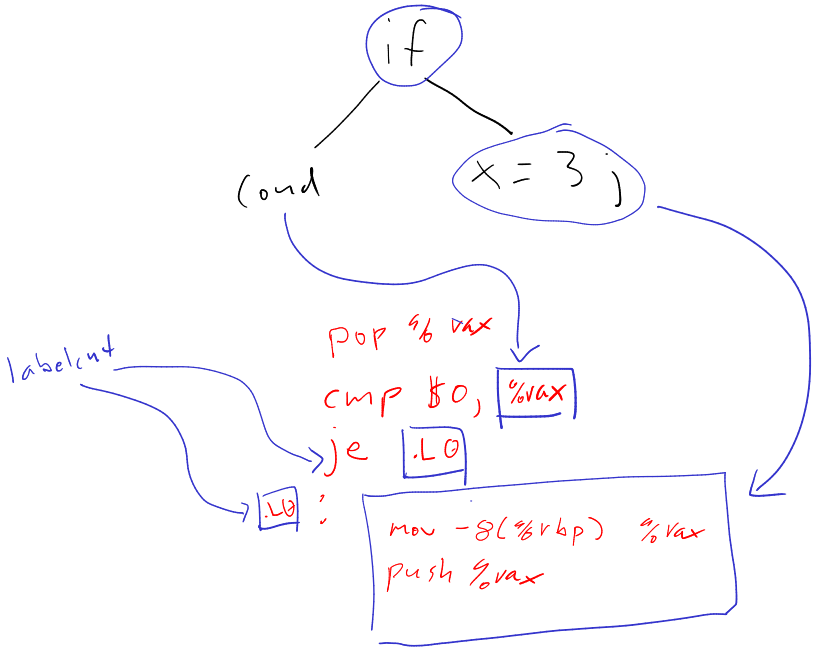
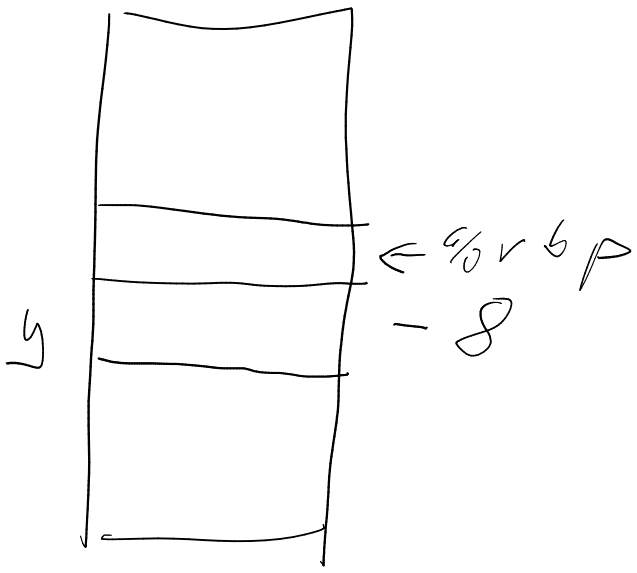

if
(cond
x = 3)
labelcnt
pop %rax
cmp $0, %rax
je .L0
.L0 :
mov -8(%rbp) %rax
push %rax

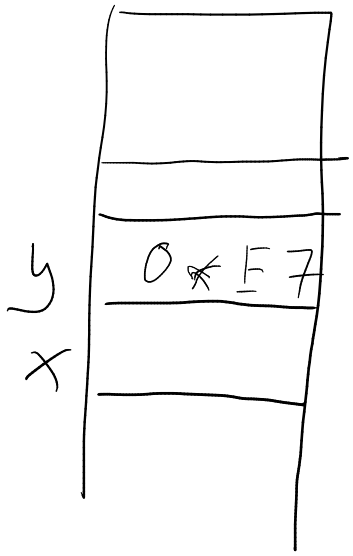


source dest

mov $3 -8(%rbp)

3 [mov $3 %rax
push %rax]

y = 1 pop %rax
mov %rax -8(%rbp)



