

MAC 0329

08.07.2021



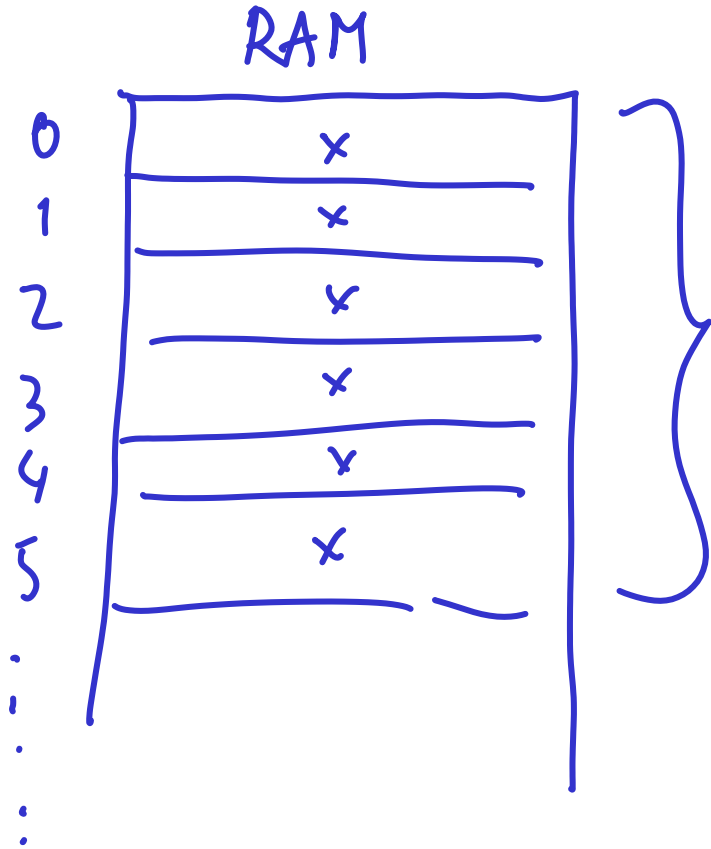
1. Programa em alto nível → ling. C.

↓  
compilação (compilador)      gcc

↓  
programa executável / binário  
(instruções de máquina)

# FDX

início  
do programa →



Programa  
binario

```

00: 0110 -- Copie [10] para o AC
01: 0211 -- Copie [AC] para a posição de endereço 11
02: 0712 -- Leia um número e guarde-o na posição de endereço 12
03: 0811 -- Imprima [12]
04: 0112 -- Copie [12] para o AC
05: 0D09 -- Desvie para 09 se [AC] < 0
06: 0311 -- Some [11] com [AC] e guarde o resultado em AC
07: 0211 -- Copie [AC] para a posição de endereço 11
08: 0A03 -- Desvie para 02
09: 0811 -- Imprima [11]
0A: 0900 -- Pare

```

```

... ..
0F: ...
10: 0000 -- Zero
11:      -- Soma
12:      -- Num

```

End Instr.

Soma = 0  
 Leia Num  
 Ler: Imprima Num  
 Se Num < 0 :  
     Soma = Soma + Num  
     Desvio p/ ler  
 Imprim...

```

AC <-- Zero
Soma <-- AC
Ler: Leia Num
Imprima Num
AC <-- Num
Se AC < 0, desvie para Fim
AC <-- Soma + AC
Soma <-- AC
Desvie para Ler
Fim: Imprima Soma
Pare

```

```

10 ~ Zero
11 ~ Soma
12 ~ Num

```

Soma = 0

Num ← leia

while num ≥ 0 :

