

# Select instructions

Thursday, 24 February 2022 8:31 AM

1) (if (eq? atm1 atm2))  
↳ goto l1  
else goto l2

→ cmpq, atm2 atm1  
je l1  
jmp l2

2) var = (eq? atm1 atm2)  
↑  
(let [var (eq? atm1 atm2)] → sete %al  
body ) movzbq %al var

## Patch instruction

1) movzbq → target has to be register

movzbq %al var → reg.

↳ movzb %al r2

movq r2 +8(%rsp)

2) cmpq → 2<sup>nd</sup> immediate

# Register allocation

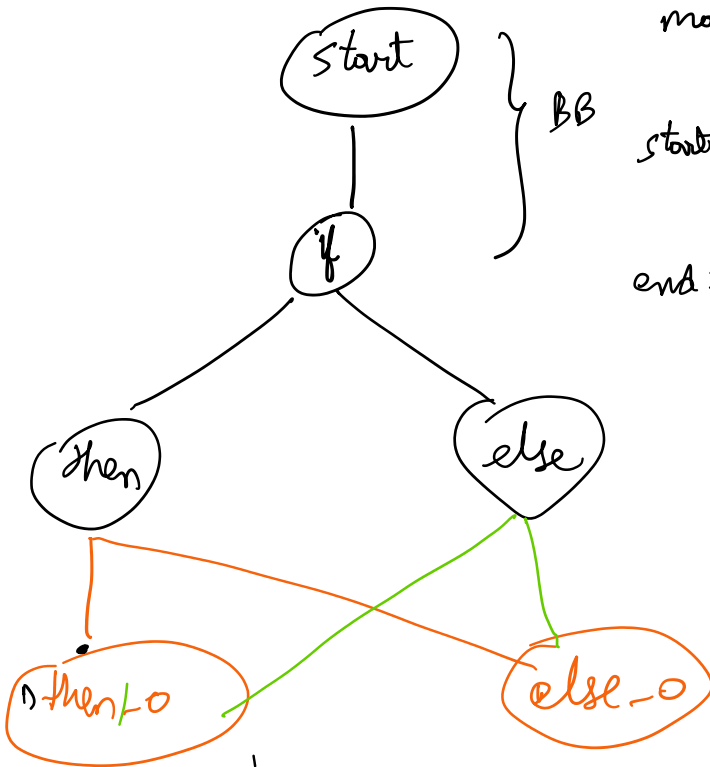
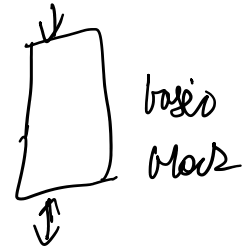
Thursday, 24 February 2022 8:49 AM

R1

main:  $\bigcirc$

start  $\square$

end:  $\bigcirc$



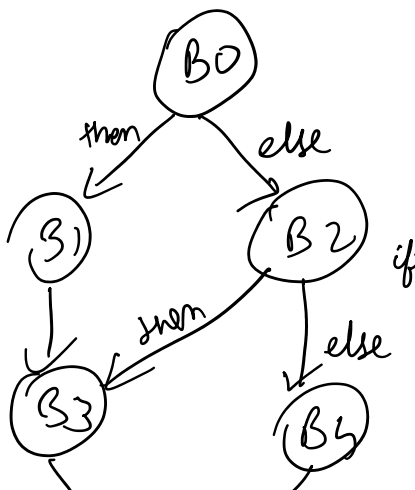
abdg  $w \ x \rightarrow \{w, x\}$

liveness for BB?  $B_0$

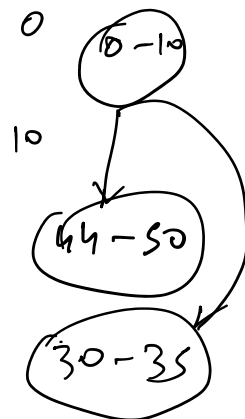
if ( $x > 0$ )  $\rightarrow \{x, w\}$

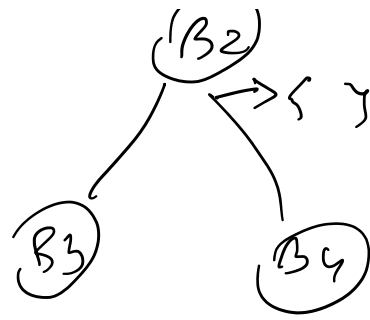
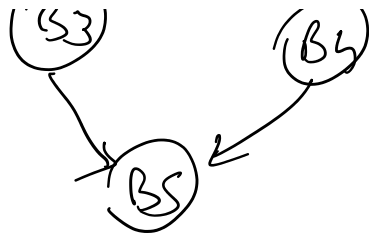
~~if ( $x > 1$ )  $e_2$   
 $x+w \rightarrow B_1$   
else~~

