

Ficha de Avaliação 1

Semana de 11-Nov-02

Alberto José Proença

Metodologia

A ficha deverá ser lida e individualmente respondida no prazo de 1h, nas folhas fornecidas, durante a 1ª parte da sessão teórico-prática. Na 2ª parte proceder-se-á à sua correcção colectiva. Apresente os cálculos que efectuar para chegar aos resultados, no verso da folha.

1. Sistemas de numeração e representação de inteiros

1.1. Efectue as seguintes conversões

a) De 10101.01_2 para hexadecimal 15.4₁₆

e para decimal 21.25₁₀

b) De -251 para uma representação binária com 10-bits, em complemento para 2

11 0000 0101₂

c) De $10\ 0100\ 0101_2$ (representado em excesso 2^{n-1} , com $n = 10$), para decimal

69₁₀

1.2. Considere que está a executar código num computador de **8-bits**, o qual usa complemento para 2 para representar valores do tipo inteiro. Um inteiro "short" é codificado usando 4-bits. Complete a tabela, considerando as seguintes definições:

```
short sy = -5;
int y = sy;
int x = -25;
unsigned ux = x;
```

Expressão	Decimal	Binário
x	-25	1110 0111 ₂
ux	231	1110 0111 ₂
y	-5	1111 1011 ₂
x>>1	-13	1111 0011 ₂

2. Representação de valores em vírgula flutuante

Considere o seguinte formato de vírgula flutuante, representado com 10-bits, baseado na norma IEEE:

▪ formato PEQUENO:

- o bit mais significativo contém o bit do sinal
- os 4 bits seguintes formam o expoente (em excesso de 15)
- os últimos 5 bits representam a mantissa

Para todos os restantes casos, as regras são as mesmas que as da norma IEEE (valor normalizado, desnormalizado, representação do 0, infinito, e NaN).

2.1. Represente, em decimal,

o valor negativo desnormalizado mais próximo de zero

$-2^{-5} * 2^{-14}$

2.2. Converta 1 0110 01100, formato PEQUENO, para decimal:

$-1.375 * 2^{-9}$

Nº

Nome:

Turma: Seg1 - Seg2 - Ter

3. Acesso a operandos e operações aritméticas

Considere que os seguintes valores estão armazenados em registos e em endereços de memória:

Endereço	Valor
0x100	0xFF
0x104	0xAB
0x108	0x13
0x10C	0x11

Registo	Valor
%eax	0x100
%ecx	0x5
%edx	0x3

Preencha a seguinte tabela mostrando os efeitos das instruções seguintes, quer em termos de localização dos resultados (em registo ou endereço de memória), quer dos respectivos valores:

Instrução	Destino	Valor
subl %ecx, 8(%eax)	0x108	0xE
imull \$4, (%eax, %edx, 4)	0x10C	0x44
incl 4(%eax)	0x104	0xAC
leal 256(%eax, %eax, 4), %edx	%edx	0x600

4. Estruturas de controlo

A partir de um programa em C, que invoca uma função simples com uma estrutura de controlo (while), obteve-se um fragmento de programa em *assembly*, aqui representado, referente ao corpo dessa função.

4.1. Substitua os valores “xx” (em 2 lugares) pelos correspondentes valores numéricos.

4.2. Sabendo que o único argumento passado para esta função se encontra em Mem[(%ebp)+8], introduza comentários no corpo da função, em *assembly*.

```

8048397: 8b 4d 08      mov    0x8(%ebp), %ecx /*Coloque o parâm. n em %ecx */
804839a: b8 01 00 00 00  mov    $0x1, %eax /*Inicialize a 1 a var de retorno */
804839f: 39 c8        cmp    %ecx, %eax /*Compare 1 : n */
80483a1: ba 01 00 00 00  mov    $0x1, %edx /* Inicialize a 1 a var de contagem */
80483a6: 7f 08        jg     xx -> 80483b0 /*Termine se n < var_cont (=1) */
80483a8: 0f af c2     imul  %edx, %eax /*Ciclo: var_ret=var_ret * var_cont */
80483ab: 42          inc    %edx /*var_cont ++ */
80483ac: 39 ca        cmp    %ecx, %edx /*Compare var_cont : n */
80483ae: 7e xx -> f8  jle   80483a8 /*Se var_cont <= n, repetir ciclo */

      Fim do corpo da função
80483b0: c9          leave
80483b1: c3          ret

```

4.3. Recupere o ficheiro original em C, contendo esta função aqui descrita em *assembly* (reverse engineering).

```

int fact (int n)
{
    int var_ret = 1, var_cont = 1 ;
    while (var_cont <= n)
    {
        var_ret = var_ret * var_cont ;
        var_cont ++ ;
    }
    return var_ret ;
}

```

Nº

Nome:

Turma: Seg1 - Seg2 - Ter