```
.text
main:
  push  %ebp
  mov   %esp, %ebp
  push  $6
  call  fib
  add   $4, %esp
  leave
  ret

.data
global_var:
  .long 23
```

Sections

.text

```
55 89 e5 6a 06    AX
e8 00 00 00 00
83 c4 04 c9 c3
```
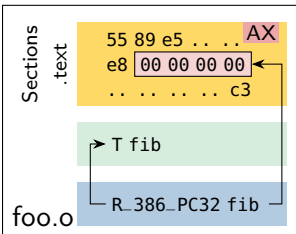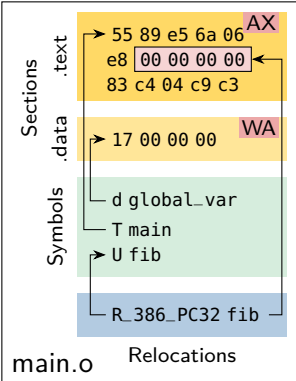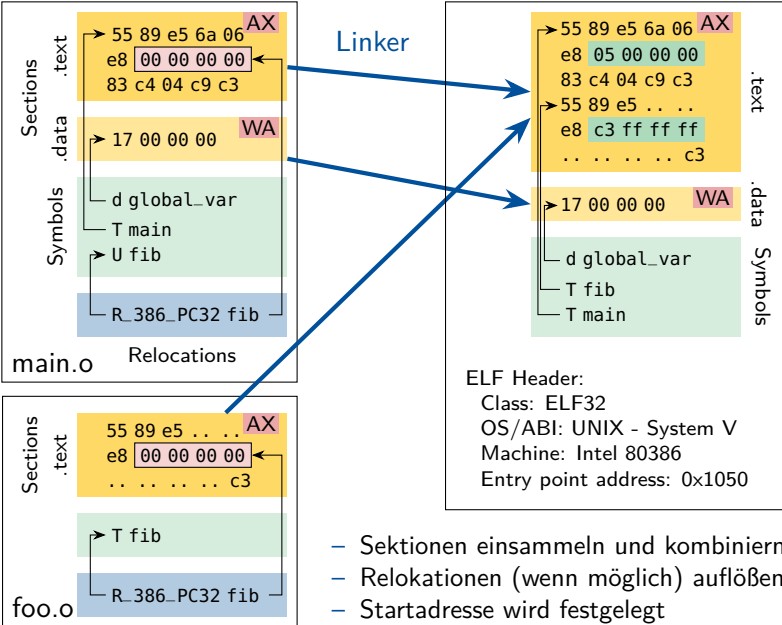
.data

```
17 00 00 00    WA
```

Symbols

```
d global_var
T main
U fib
```

Relocations

```
R_386_PC32 fib
```

main.o

main.o

Sections
- .text
  - `55 89 e5 6a 06` `AX`
  - `e8` `00 00 00 00`
  - `83 c4 04 c9 c3`
- .data
  - `17 00 00 00` `WA`

Symbols
- `d global_var`
- `T main`
- `U fib`

Relocations
- `R_386_PC32 fib`

foo.o

Sections
- .text
  - `55 89 e5 .. ..` `AX`
  - `e8` `00 00 00 00`
  - `.. .. .. .. c3`

- `T fib`
- `R_386_PC32 fib`

Linker

.text
- `55 89 e5 6a 06` `AX`
- `e8` `05 00 00 00`
- `83 c4 04 c9 c3`
- `55 89 e5 .. ..`
- `e8` `c3 ff ff ff`
- `.. .. .. .. c3`

.data
- `17 00 00 00` `WA`

Symbols
- `d global_var`
- `T fib`
- `T main`

ELF Header:
  Class: ELF32
  OS/ABI: UNIX - System V
  Machine: Intel 80386
  Entry point address: 0x1050

– Sektionen einsammeln und kombinieren
– Relokationen (wenn möglich) auflösen
– Startadresse wird festgelegt