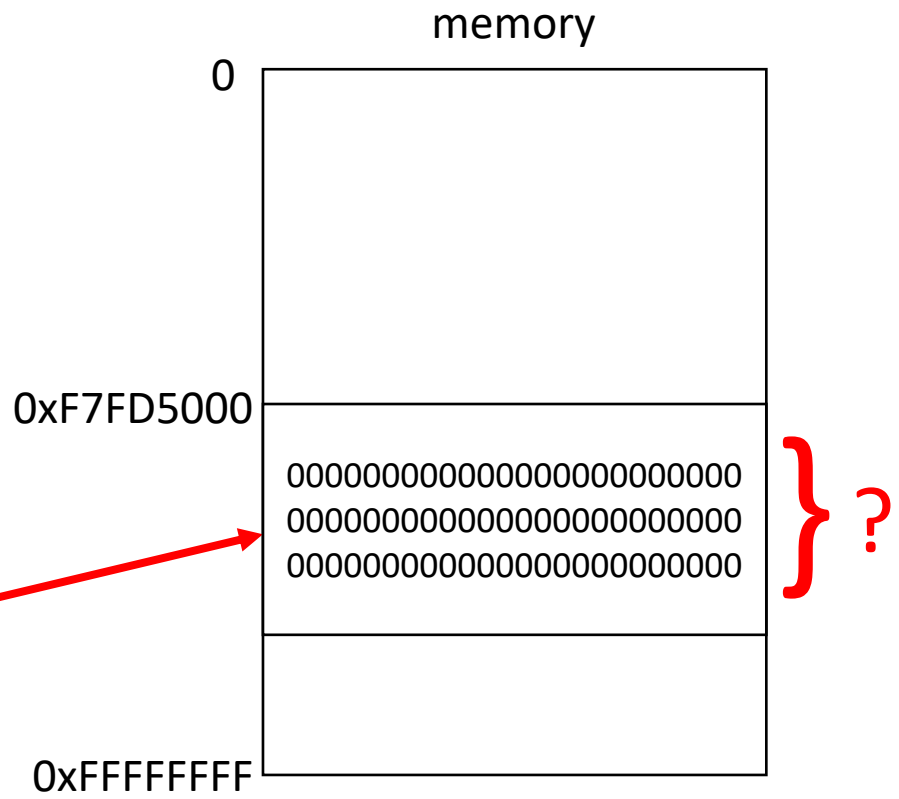


HW2-ELF

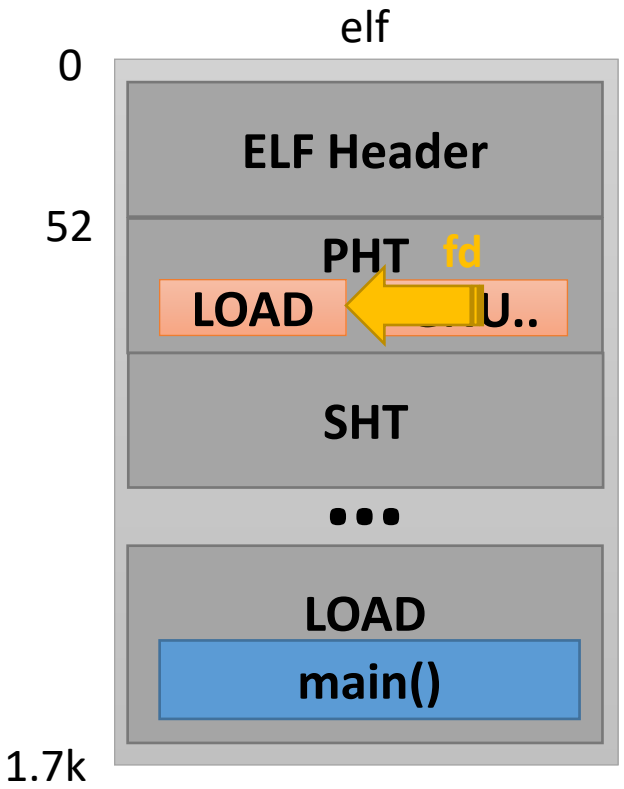


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(…)
```

