

RBP

Print x86

start :

```

mov $10, rbp8(%rbp)
neg rbp8(%rbp)
mov rbp8(%rbp)
add jmp concl

```

main :

1 1 1 1 1 1

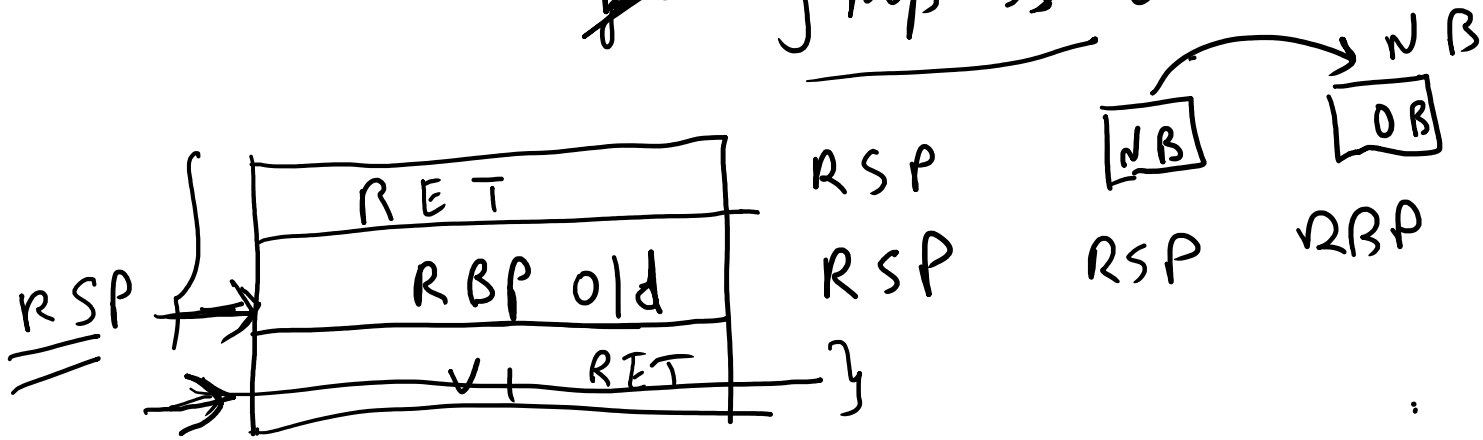
main :

hush %rbp ←

movq %rsp, %rbp

→ sub \$8, %rsp

~~jeq~~ jmp start



main

cond :

addq \$8, %rsp

pop %rbp

main :

