

# Arquitectura de Computadores 2009/10

## Aula prática 8

1. Considere que uma string representa um número em binário.

Por exemplo "1011" corresponde ao número decimal 11.

- a. Programe uma subrotina em assembler, seguindo as convenções de passagem de parâmetros do C, que verifica se a string corresponde a um número em binário válido, i.e. se apenas contém os caracteres '0' e '1'.
- b. Teste a subrotina com um programa em C.
- c. Programe uma subrotina que retorna o valor do número representado pela string.
- d. Teste com um programa em C.

2. Considere a subrotina que efectua a troca de duas variáveis inteiras:

```
troca (int *x, int *y);
```

- a. Implemente a subrotina em assembler, seguindo as convenções da passagem de parâmetros do C.
- b. Teste a subrotina com um programa em C.
- c. Faça um programa completo em assembler que efectua a chamada à subrotina (troca) desenvolvida na alínea a. Teste o seu programa utilizando o debugger ddd.

3. Considere a série de Fibonacci:

$$\begin{aligned}f(0) &= 0 \\f(1) &= 1 \\f(n) &= f(n-1) + f(n-2)\end{aligned}$$

Que pode ser implementada iterativamente através do seguinte código:

```

int fibonacci( int n )
{
    int i;
    int fi;           // fib(i)
    int fi1;          // fib(i-1)
    int fi2;          // fib(i-2)

    if( n == 0 )
        fi = 0;
    else
    {
        fi1 = 0;       // f(0)
        fi = 1;        // f(1)
        for( i = 2; i <= n; i++ )
        {
            fi2 = fi1;
            fi1 = fi;
            fi = fi2 + fi1;
        }
    }
    return fi;
}

```

- a. Implemente em assembler, a função de Fibonacci de forma iterativa, utilizando as convenções de passagem de parâmetros do C
- b. Teste a subrotina com um programa em C.
- c. Faça um programa completo em assembler que efectua a chamada à função desenvolvida na alínea a. Teste o seu programa utilizando o debugger ddd.

## Números reais

1. Considere seguinte função, que calcula a área de um círculo:

```
float areacirculo (float raio);
```

( Cujas função é dada por:  $area = \pi \cdot r^2$  )

- a. Programe a função como uma subrotina em assembler.
- b. Desenvolva um programa em C para testar a subrotina desenvolvida na alínea anterior.
- c. Faça um programa completo em assembler que efectua a chamada à subrotina (areacirculo) desenvolvida na alínea a. Teste o seu programa utilizando o debugger ddd.

2. Considere a função potência:

```
float potencia (float base, int expoente);
```

Que calcula o valor de  $base^{\text{expoente}}$

- a. Programe a função como uma subrotina em assembler, assumindo que o expoente é positivo.
- b. Altere a subrotina da alínea anterior de forma a poder efectuar os cálculos com valores de expoente negativos.
- c. Desenvolva um programa em C para testar as subrotinas desenvolvidas nas alíneas anteriores.