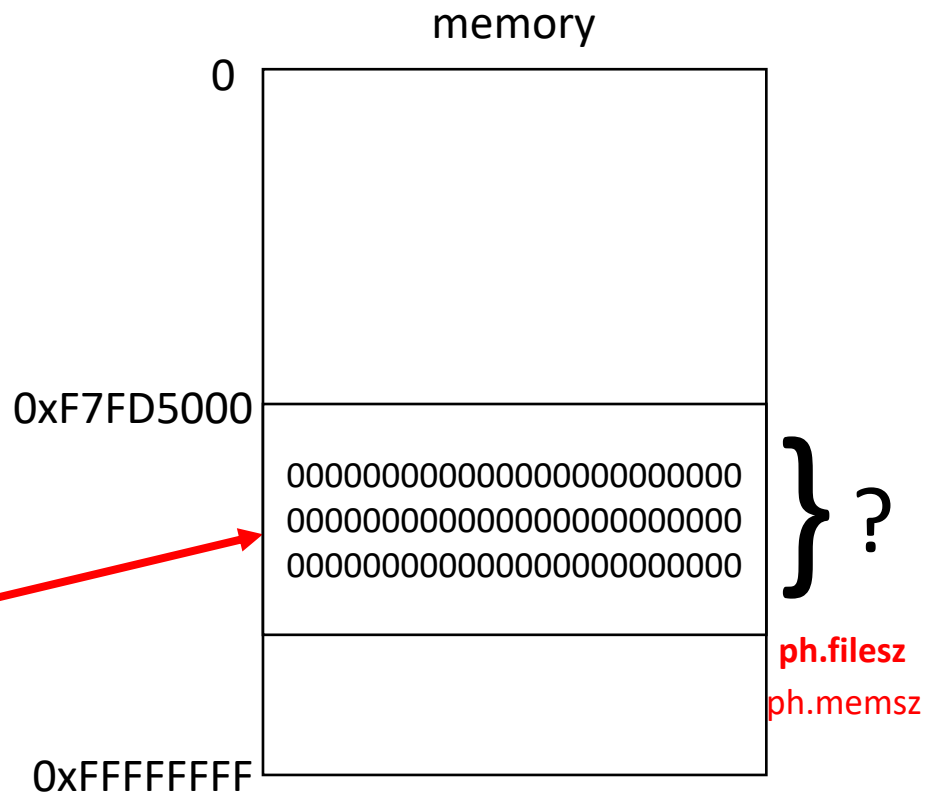


HW2-ELF

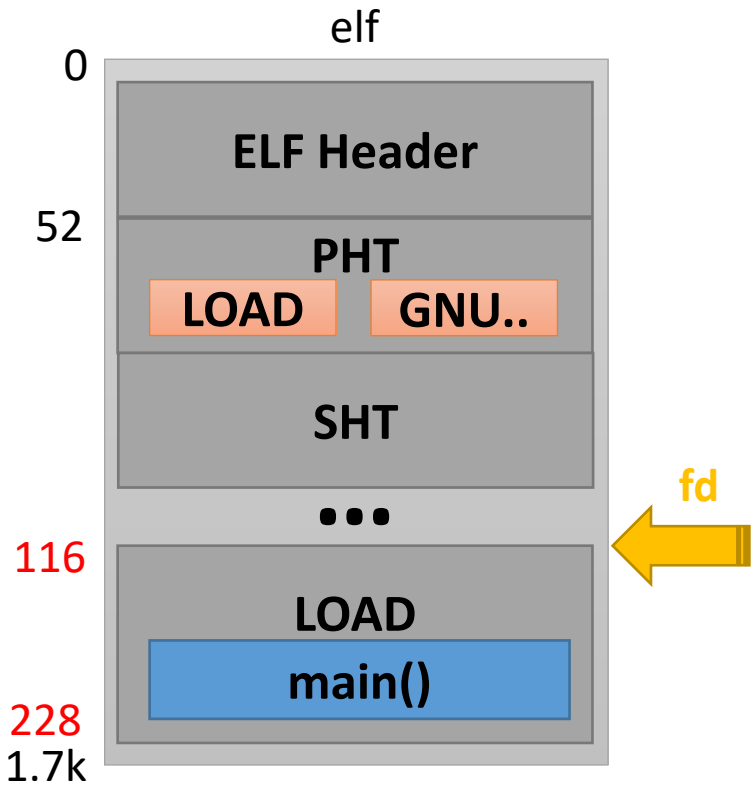


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(…)
```

