

PUC-Rio – Software Básico – INF1018
Prova 1 – Turma 3wb – 14/05/2019

1. (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-- > 0) {
        printf("%p - %02x\n", p1, *p1);
        p1++;
    }
}
struct S1 {
    short val_short;
    int   val_int;
    char *val_string;
};
struct S2 {
    void *p;
    int   i;
    char  c;
};
char minha_string[] = "turma 3wb";
struct S1 s1 = {12, 34, minha_string};
struct S2 s2[2] = {{&s1.val_int, -129, 'E' << 4},
                  {&s1.val_string, 1026, 0x55 & 0x0f}};
int main (void) {
    dump (s2, sizeof(s2));
    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), diga o que esse programa irá imprimir quando executado. Coloque **PP** nas posições correspondentes a *padding*.

endereço de <code>minha_string</code> na memória	0x55875cfee010
endereço de <code>s1</code> na memória	0x55875cfee020
endereço de <code>s2</code> na memória	0x55875cfee040
valor do caracter 'A' na tabela ASCII	65 (decimal)

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

2. Traduza as funções `foo` e `boo` a seguir 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.**

