

PUC-Rio – Software Básico – INF1018
Prova 2 – Turma 3wb – 16/06/2016

1. (2,0 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 X {
    int i;
    double d;
    float f;
} x = {-258, 4.0, -13.625};

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

Supondo que `x` seja alocado na posição de memória `0x601040`, mostre o que esse programa irá imprimir quando executado. Considere que a máquina de execução é *little-endian* com as convenções de alinhamento do Linux no IA-64 (vistas em sala). Coloque **PP** nas posições correspondentes a *padding*.

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

2. Traduza as funções `foo` e `boo` abaixo para assembly IA-64 (o assembly visto em sala), utilizando as regras usuais de alinhamento, passagem de parâmetros, salvamento de registradores e resultados em C/linux. Traduza o mais diretamente possível o código de C para assembly.

(Não se preocupe em entender o que as funções fazem, apenas traduza-as literalmente.)

Comente seu código!

- (a) (2,5 pontos)

```
double f1(double x);
double f2(double y);

double foo (float *a, int n) {
    double acc = 0.0;
    while (n--) {
        acc += f1(*a) * f2(*a);
        a++;
    }
    return acc;
}
```

(b) (2,5 pontos)

```
#define MAX 4
int g1(int i);
int g2(int v[], int n);

int boo(int n) {
    int i, lim;
    int vals[MAX];
    lim = (n < MAX)? n : MAX; /* if (n < MAX) lim = n; else lim = MAX; */
    for (i = 0; i < lim; i++) {
        vals[i] = g1(i);
    }
    return g2(vals, lim);
}
```

3. Considere o módulo abaixo:

```
#include <stdio.h>
#define TAM 2

extern float val_max;
void guarda(void *v, int n);

float vals[TAM] = {0.5, 2.0};

int main() {
    int i;
    for (i = 0; i < TAM; i++) {
        if (vals[i] > val_max)
            printf("achei maior em %d\n",i);
    }
    guarda(vals,TAM*sizeof(float));
    return 0;
}
```

(a) (1,0 ponto) Liste os símbolos exportados (definidos neste módulo) e importados (referências externas) para este arquivo, ou seja, o que apareceria como D (dados exportados), T (código exportado) ou U (undefined) na listagem do nm.

(b) (1,0 ponto) Considere agora o seguinte módulo:

```
double val_max = 1.0;
void guarda(void *v, int n) {
    ... /* algum código */
}
```

Se este módulo fosse “ligado” com o módulo anterior para compor um programa executável, o que seria impresso por este programa? **Justifique a sua resposta!**

Boa Prova!