Universidade Federal Fluminense
Campus de Rio das Ostras
Instituto de Ciência e Tecnologia

Programação de Computadores II
Primeira Prova (P2) 30/11/2017
Prof. Alessandro Copetti

Nome: _____**GABARITO**_____

Obs.: A prova pode ser feita a lápis!

1. O que será impresso na tela? Mostre o valor que cada variável assume durante a execução e também circule a resposta; não é para explicar o que cada linha faz.

a) (1pt) Obs: em caso de sucesso, sprintf retorna o tamanho da string criada.

```
#include <stdio.h>

int main () {
  char buffer [50];
  int n, a=5, b=3;
  n=sprintf (buffer, "%d mais %d = %d", a, b, a+b);
  printf ("[%s] tem tamanho %d\n",buffer,n);
  return 0;
}
```

[5 mais 3 = 8] tem tamanho 12

b) (1pt)

```
main() {
    char * estado[27] =
{"AC","AL","AM","AP","BA","CE","DF","ES","GO","MA","MG","MS","MT","PA","PB","PE","PI","PR","RJ
","RN","RO","RR","RS","SC","SE","SP","TO"};
    char uf[5];
    strcpy(uf,estado[5]);
    strcat(uf,estado[8]);
    printf("\n%s", uf);
    printf("\n%c",estado[1][1]);
}
```

CEGO
L

2. Tendo:

```
typedef struct ponto {
    int x;
    int y;
} Ponto;

typedef struct circulo {
    Ponto p;
    int raio;
} Circulo;

int main() {
    Ponto p1 = {1,2};
    Circulo c1 = {20,40,60}, c2 = {30,50,70}, *vet;

    mostraCirculo(c1);
    vet = alocaVetorCirculo(10);
    dobro(&c1);
}
```

a. (2pts) Desenvolva a função mostraCirculo que recebe um Circulo e chama uma função mostraPonto (para mostrar o seu x e y) e também mostra o raio.

b. (2pts) Desenvolva a função alocaVetorCirculo, a qual retorna um novo vetor de Circulo, com todos os campos zerados.

c. (2pts) Desenvolva a função dobro que dobra os valores de x, y e raio.

```
#include <stdio.h>
#include <stdlib.h>

typedef struct ponto {
   int x;
   int y;
} Ponto;

typedef struct circulo {
   Ponto p;
   int raio;
} Circulo;

void dobro(Circulo *p);
Circulo *alocaVetorCirculo(int n);
void mostraPonto(Ponto p);
void mostraCirculo(Circulo c);

int main() {
   Ponto p1 = {1,2};
   Circulo c1 = {20,40,60}, c2 = {30,50,70}, *vet;

   mostraCirculo(c1);
   vet = alocaVetorCirculo(10);
   dobro(&c1);

}

void dobro(Circulo *c) {
   c->p.x=c->p.x*2;
   c->p.y=c->p.y*2;
   c->raio=c->raio*2;
}

void mostraPonto(Ponto p) {
   printf("\nx=%d, y=%d", p.x, p.y);
}

void mostraCirculo(Circulo c) {
   mostraPonto(c.p);
   printf("\nraio=%d", c.raio);
}

Circulo *alocaVetorCirculo(int n) {
   Circulo * vet = (Circulo *) malloc(n * sizeof(Circulo));
   int i;
   for (i=0; i<n;i++) {
      vet[i].p.x=0;
      vet[i].p.y=0;
      vet[i].raio=0;
   }
   return vet;
}
```

```
void gravaCirculo(Circulo *vetor, int n, char *nomeArq) {
   FILE *fp=fopen(nomeArq, "wb");
   fwrite(vetor, n, sizeof(Circulo), fp);
   fclose(fp);
}

void abreCirculo(Circulo *vetor, int n, char *nomeArq) {
   FILE *fp=fopen(nomeArq, "rb");
   fread(vetor, n, sizeof(Circulo), fp);
   fclose(fp);
   int i=0;
   for (;i<n;i++) mostraCirculo(vetor[i]);
}
```

3) (2 pts) Para promover a remoção de um nó, considerando que há um ponteiro lst que aponta para o início da lista, quais são as duas linhas de código faltantes abaixo?

```
Lista* lst_retira (Lista* lst, int val) {
   Lista* ant = NULL;
   Lista* p = lst;
   while (p != NULL && p->info != val) {
      ant = p;
      p = p->prox;
   }
   if (p == NULL) return lst;
   if (ant == NULL)

            _____________________________
   else

            _____________________________
   free(p);
   return lst;
}
```

```
if (ant == NULL)
      lst = p->prox;
else
      ant->prox = p->prox;
```