

**PUC-Rio – Software Básico – INF1018**  
**Prova 1 – Turma 3WA – 16/10/2018**

1. (2,0 pontos) Considere uma mensagem codificada num *short* da seguinte forma:

Bits:	15	14-12	11-8	7-4	3-0
	oper	lixo	val2	val1	val0

Onde *val0*, *val1* e *val2* correspondem a três números inteiros sem sinal de 4 bits cada.

Escreva em C uma função que dado uma mensagem do tipo *short* no formato acima, retorne a soma (*val0* + *val1* + *val2*) se o bit *oper* for igual a zero, ou retorne o produto (*val0* x *val1* x *val2*) se o bit *oper* for igual a um. Use o protótipo seguinte para sua função:

```
unsigned int decod_short(short msg);
```

2. (3,0 pontos) Considere o programa C a seguir:

```
#include <stdio.h>
void dump (void *p, int n) {
    unsigned char *p1 = (unsigned char *) p;
    while (n--) {
        printf("%p - %02x\n", p1, *p1);
        p1++;
    }
}

struct prof {
    short nivel;
    int grau;
    char *cursos;
    char ano;
};

int main(void) {
    struct prof univ[2];
    char nomes[] = {'c', 'e'};
    univ[0].nivel = 12 << 3;
    univ[0].grau = -3;
    univ[0].cursos = &nomes[1];
    univ[0].ano = 'd';
    dump(univ, sizeof(struct prof));
    return 0;
}
```

Descreva o que este programa irá imprimir quando executado, *justificando* os valores exibidos mostrando as contas e outras informações usadas para chegar ao resultado. Suponha que a máquina de execução seja *little-endian* com as convenções de alinhamento do Linux no IA-64 vistas em sala. Além disto considere que *univ* seja alocado na posição de memória 0x7ffc396a6e0, que o valor do caracter 'a' na tabela ASCII seja 97 (decimal) e que *nomes* seja alocado na posição de memória 0x7ffc396a711. Coloque **PP** nas posições correspondentes a *padding*.

3. Traduza as funções `foo` e `vai` abaixo para assembly IA-64 (o assembly visto em sala), utilizando as regras usuais de alinhamento, passagem de parâmetros, salvamento de registradores e resultados em C/linux. Traduza o mais diretamente possível o código de C para assembly.

(Não se preocupe em entender o que as funções fazem, apenas traduza-as literalmente.)

**Comente seu código!**

- (a) (2,5 pontos)

```
int foo (int n, int bias)
{
    if (n==0)
        return 1;
    else
        return n * foo(n-1, bias) + bias;
}
```

- (b) (2,5 pontos)

```
int ebom(int val);

struct Fam
{
    int eva;
    int del;
};

void vai(int n, struct Fam v[])
{
    int i, r;

    for (i=0; i<n; i++)
    {
        r = ebom(v[i].eva);
        if (r > 0)
            v[i].del = i;
        else
            v[i].del = 0;
    }
}
```

Boa Prova!