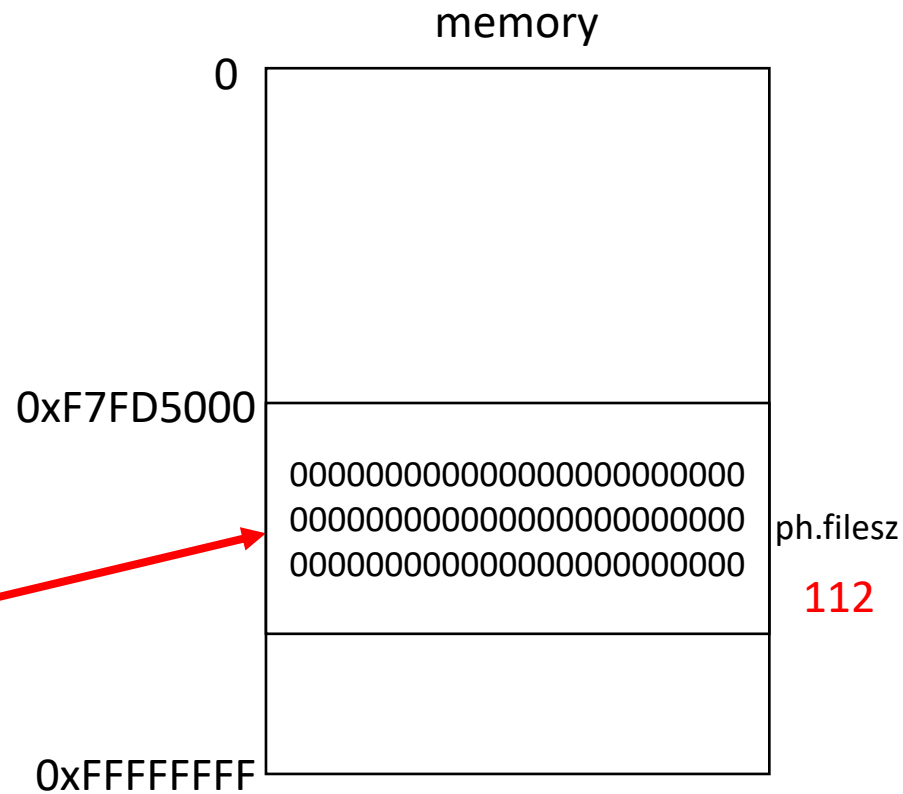


HW2-ELF

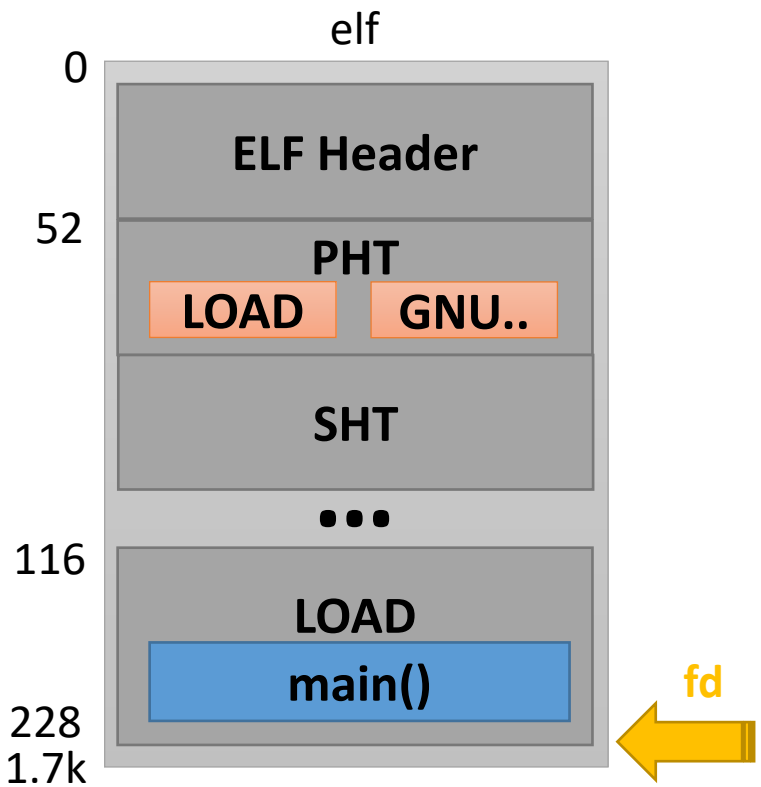


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(…)

