

PUC-Rio – Software Básico – INF1612
Prova 1 – 9/10/04

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--) {
        printf("%p - %02x\n", p1, *p1);
        p1++;
    }
}
struct s {
    short i;
    float n;
    signed char c;
};
struct s a = {-1000, 75.75, -25};
int main (void) {
    dump (&a, sizeof(a));
    return 0;
}
```

Mostre o que esse programa irá imprimir quando executado, **explicando** como você chegou aos valores exibidos (mostre suas contas!!). Suponha que a máquina de execução é *little-endian*.

2. (3,5 pontos) Traduza a função *main* abaixo para o assembly usado nos laboratórios (IA32, sintaxe Linux). Use a função *printnum*, vista nos laboratórios, para traduzir a chamada a *printf*.

```
#include <stdio.h>

int a[5] = {2,2,1,3,4};
int b[5] = {10,11,12,13,14};

int main (void) {
    int soma = 0;
    int i;
    for (i=0;i<5;i++)
        soma += a[i] + b[a[i]];
    printf("%d\n", soma);
    return 0;
}
```

(Não se preocupe com a declaração de *a* e *b* no programa assembly, apenas suponha que os nomes *a* e *b* se referem ao início dessas estruturas de dados)

3. (3,5 pontos) Escreva um trecho de código assembly que, dado um número em formato *float* no registrador *%eax*, produza o valor inteiro correspondente (truncando a parte decimal, se existir), deixando esse resultado também no registrador *%eax*. Por exemplo, se *%eax* contiver o valor 0x42978000, após a execução desse trecho de código deverá conter o valor 0x0000004b. Assuma que o valor do *float* é sempre positivo.