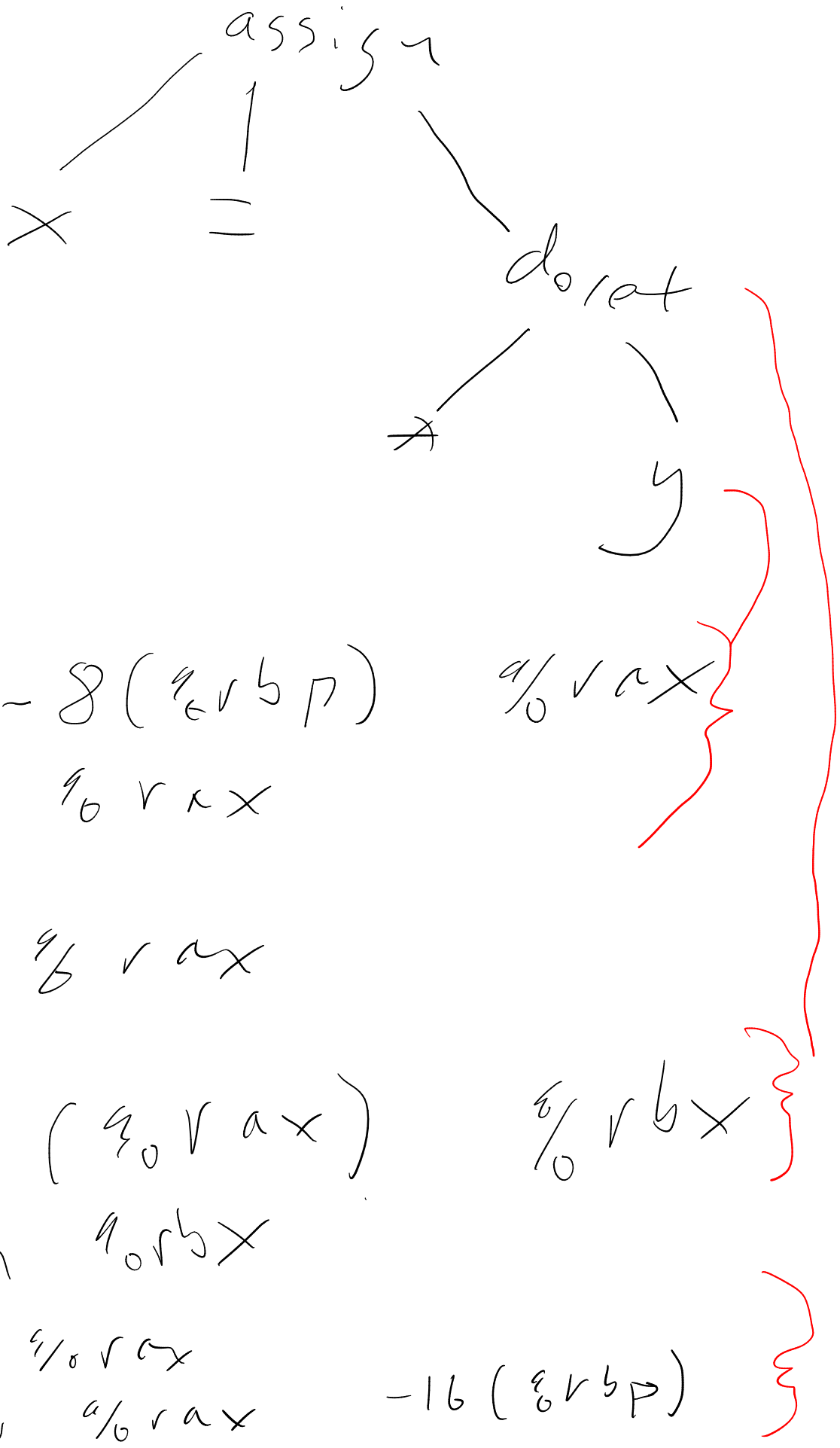
y mov -8(%rbp) 0xE7 %rax

*y(%rax)

x = *y

mov (%rax) %rbx

mov %rbx -16(%rbp)


assign
x
=
deref
*
y
mov -8(%rbp) %rax
push %rax
pop %rax
mov (%rax) %rbx
push %rbx
pop %rax
mov %rax -16(%rbp)

