

PUC-Rio – Software Básico – INF1018
Prova Final – 24/6/2014

1. (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++;
    }
}
struct X {
    int i1;
    short s1;
    float f1;
    char c1;
} x = {-2050, 40, -3.75, 'e'};
int main(void) {
    dump((void *)&x, sizeof(x));
    return 0;
}
```

Supondo que x seja armazenado no endereço de memória 0x80500c2, e sabendo que o código ascii de 'a' é 97 (na base 10), diga o que o programa irá imprimir quando executado, deixando claro como você chegou a esses valores. Considere que a máquina de execução é *little-endian*, e que as convenções de alinhamento são as do Linux no IA-32. Se houver posições de *padding*, indique seu conteúdo com **pp**. (ATENÇÃO: valores sem contas e explicações **NÃO** valem ponto!)

2. (2,5 pontos) Escreva em C uma função `pack` que recebe como argumentos um array de 4 bytes e retorna um `unsigned int` composto por estes 4 bytes, com o primeiro elemento do array ocupando o byte menos significativo do `unsigned int`, e assim por diante.

```
unsigned int pack (char bytes[]);
```

3. Traduza as funções `foo` e `bar` abaixo para assembly IA-32 (o assembly visto em sala), utilizando as regras usuais de alinhamento, passagem de parâmetros e resultados de C/Linux/IA-32. Traduza o mais diretamente possível o código de C para assembly. Comente seu código.

- (a) (2,5 pontos)

```
struct no {
    int val;
    struct no *esq;
    struct no *dir;
};
int bar (struct no *n, int val) {
    if (!n)
        return 0;
    if (n->val==val) return 1;
    if (bar(n->esq,val))
        return 1;
    return bar(n->dir, val);
}
```

(b) (2,5 pontos)

```
double sin(double x);
double foo (double a[], int n) {
    double res = 0.0;
    while (n--) {
        res += sin(*a);
        a++;
    }
    return res;
}
```