$x+1$   
else  $\rightarrow \{x, w\}$   
if ( $x < 0$ )  $e_3$   
 $x+w \rightarrow \{x, w\}$   
else  $\rightarrow B_2$   
 $x+1$



if jmp





$$\{B3 \mid \mathcal{L}(B3) \cup \mathcal{L}(B4)\}$$

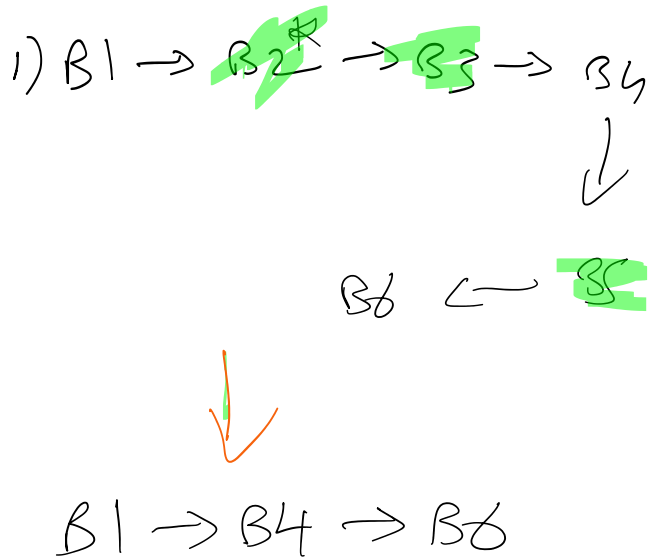
max  
max

$$\frac{\text{val} - \kappa}{\text{max} - \kappa}$$

```

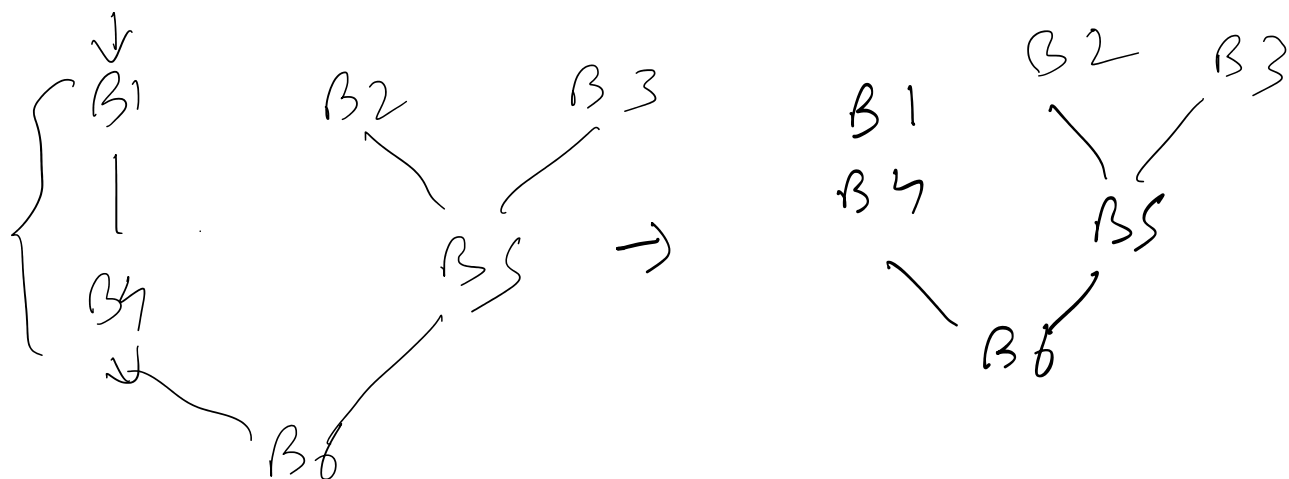
block8482:
  if (eq? x8473 2)
    goto block8479;
  else
    goto block8480; // block8476
block8481:
  if (eq? x8473 0)
    goto block8477;
  else
    goto block8478;
block8480:
  goto block8476;
block8479:
  goto block8475;
block8478:
  goto block8476;
block8477:
  goto block8475; }
block8476:
  return (+ y8474 10);
block8475:
  return (+ y8474 2);
start:
  x8473 = (read);
  y8474 = (read);
  if (< x8473 1)
    goto block8481;
  else
    goto block8482;

```



From <https://iucompilercourse.github.io/IU-P423-P523-E313-E513-Fall-2020/lecture-Sep-29.html>

② remove\_jump



2