x = 2

cmp $0 cond

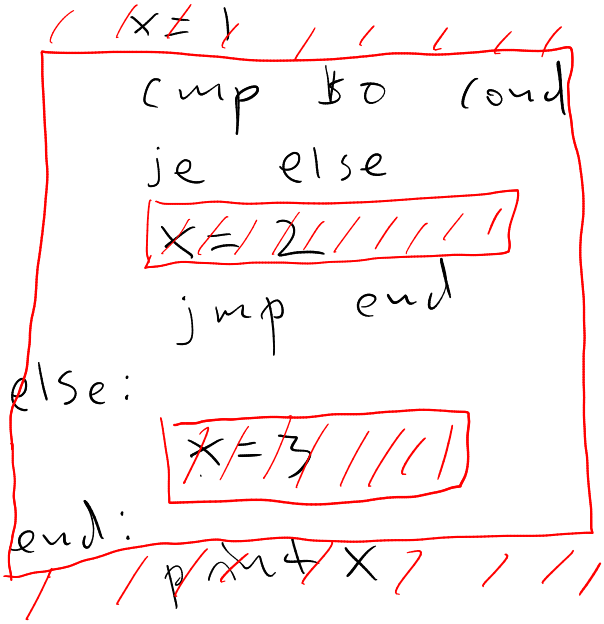
je end

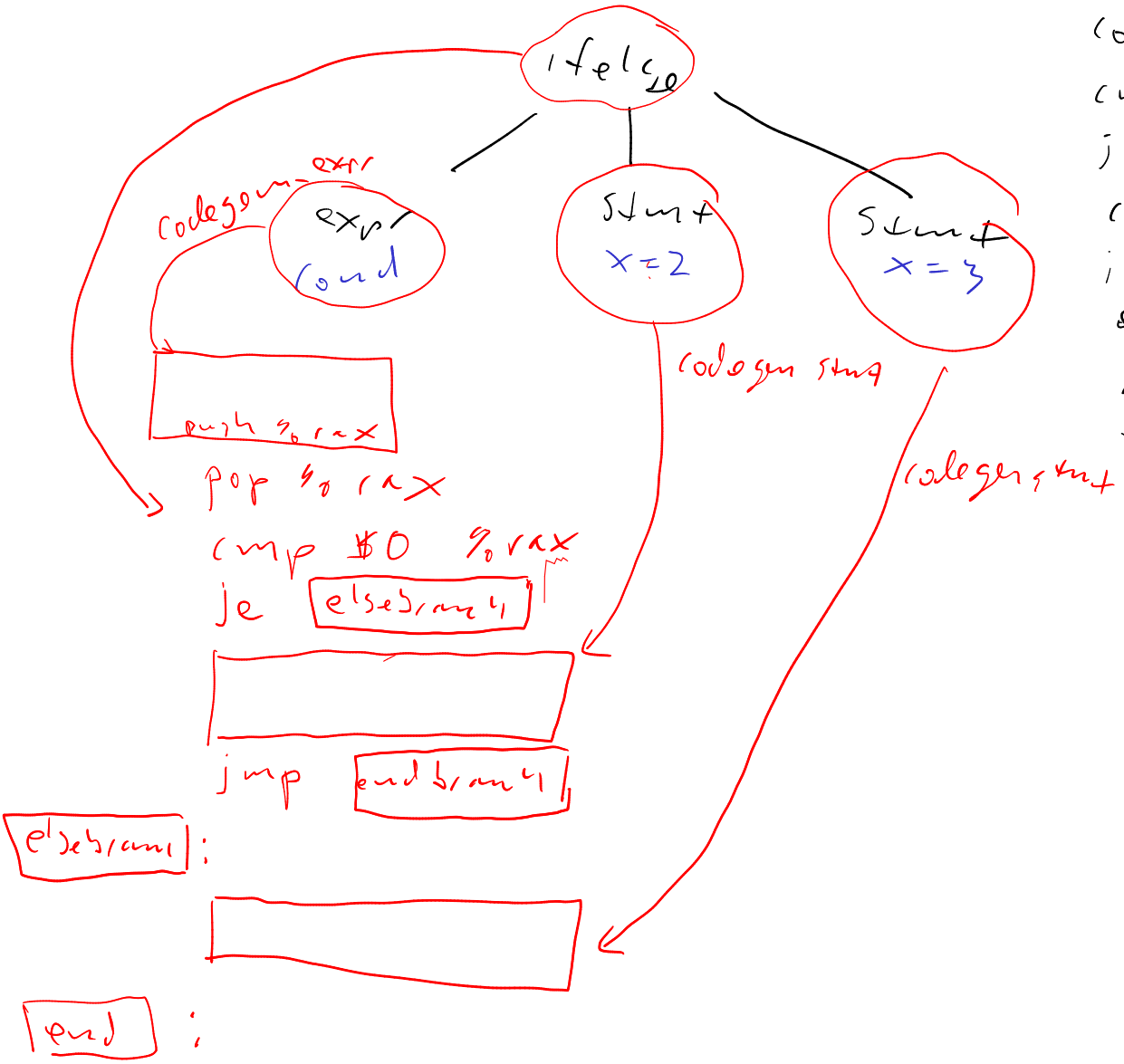
x = 3
jmp end

end:

half, if

```
x = 1
if (cond)
    x = 2
else
    x = 3
print x
```





codegen(expr)
cmp
je elsebranch
codegen(stmt1)
jmp endlabel
elselabel:
codegen(stmt2)
end:

```
head:
    cmp $0 cnt
    je end
    cnt = cnt - 1
    jmp head
end:
```

```
if (A)
    if (B)
        x=1
```

```
cmp $0 A
je end

cmp $0 B
je end

x=1
end.
```

```
if((A && B))
    result = 1
else
    result = 0
```

codegen_expr(lhs)
codegen_expr(rhs)

```
pop %rbx
pop %rax
cmp $0 %rax
je false

cmp $0 %rbx
je false
mov $1 %rax
jmp end
false:
mov $0 %rax
end:
push %rax
```

OR