



Departamento de Informática
PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO



Rua Marquês de S. Vicente, 225, Gávea
Secretaria de Graduação da Informática
Rio de Janeiro, RJ 22453-900
Tels: (21) 3114-1500 Fax: (21) 3114-1530
<http://www.inf.puc-rio.br>

Curso : Engenharia de Computação
Disciplina : INF1612/INF1600 — Software Básico
Professor : Renato Cerqueira
Período : 2002.1

Nota

Aluno: _____

Matrícula: _____

1ª. Prova

Instruções:

1. A prova deve ser realizada sem consulta a qualquer material.
2. A prova pode ser feita a lápis.
3. Não é permitido o uso de calculadoras.
4. A interpretação das questões faz parte da prova.
5. Não desgrampeie a prova.
6. Coloque o seu nome na folha de respostas.
7. Esta prova tem 4 (quatro) páginas com um total de 4 (quatro) questões; confira a sua antes de começar.

Boa Sorte !!



1ª Questão (2,5 pontos)

Considere o código do programa em C abaixo:

```
#include <stdio.h>

/* Funcao que imprime um dump de memoria */
void dump (void *p, int n) {
    unsigned char *p1 = (unsigned char *) p;
    printf("Numero de bytes: %d", n);
    while (n-- > 0) {
        printf("%p - %02x\n", p1, *p1); /*atenção! Valores impressos em HEXA*/
        p1++;
    }
}

struct X {
    short a;
    char b;
    int c;
} x = {-1024, -300, 640};

struct Y {
    int val;
    char c;
} y[] = {{130, 68}, {-120, -32}};

int main (void) {
    dump(&x, sizeof(x));
    dump(&y, sizeof(y));
    return 0;
}
```

Considerando que o início da `struct x` será carregado no endereço `00237A20` (hexadecimal) e o da `struct y` será carregado no endereço `00237F00`, mostre qual será a saída desse programa **justificando** com os cálculos necessários.



2ª Questão (1,0 ponto)

Faça uma função para testar se a sua plataforma de execução (processador) é *little-endian* ou não. Esta função deve ser implementada em C e obedecer o protótipo

```
int isLittleEndian(void);
```

Essa função deve retornar verdadeiro (1) se a plataforma de execução for *little-endian* e falso (0) caso contrário.

3ª Questão (2,5 pontos)

Traduza a função C apresentada a seguir para a linguagem Assembly (da família de processadores Intel Pentium), utilizando as convenções “usuais” de passagem de parâmetros e retorno de valores de funções em C.

```
struct Arv {  
    int info;  
    struct Arv* esquerda;  
    struct Arv* direita;  
};  
  
int somatorio (struct Arv* a){  
    if (a == 0)  
        return 0;  
    else  
        return a->info + somatorio(a->esquerda) + somatorio(a->direita);  
}
```



4ª Questão (4,0 pontos)

Considerando o código C apresentado abaixo, faça:

- a) a tradução da função `main` para a linguagem Assembly (da família de processadores Intel Pentium);
- b) a implementação em Assembly da função `calc_medias`, que recebe como parâmetros três vetores de inteiros (`medias`, `p1`, `p2`) e o comprimento desses vetores (`n`), e preenche cada posição do vetor `medias` com a média dos valores contidos nas posições correspondentes dos vetores `p1` e `p2`.

Utilize as convenções “usuais” de passagem de parâmetros e retorno de valores de funções em C.

Obs.: Assuma a existência da função `le_notas`.

```
#include <stdio.h>

void le_notas(int n, int* notas);
void calc_medias(int n, int* medias, int* p1, int* p2);

int main (void) {
    int p1[50], p2[50], medias[50];
    int i;

    le_notas(50, p1);
    le_notas(50, p2);
    calc_medias(50, medias, p1, p2);
    for (i = 0; i < 50; i++)
        printf("Media do Aluno %d: %d\n", i + 1, medias[i]);

    return 0;
}
```