

Romcu 43577

24/06

PL de LM

9,7

Q01) Descreva um algoritmo que simule o modo como ~~o computador~~ execute programas e instruções.

// Passos para a execução de instruções

PC = Endereço inicial // contador de programa (RIP)

Byte de execução = True

while (Byte de Execução):

R = Memória [PC]

PC = PC + 1

Determine Tipo Instrução (IR, Tipo de Instrução)

Arco Dado (Tipo de Instrução, IR, Localização Dado, Dado Requerido)

if (Dado Requerido):

dado = Memória [Localização Dado]

Execute (Tipo de Instrução, dado, memória, AC, PC, Bit de Instrução)

Q02) 0705₁₆

Q03) a) EF

b) CDEF

c) 89ABCDEF

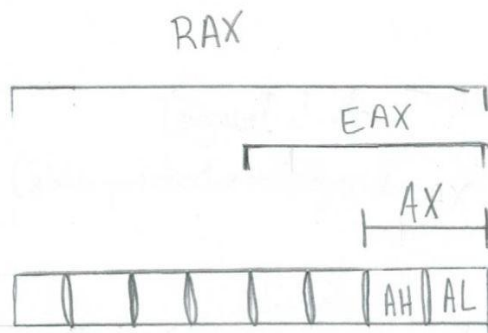
d) 0123456789ABCDEF

Q04) No máximo, cada registrador alcança de 1 byte ou 8 Bits a 8 bytes ou 64 bits por endereço. Depende do tipo de variável ou registrador.

Q05) a) 0

b) 0

Q05)



Q06) a) 8

b) 64

c) 16

d) 32

Q08) a) $MOV R, Var$ carrega o endereço de Var em R ou o conteúdo de Var no caso de $MOV R, [VAR]$

LEA - Carrega o endereço efetivo + deslocamento para R , não é necessário adicionar. Assim, $LEA R, [VAR]$ faz o mesmo que $MOV R, Var$. Contudo, a instrução $MOV R, VAR + deslocamento$ não é aceita. Para isso temos 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 de linha de instrução

RSI : Registrador de índice e origem

RDI : Registrador de índice e destino

RSP Registrador que aponta para o topo do pilho de memória

RBP : Registrador de base do pilho

OF : Flag overflow flag

DF : Flag de manipulação de dados

IF : Flag de interrupção

TF : Flag de busca (Trap)

ZF : Flag para resultado zero

PF : Flag de resultado par

CF : Flag de carry um

d) RIP

$$Q9) a) = 0000 \ 0000 \ 0111 \ 1011_2 = 000753_8 = 007B_{16}$$

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

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

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

0 C 7

$$Q10) a) 110 \ 111 \ 100 \ 000$$

$$b) 001 \ 000 \ 000 \ 010 \ 001$$

$$c) 100 \ 111 \ 100 \ 000$$

$$d) 100 \ 111 \ 111$$

C