

Correcção do teste-ensaio de SC (08/05/33)

1)

a) "a" = 341₈

"A" = 101₈

"JaDe" = ?

"J" = 101₈ + 3(30) = 101₈ + 011₈ = 112₈

"D" = 101₈ + 3(30) = 101₈ + 03₈ = 104₈

"e" = 341₈ + 4(30) = 341₈ + 04₈ = 345₈

Base 8: [0..7]

$$\begin{array}{r} 3 \overline{) 341} \\ \underline{30} \\ 41 \\ \underline{40} \\ 1 \end{array}$$

$$\begin{array}{r} 101 \\ + 011 \\ \hline 112 \end{array}$$

$$\begin{array}{r} 3 \overline{) 101} \\ \underline{30} \\ 71 \\ \underline{60} \\ 11 \end{array}$$

$$\begin{array}{r} 101 \\ + 03 \\ \hline 104 \end{array}$$

$$\begin{array}{r} 4 \overline{) 341} \\ \underline{40} \\ 11 \\ \underline{10} \\ 1 \end{array}$$

$$\begin{array}{r} 341 \\ + 04 \\ \hline 345 \end{array}$$

"JaDe" = 112 141 104 345 (8)

Base 16: [0..9, A..F]

b) 9 = 1001₂ = 0x9

"2013" = ?

2(30) = 0010(2) = 2(16)

0(30) = 0000(2) = 0(16)

1(30) = 0001(2) = 1(16)

3(30) = 0011(2) = 3(16)

$$\begin{array}{r} 2 \overline{) 2013} \\ \underline{20} \\ 13 \\ \underline{12} \\ 1 \end{array}$$

$$\begin{array}{r} 0 \overline{) 2013} \\ \underline{00} \\ 2013 \end{array}$$

$$\begin{array}{r} 1 \overline{) 2013} \\ \underline{10} \\ 1013 \\ \underline{1000} \\ 13 \end{array}$$

$$\begin{array}{r} 3 \overline{) 2013} \\ \underline{30} \\ 13 \\ \underline{12} \\ 1 \end{array}$$

2(30) = 010(2) = 0010 / 0(30) = 00 = 0000 / 1(30) = 01(2) = 0001 / 3(30) = 011(2) = 0011
(porque 2⁴ = 16, logo agrupa-se em 4 bits, acrescentando zeros à esquerda)

$$\begin{array}{ccccccc} 0010 & 0000 & 0001 & 0011 & = & 2013 & (16) \\ \downarrow & \downarrow & \downarrow & \downarrow & & & \\ 2^1 = 2 & 0 & 2^0 = 1 & 2^0 + 2^1 = 3 & & & \end{array}$$

Logo, 2013(30) = 2013(16)

2)

a) ...

movl 8(%ebp), %eax → %eax ← int arg (1º arg) (eax toma o valor do 1º argumento)

cmpl \$1, %eax → %eax = 1 (case 1)

movl 12(%ebp), %edx → %edx ← char x = (2º arg) (edx toma o valor do 2º argumento)

je .L6 → se case 1, salta para o fim

.L1: senão

movl %ebp, %esp → coloca o %esp a apontar para o mesmo sítio que o frame pointer %ebp (atualiza o stack pointer)

popl %ebp → recupera o anterior valor do frame pointer

ret Fim do programa

.L6:

movsbl %edx, %eax → %eax ← char x e estende o resto de %eax com o bit do sinal

incl %eax → %eax = x + 1

jmp .L1

b) ...

movl 8(%ebp), %edx → %edx ← short arg (1º arg, colocado no stack com 32 bits)

testw %dx, %dx → testa se arg > 0

movl 12(%ebp), %ecx → %ecx ← int x (2º arg) (ecx toma o valor do 2º argumento)

jle .L8 → se arg ≤ 0, salta para o fim

.L6: senão

movswl %dx, %eax → %eax ← short arg e estende o resto de %eax com o bit do sinal

decl %edx → %edx = arg - 1

subl %ecx, %eax → faz a operação x = x - arg

testw %dx, %dx → testa se arg > 0

jg .L6 → se arg > 0, salta para o início

.L8: senão

movl %ecx, %eax → %eax ← x, o valor a devolver pela função

movl %ebp, %esp → coloca o %esp a apontar para o mesmo sítio que o frame pointer

popl %ebp → recupera o anterior valor do frame pointer

ret Fim do programa

a) $\%ax = 0x94$
 $\%cx = 0xc$

addw $\%cx, \%ax \rightarrow$ soma o registro $\%ax$ com o registro $\%cx$

addn $y \cdot cx$, $y \cdot ax = ?$

$$C(13e) = 12(10)$$

65

$C = 12$; $4 + 12 = 16$; $16 = 0$ e "solva" 1

0	1	2	3	4	5	6	7	8	9	Deci.
0	1	2	3	4	5	6	7	8	9	Hexa.
J0	J1	J2	J3	J4	J5					
A	B	C	D	E	F					

b) $[-9000, +40000]$
36 bits, c/2 36 bits em complemento para 2 representa uma gama de valores entre $[-2^{15}, 2^{15}]$, isto é, entre $[-32\text{ K}, 32\text{ K}]$.

16 bits, c/2

$[-32\text{ K}, 32\text{ K}]$

Logo, o banco de registros deste processador não está preparado para conter valores escalares inteiros pertencentes ao intervalo $[-9000, +40000]$.

a) 12 bits $-26375 \times 10^3 = -26.375$

$$= -11610.0112$$

$$(=) V = -1.101001 \times 2^4 \quad \text{or} \quad V = -1.101010 \times 2^4$$

mantissa = 6 bits

Aproximando para o ímpar mais próximo: 101003

Aparelho para o povo mais próximo! Jo Jo Jo

$$0,375 = 0,375 \times 2 + 0,75 \times 2 + 0,5 \times 2 = 011$$

(agendas parte inteira)

$$\text{Exp} = 20, 20 - 16 = 4$$

$$b) \frac{1100 \ 111 \ 1011}{s = 678 \quad \text{Nanissa}} / 10011 = 2^4 + 2^1 + 2^0 = 19 / 2^{19-16} = 2^3$$

$$V = -1.111011_2 * 2^3 = -1111.011_2 = 2^3 + 2^2 + 2^1 + 2^0 + 2^{-1} + 2^{-2} = -15.375_{(10)}$$