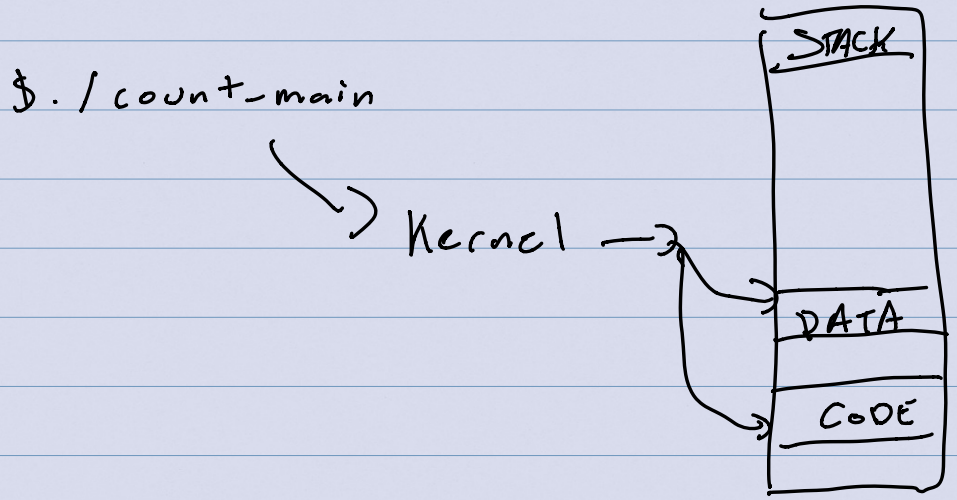
Project04 - remaining details
Project03 offer

ELF

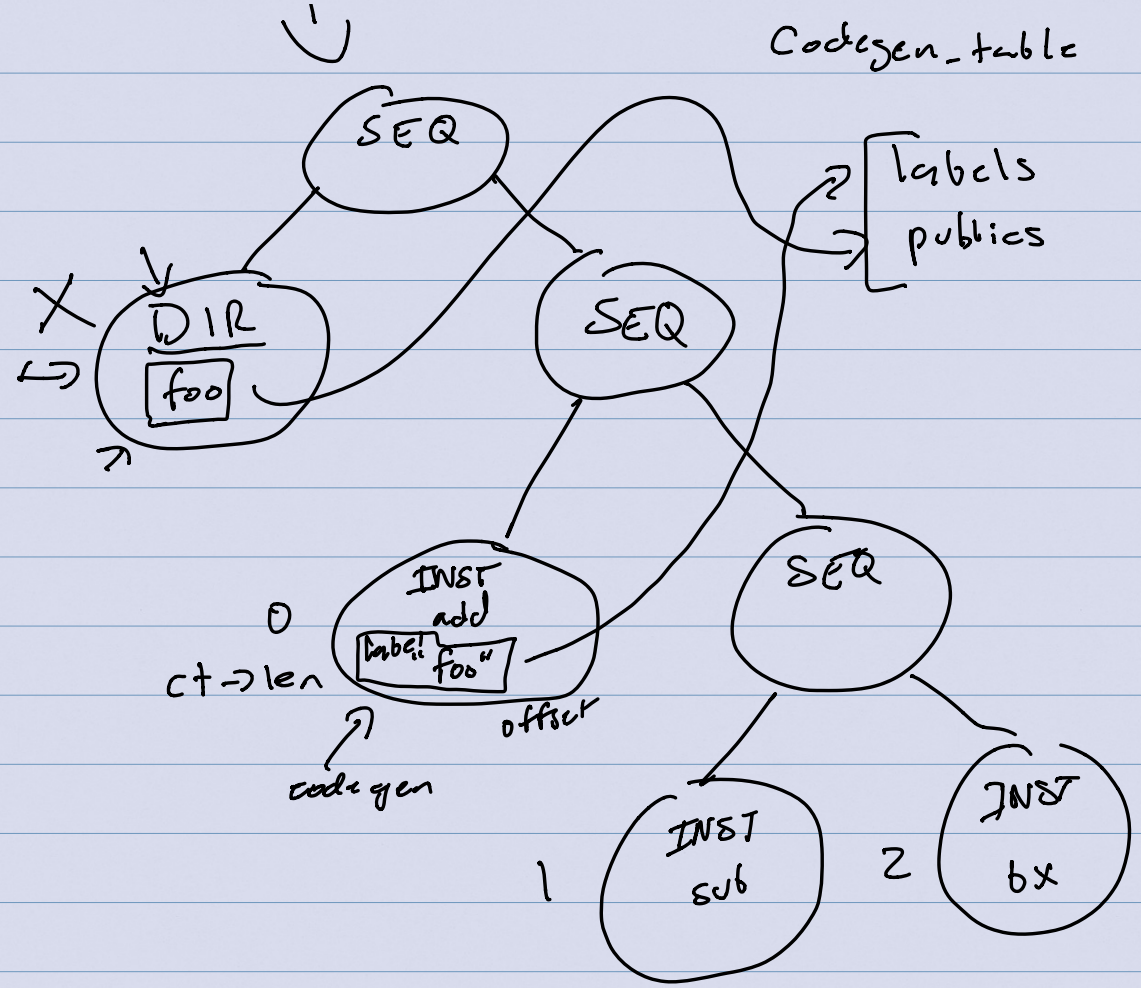
object files .o → gcc, linker
executables