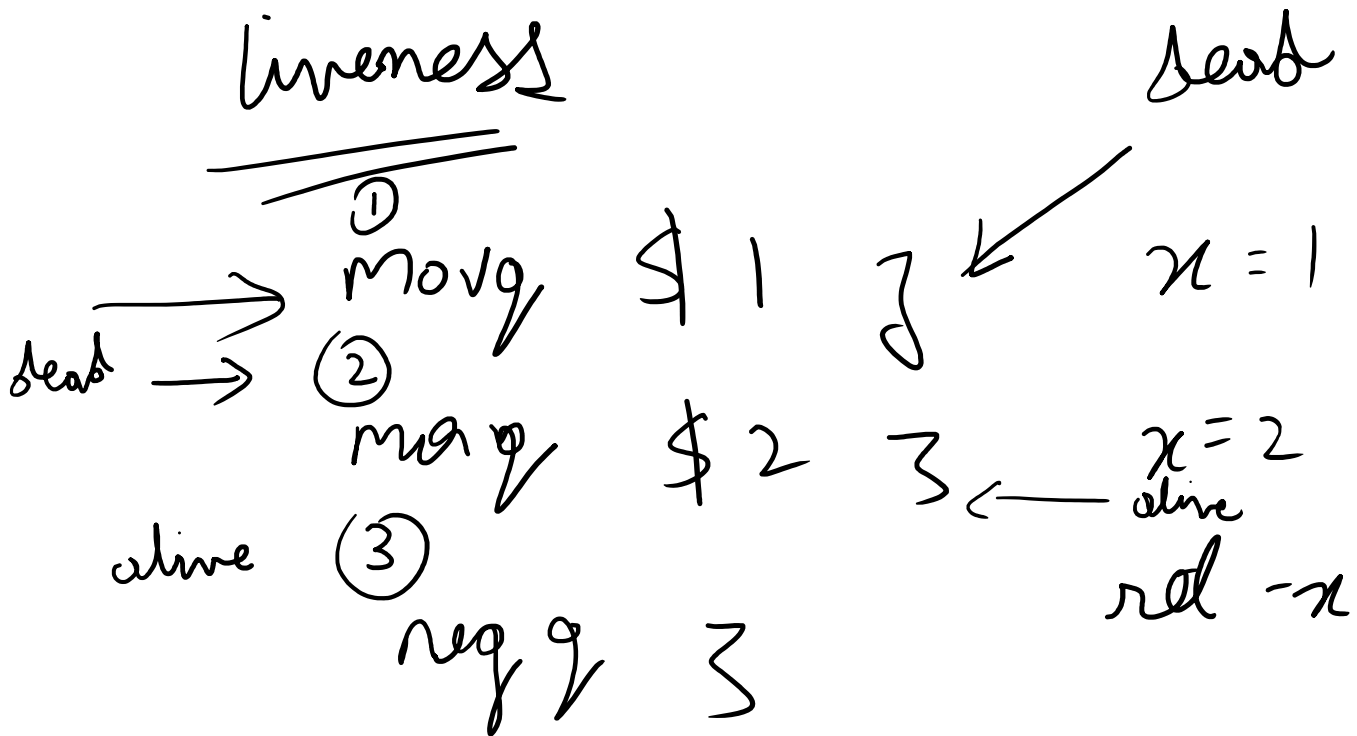


retq

arg: r di rsi rdx r cx
r8 r9

ret: rax

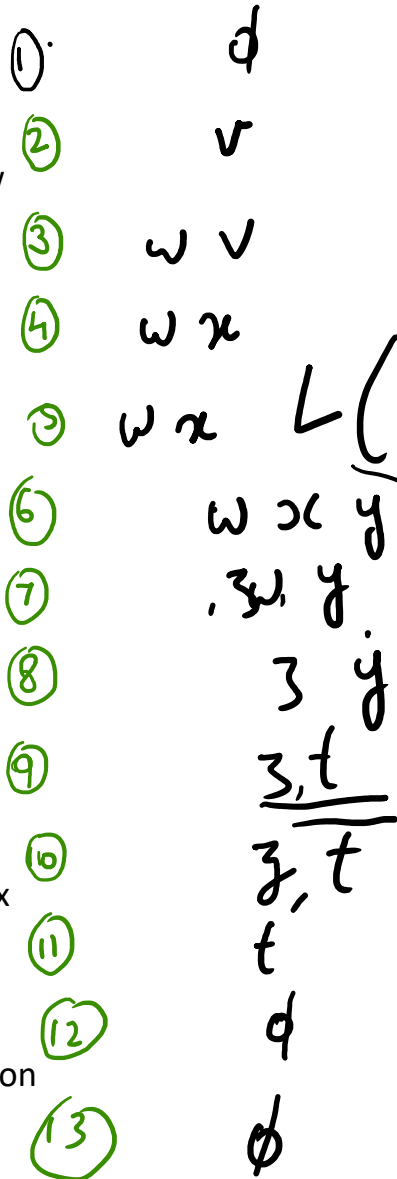
mark regs



```

start:
  movq $1, v
  movq $42, w
  movq v, x
  addq $7, x
  movq x, y
  movq x, z
  addq w, z
  movq y, t
  negq t
  movq z, %rax
  addq t, %rax
  jmp conclusion

```



$$L(k) = \{ L(k+1) - w \} \cup \{ R \}$$

$L(k+1) = z, t$
 $w = t$
 $r = y$
 $\{ z, y \}$

$w \rightarrow \text{remove}$
 $r \rightarrow \text{add}$

Interference graph

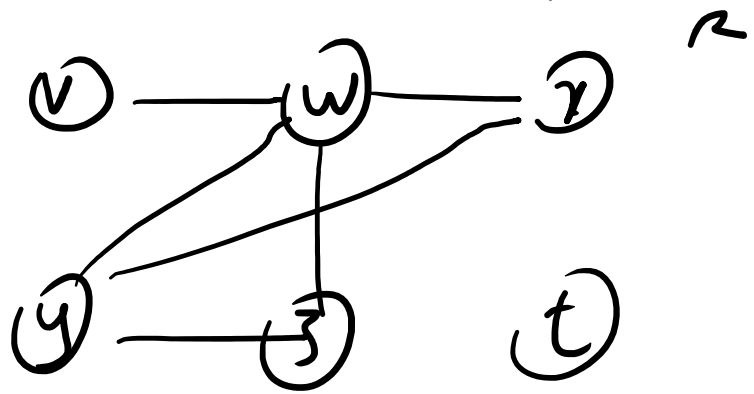
10 February 2022 04:01

```

1      {}
   movq $1, v
2      {v}
   movq $42, w
3      {v,w}
   movq v, x
4      {w,x}
   addq $7, x
      {w,x}
   movq x, y
      {w,x,y}
   movq x, z
      {w,y,z}
   addq w, z
      {y,z}
   movq y, t
      {t,z}
   negq t
      {t,z}
   movq z, %rax
      {t}
   addq t, %rax
      {}
   jmp conclusion
      {}
    
```

→ $v \leftrightarrow rdi$
 $w \leftrightarrow rdi$
 $v \leftrightarrow w$

$v_1 \leftrightarrow r1$
 $v_2 \leftrightarrow r2$
 $v_3 \leftrightarrow r1$
 ...



1) ~~addq s, d, v~~
 $v_1 = d$ $E(v, d) = 1$