

**PUC-Rio – Software Básico – INF1018**  
**Prova 1 – Turma 3wb – 10/05/2018**

1. (2,5 pontos) Suponha as declarações abaixo, usadas em nosso primeiro trabalho:

```
#define NUM_BYTES 16
typedef unsigned char BigInt[NUM_BYTES];
```

O tipo `BigInt` representa um inteiro de 128 bits, em complemento a dois, armazenado em *little-endian*.

Escreva, em C, uma função chamada *big\_min* que retorna o **menor** dentre dois `BigInt`. Essa função deve ter o seguinte protótipo:

```
void big_min(BigInt res, BigInt a, BigInt b);
```

2. (2,5 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++;
    }
}

char strings[2][5] = {"uma", "duas"};
struct S {
    char c;
    int i;
    char *p;
};
struct S s = {(33 << 2) & 0x55, -4100, strings[1]};
int main (void) {
    dump (&s, sizeof(s));
    return 0;
}
```

Sabendo das informações abaixo e supondo que a máquina de execução é *little-endian* com as convenções de alinhamento do Linux no IA-64 (vistas em sala), **mostre o que esse programa irá imprimir quando executado**. Coloque **PP** nas posições correspondentes a *padding*.

endereço de <code>strings</code> na memória	0x55c31720b010
endereço de <code>s</code> na memória	0x55c31720b020

(Mostre como você chegou aos valores exibidos. Valores sem contas **NÃO** valem ponto!).

3. Traduza as funções `foo` e `boo` abaixo para assembly IA-64, utilizando as regras usuais de alinhamento, passagem de parâmetros, salvamento de registradores e resultados em C/Linux. (Não se preocupe em entender o que as funções fazem, apenas traduza-as literalmente.)

Atenção! **Traduza o mais diretamente possível o código de C para assembly.**

(a) (2,5 pontos)

```
struct X {
    int val1;
    int val2;
    int val3;
};

int foo(struct X *px, int n, int v) {
    int nao = n;
    while (n--) {
        if (px->val1 & v) {
            px->val2 = px->val3 * 2;
            nao--;
        }
        px++;
    }
    return nao;
}
```

(b) (2,5 pontos)

```
int ix(int x);

void boo(int va[], int n) {
    int i, j;
    for (i = 0; i < n; i++) {
        j = ix(i);
        if ((j >= 0) && (j < n)) {
            va[j] = i;
        }
    }
}
```

Boa Prova!