

# Sistemas Embebidos I

Ano Letivo 2014/2015 - Semestre de Inverno

## Lab2 – Introdução à programação em *assembly*

1. Considere a seguinte declaração da função `Div`, que realiza a divisão inteira do valor `numer` pelo valor `denom`, retornando o quociente desta operação.

```
unsigned int Div(unsigned int numer, unsigned int denom);
```

- a) Escreva o código da função `Div` em *assembly* da arquitetura ARM, respeitando a norma APCS.
- b) Escreva, em *assembly* da arquitetura ARM, um programa que teste a função implementada.
- c) Admitindo que a função `Div` deve retornar o quociente e o resto da divisão, escreva uma implementação alternativa desta função em *assembly* da arquitetura ARM.

2. Considere a seguinte declaração da função `Log`, que devolve no parâmetro `result` a parte inteira do logaritmo do parâmetro `value`, na base indicada pelo parâmetro `base`. A função retorna o valor `-1` em caso de erro, ou `0` em caso contrário.

```
int Log(unsigned int *result, unsigned int value, unsigned char base);
```

- a) Implemente a função `Log` em *assembly* da arquitetura ARM, respeitando a norma APCS.
- b) Realize, em linguagem C, um programa que permita testar a função `Log`.

3. Considere a seguinte definição:

```
typedef struct {  
    unsigned int value;  
    unsigned char base;  
} ValueDsc;
```

a) Escreva, em *assembly* da arquitetura ARM e respeitando a norma APCS, o código da função `LogArray` que calcula a parte inteira do logaritmo de cada um dos `numValues` elementos do `array values` e armazena-o, com o mesmo índice, no `array result`. O logaritmo é calculado pela função passada no parâmetro `funct`. A função retorna o número de elementos para os quais foi possível calcular o logaritmo, ou `-1` em caso de erro.

```
int LogArray(unsigned int *result[], ValueDsc *values[],  
             unsigned char numValues,  
             int (*funct)(unsigned int *, unsigned int, unsigned char));
```

- b) Realize, em linguagem C, um programa que permita testar a função `LogArray`, usando a função `Log` desenvolvida no exercício 2.