|| Soma = soma + num

|| Num ← leia

fim → Imprima soma


```
int main() {
    int x=5;
    int y=7;

    return 0;
}
```

```
.file "prog1.c"
.text
.globl main
.type main, @function

main:
.LFB0:
.cfi_startproc
pushq %rbp
.cfi_def_cfa_offset 16
.cfi_offset 6, -16
movq %rsp, %rbp
.cfi_def_cfa_register 6
movl $5, -8(%rbp)
movl $7, -4(%rbp)
movl $0, %eax
popq %rbp
.cfi_def_cfa 7, 8
ret
.cfi_endproc

.LFE0:
.size main, .-main
.ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"
.section .note.GNU-stack,"",@progbits
```



```

int main() {
  int x=5;
  int y=7;

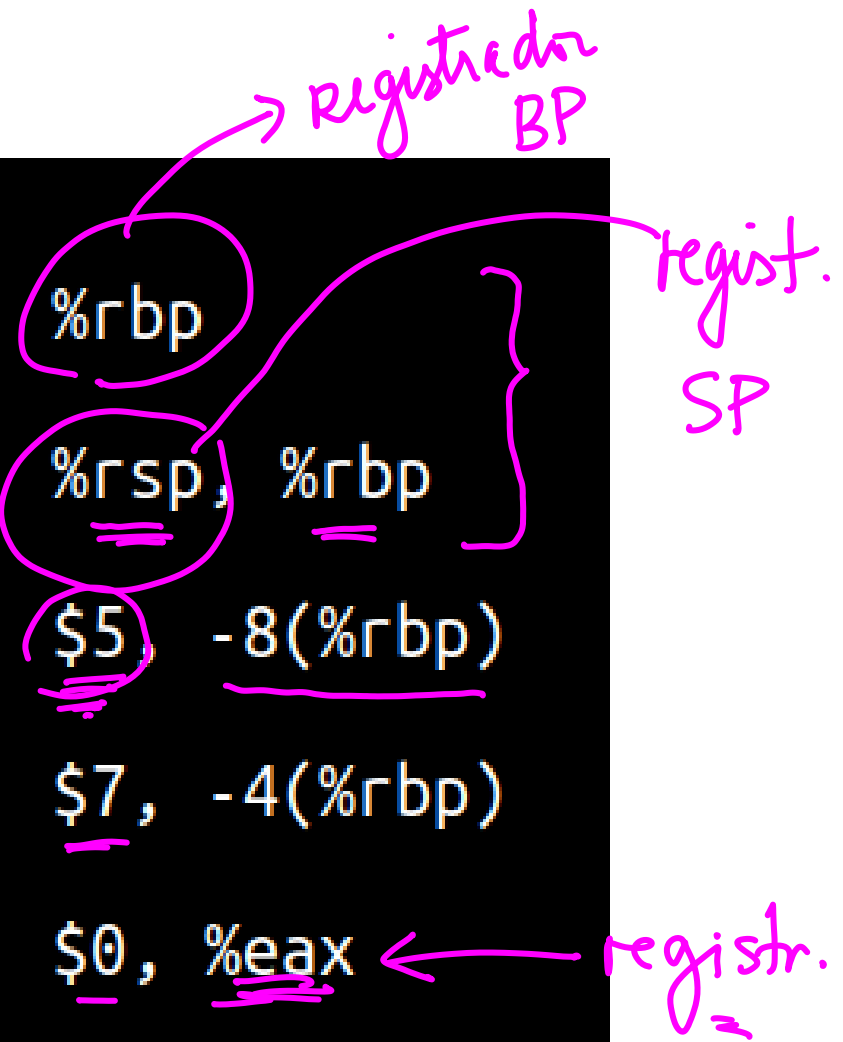
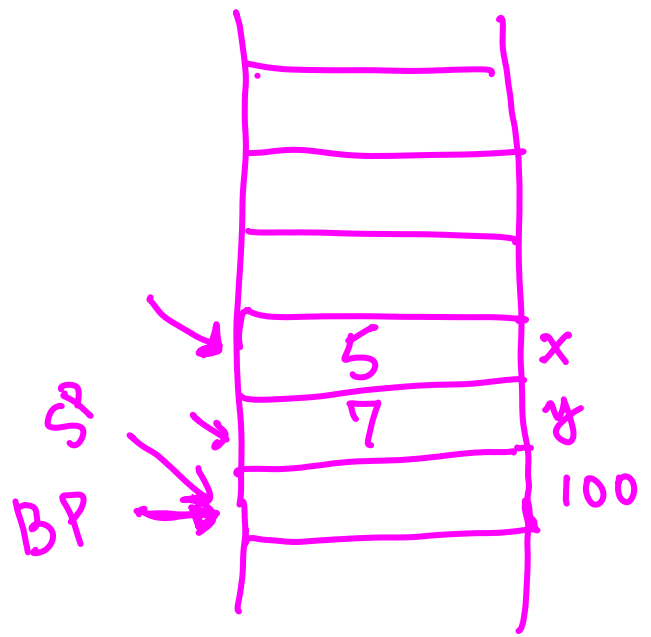
  return 0;
}