lseek(fd, ph.off, SEEK_SET);
          116
ret = read(fd, code_va, ph.filesz);
```

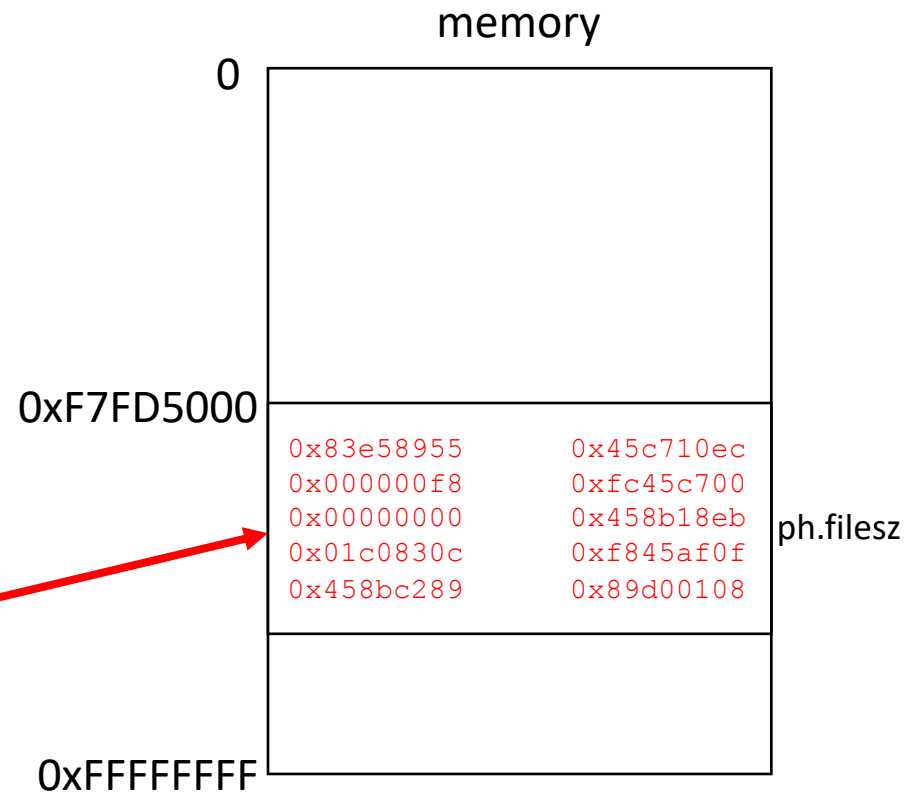


HW2-ELF

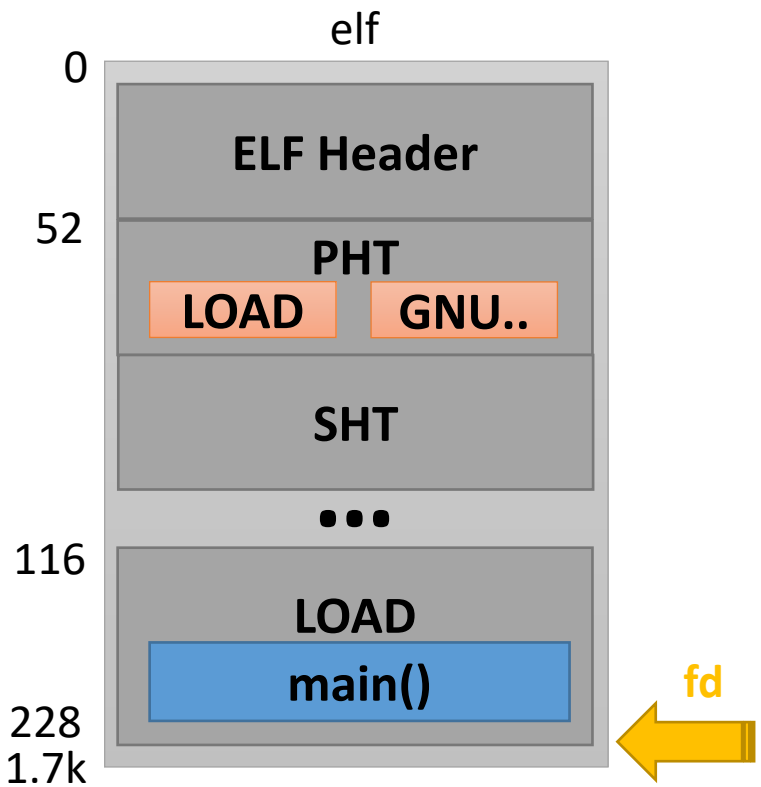


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(…)

lseek(fd, ph.off, SEEK_SET);

ret = read(fd, code_va, ph.filesz);
```