---

armasm and ELF object files

```
.global foo
foo:
0       add
1       sub
2       bx
```



codegen_b (ct , np)

→ codegen_get_index(tree, label, 0)

Offsets

when we add global dir label to publics → offset should be 0.

when we add label to labels array we need index/offset (byte or word)

ELF needs byte offsets

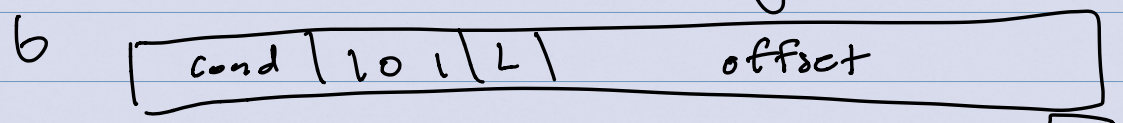
codegen_b(ct, np)

→ codegen_get_index(tree, label, 0)

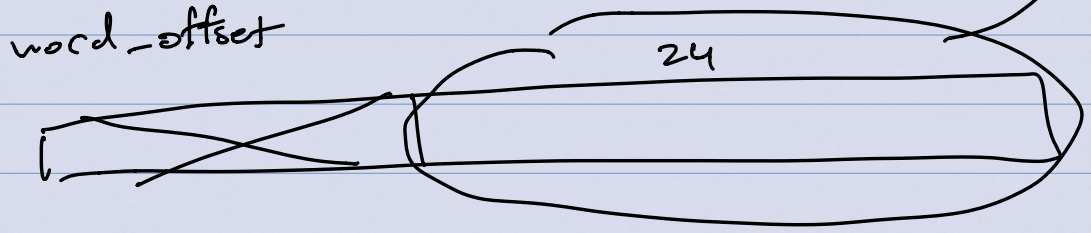
index = ct->len;

target_index =

uint32_t int32_t
word_offset = target_index - (index + 2)



24 bit 2's comp word offset



word_offset = word_offset & 0xFFFFFF

cmp → 2 operands, does not use rd
rd set to ∅

S bit to 1 CC to be set

cmp r0, r1
cmpi r0, #2

mov r0, #-1 → mvn r0, #00

mov

int32_t imm

imm

00

0000 0000 ←

31 1 1 1 1 ..... 1 1 1 1 1 1 1 1 1

→ (-1) ←

mov r0, #-2 → mvn r0, #01

0000 0001

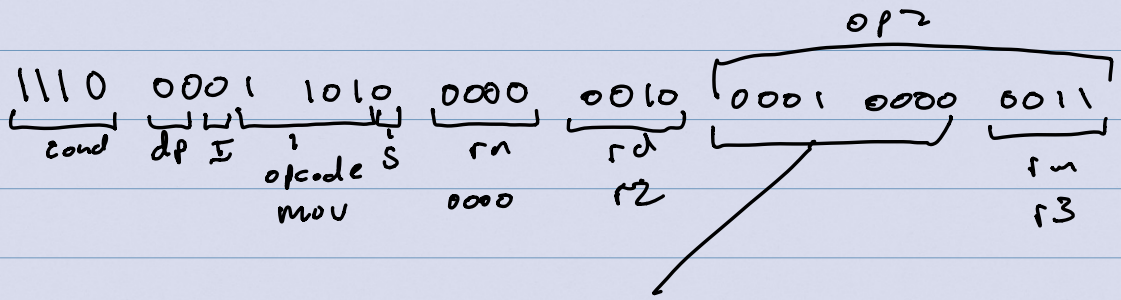
1 1 1 1 ..... 1 1 1 1 1 1 1 1 1 1 1 0 (-2) ←

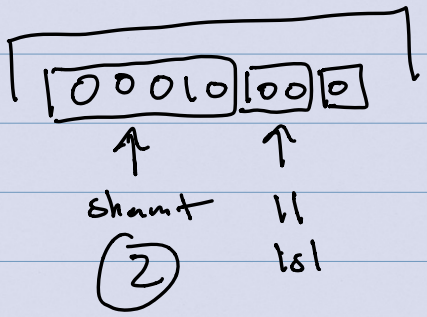
lsl, lsr

rd rm

lsl r2, r3, #2

E1A0 2103 DP → mov





lsl

lsr