```

```

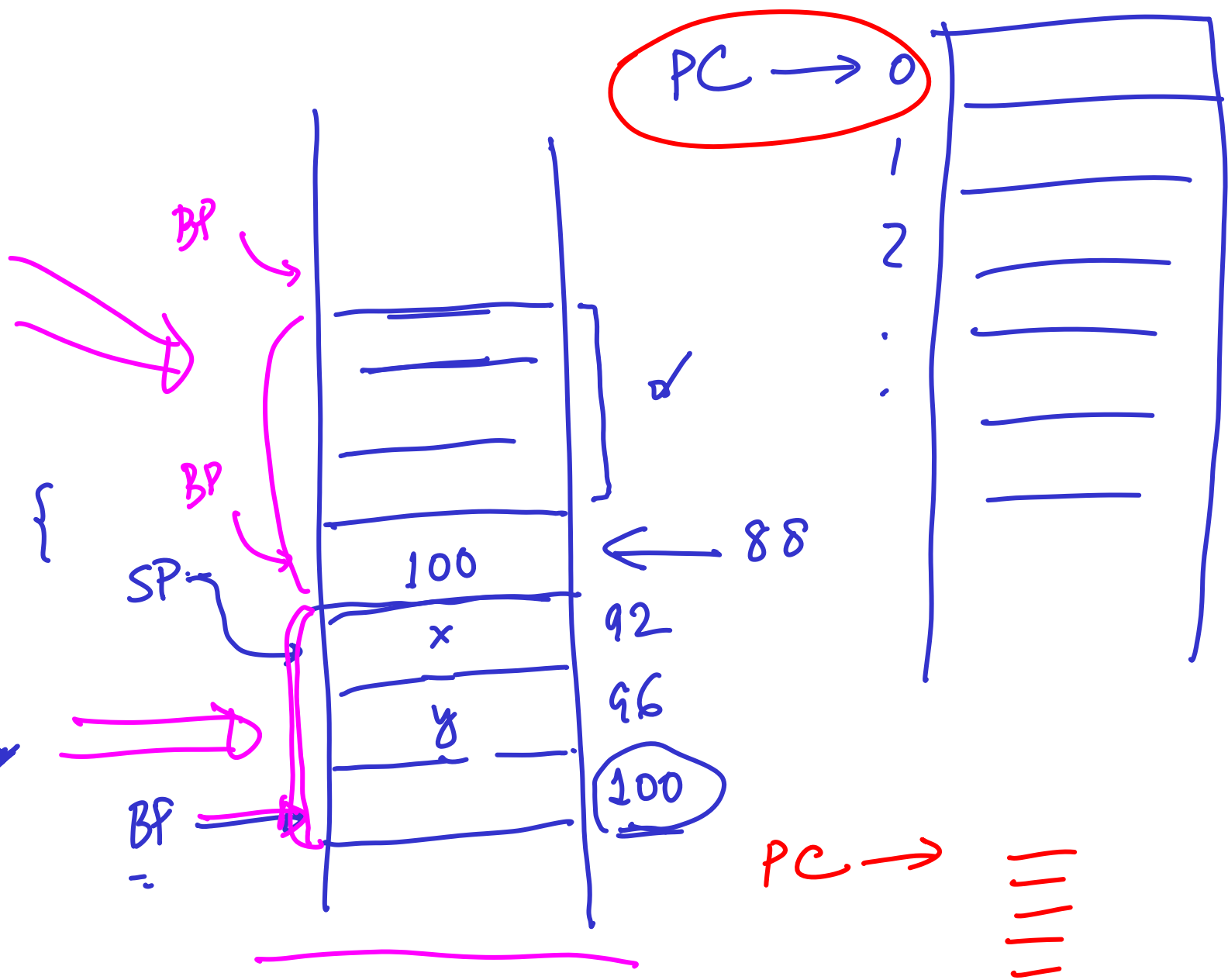
main:
  pushq   %rbp
  movq    %rsp, %rbp
  movl    $5, -8(%rbp)
  movl    $7, -4(%rbp)
  movl    $0, %eax
  popq   %rbp
  ret

```



```
int fat ( ) {  
    .  
    .  
    .  
}
```

```
int main ( ) {  
    .  
    .  
    .  
    fat ( )  
    .  
}
```



pilha de execução

PC →  
|  
|  
|  
|  
|



0, 1

E, OU, NÃO

E, NÃO

OU, NÃO