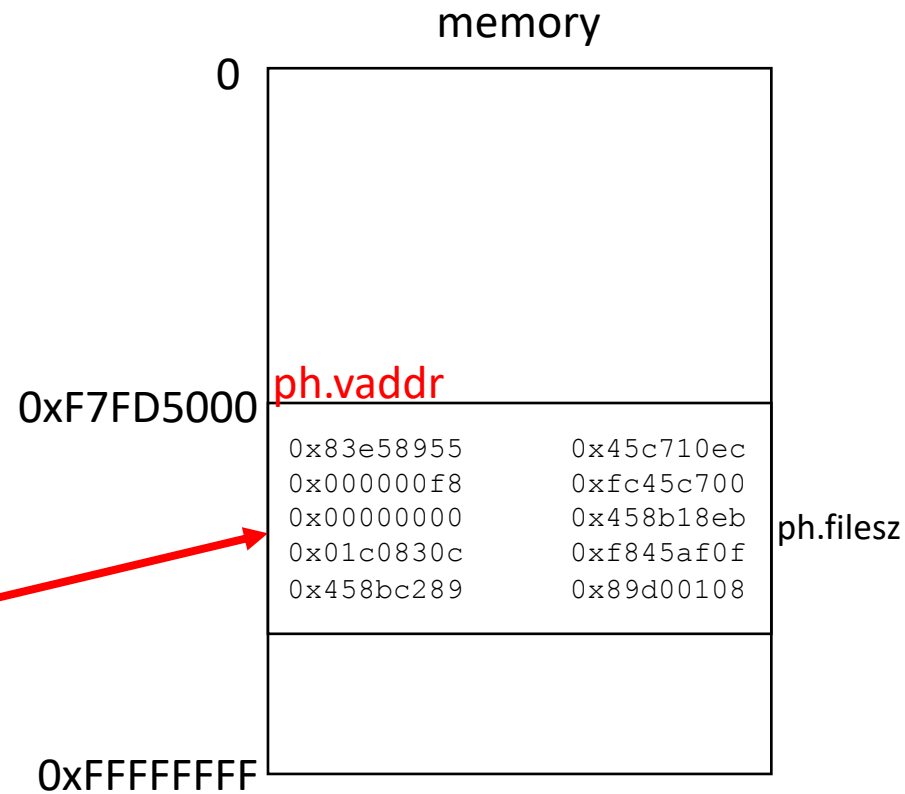


HW2-ELF

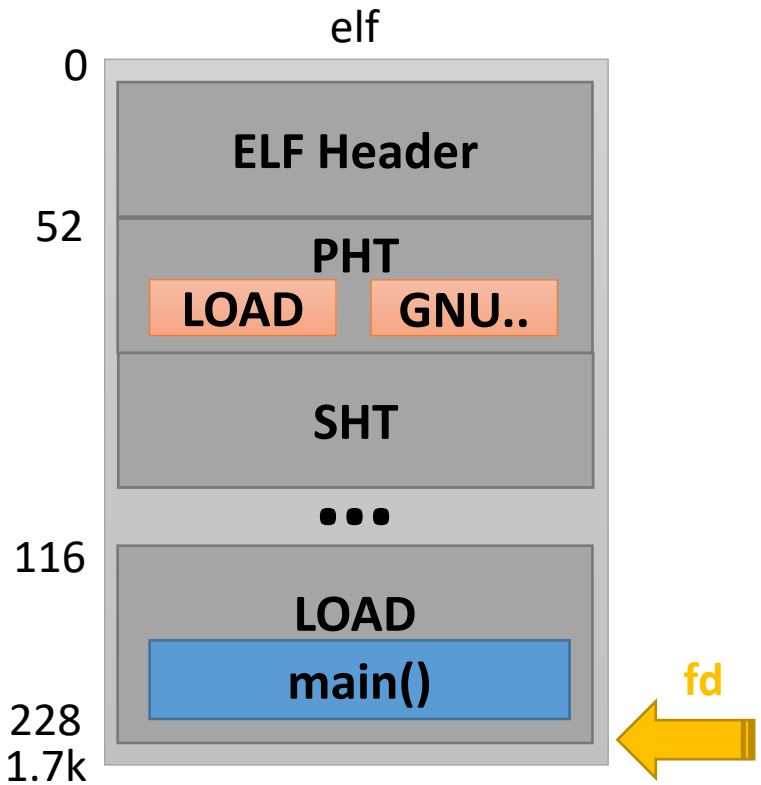


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(…)

