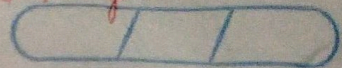


P1-LM

Nome: Victor Manuel Fernandes de Souza  
nº: 43579



### 1- // Plano para execução de instruções

9,7

PC = Endereço inicial // Contador de programa

Byte de Execução = true

While (Byte de execução)

R = memória[PC]

PC = PC + 1

Determine Tipo de Instrução (R, Tipo de Instrução)

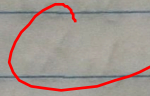
Características (Tipo de Instrução, Localização de dados, Dados requeridos)

If (Dados requeridos)

dados = memória [Localização]

Execute (Tipo de Instrução, dados, AC, PC, Bit de Instrução)

2- 0705<sub>16</sub>

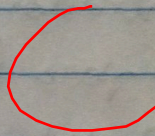


3- a) EF

b) CDEF

c) 89ABCDEF

d) 0123456789ABCDEF

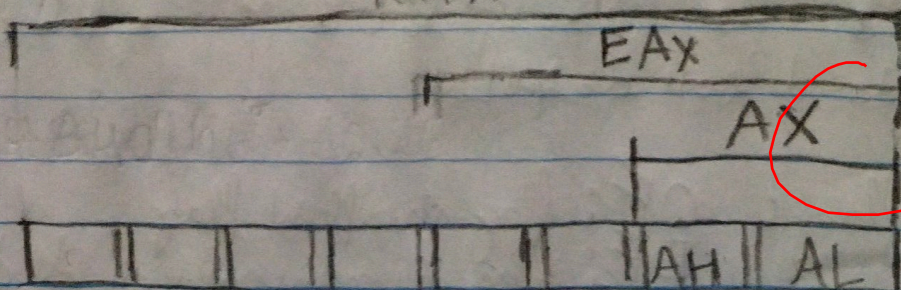


4- No máximo, cada registrador alcança de 1 byte ou 8 bits ou 8 bytes ou 64 bits por endereço. Dependendo do tipo de variável ou registro.

0 1 2 3 4 5 6 7 8 9 A B C D E F

5- a) 0 b) 0

RAX



5-

6- a) 8

b) 64

c) 16

d) 32

8- a) MOV R, Var carrega o endereço de Var em R e o conteúdo de Var no caso de MOV R [Var]

LEA → Carrega endereço efetivo + deslocamento e o deslocamento for 0, não é necessário adicionar. Assim, LEA R, [Var] faz o mesmo que MOV R, Var. Contudo, a instrução MOV R, Var + deslocamento não existe. Para isso toma-se que usar LEA R, [Var + deslocamento]

b) section .text: seção de código onde o programa é feito

section .data: seção onde se declaram as variáveis inicializadas

section .bss: seção de variáveis não inicializadas

c) RAX: registrador conhecido como acumulador, geralmente os resultados vão para RAX.

RBX: registrador que pode ser usado índice de base

RCX: registrador geralmente usado para contagem

RDX: registrador que serve de auxiliar em algumas operações de multiplicação/divisão

RIP: registrador contador de programa, guarda o endereço da linha de instrução

RSI: registrador de índice e origem

RDI: registrador de índice e destino

RSP: registrador que aponta para o topo da pilha de memória

RBP: registrador de base de pilha

registradores de flag: OF: flag overflow flag

PF: flag de manipulação de dados

IF: flag de manipulação de dados  
 TF: flag de busca (trap)  
 ZF: flag para resultado zero  
 PF: flag de resultado par  
 CF: flag de carry  
 RIP

9. a)  $007B_{16}$   
 ~~$0007_{16}$~~   $173$   
 $0000\ 0000\ 0111\ 1011_2$

b)  $00BF_{16}$   
 $191_{10}$   
 $0000\ 0000\ 1011\ 1111_2$

c)  $005F_{16}$   
 $95_{10}$   
 $137_8$

d)  $-20563_{10}$   
 $127655_8$   
 $1010\ 1111\ 1010\ 1101_2$

10. a)  $0000\ 0000\ 0000\ 1101\ 1110\ 0000_2$

b)  $0000\ 0000\ 0001\ 0000\ 0001\ 0001_2$

c)  $0000\ 0000\ 0000\ 1001\ 1111\ 0100_2$

d)  $0000\ 0000\ 0000\ 0001\ 0011\ 1111_2$