

P2 de Software Básico – 2002.1

Prof. Renato Cerqueira

Departamento de Informática — PUC-Rio

1. (4.0 pontos) Considere o código do programa em C abaixo:

```
#include <stdio.h>

void dump (int n, void* p) {
    unsigned char* pc = (char*) p;
    int i;
    printf("=====\n");
    printf("N. de bytes: %d\n",n);
    for (i = 0; i < n; i++,pc++)
        printf("%p - %02x\n", pc, *pc);
    printf("=====\n");
}

struct {
    char c;
    short s1;
    float f;
    short s2;
} x[] = { {'d', 0xe89f, -34.5, -34}, {10, 64, 130.6, 1024}};

struct {
    float f;
    double d;
    char n[5];
} y = {-0.625, 0.625, {'a','b','d',10,0}};

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

Considerando que x e y serão carregados a partir dos endereços 0x00237A20 e 0x00237F00 respectivamente, mostre qual será a saída desse programa, **justificando** com os cálculos necessários.

Obs. 1: o valor ascii de 'a' é 97 (0x61).

Obs. 2: assumo a arquitetura de computador que temos visto no laboratório (int de 4 bytes, *little-endian* e alinhamento “usual”).

2. (4.0 pontos) Traduza as funções abaixo para a linguagem Assembly da família de processadores Intel Pentium, utilizando as convenções "usuais" de alinhamento e de passagem de parâmetros e resultados em C. Comente o seu código.

(a) `struct Node { double value; struct Node* next; };`

```
double sumlist (struct Node* l, int offset) {  
    double d = 0.0;  
    for (; l != 0; l = l->next)  
        d += l->value;  
    return d + offset;  
}
```

(b) `double map (double a);`

```
double prod (double* a, int n) {  
    double d = 1.0;  
    int i;  
    for (i = 0; i < n; i++)  
        d *= map(a[i]);  
    return d;  
}
```

3. (2.0 pontos) Liste todos os símbolos exportados e importados pelo módulo abaixo, sob o ponto de vista do link-editor (isto é, apenas os símbolos **efetivamente** exportados e importados).

```
extern int a;  
extern float b;  
extern int c;  
  
float b = 0.0;  
  
void f (void);  
void g (int i);  
  
static int j = 23;  
  
static void f1 (void) {  
    g(c+j);  
}  
  
int main (void) {  
    f1();  
    return 0;  
}
```