lseek(fd, ph.off, SEEK_SET);

ret = read(fd, code_va, ph.filesz);
```

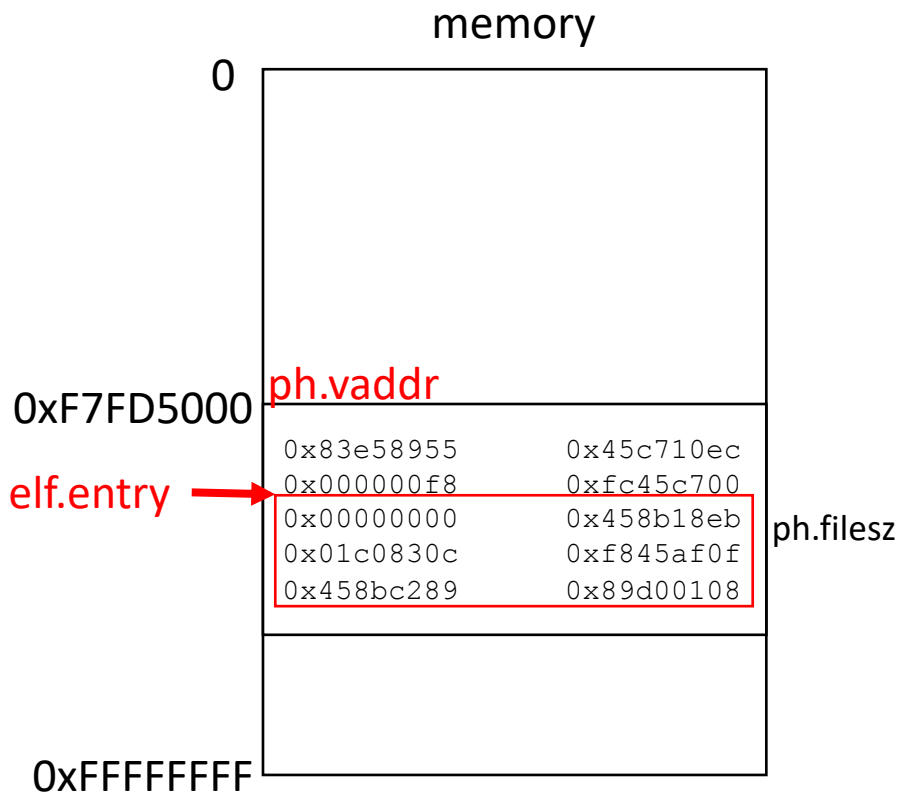


HW2-ELF

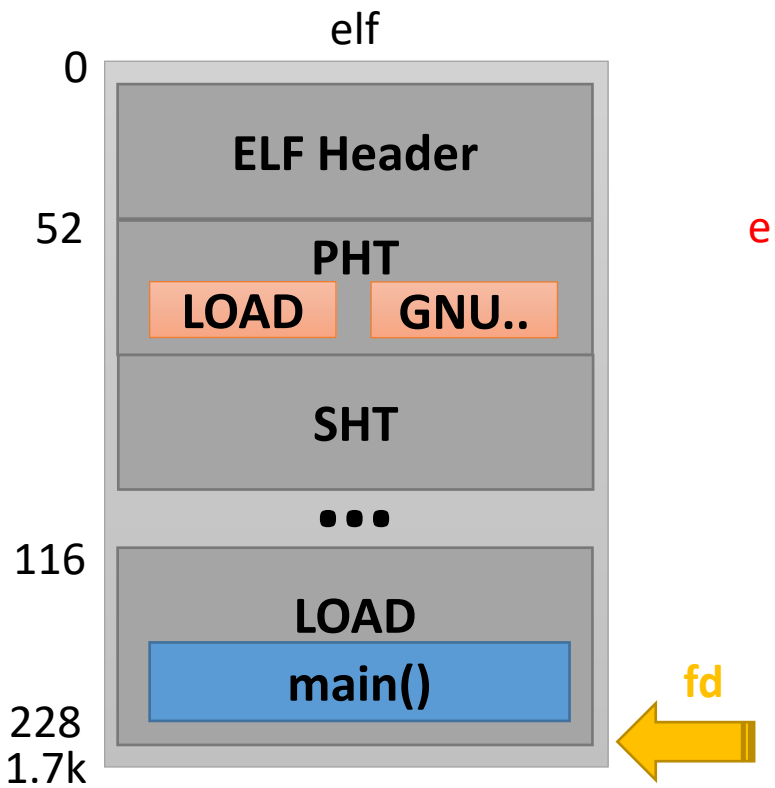


main.c

```
lseek(fd, elf.phoff, SEEK_SET);

for(i = 0 ; i < elf.phnum; i++)

ret = read(fd, (char*)&ph, sizeof(ph));

if(ph.type == ELF_PROG_LOAD)

code_va = mmap(…)

lseek(fd, ph.off, SEEK_SET);

ret = read(fd, code_va, ph.filesz);
```

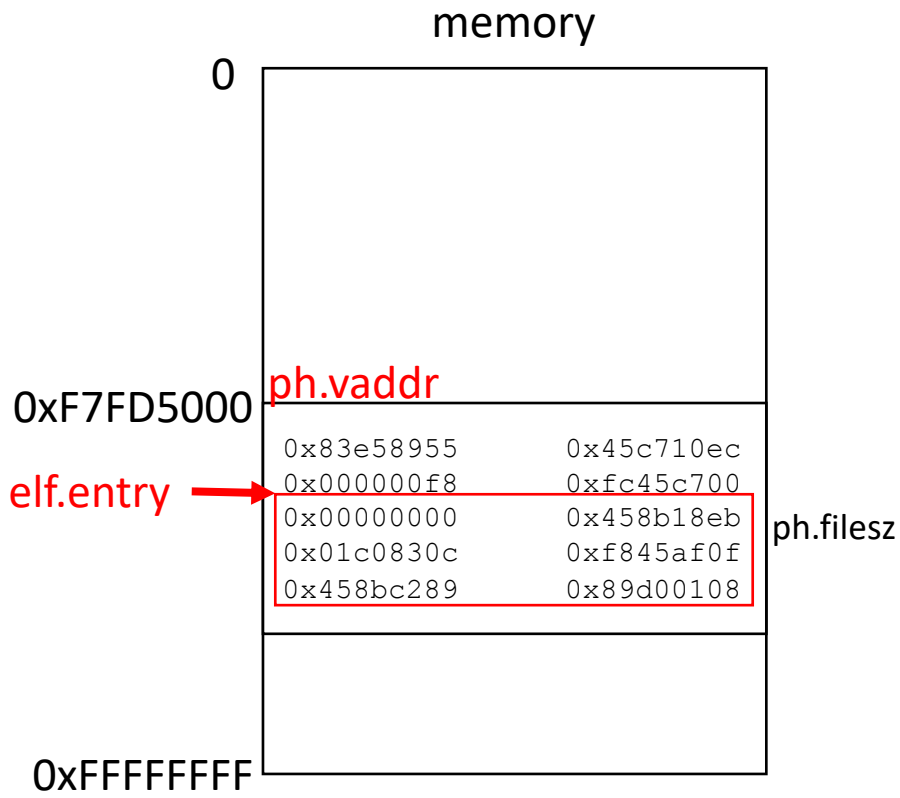


HW2-ELF



main.c

entry = code_va + (elf.entry - ph.vaddr));



In our homework, both elf.entry and ph.vaddr are 0
The picture is intentionally wrong for general cases.

Debugging Segmentation Fault

- First, where did it happen?
  Backtrace

Debugging Segmentation Fault

- First, where did it happen?
  Backtrace
- If you can locate, you're lucky
  Otherwise, follow the code

Debugging Segmentation Fault

- First, where did it happen?
  Backtrace
- If you can locate, you're lucky
  Otherwise, follow the code
- Check memory contents, variables, …