

INF 1620 – P1 - 16/09/06	Questão 1
Nome:	
Matrícula:	Turma

Considerando a fórmula para o cálculo da distância entre dois pontos (x_1, y_1) e (x_2, y_2) apresentada a seguir:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- (a) Escreva uma função que receba como parâmetros as coordenadas de dois pontos e retorne a distância entre eles, seguindo o protótipo:

```
float distancia (float x1, float y1, float x2, float y2);
```

Obs: para o cálculo da raiz quadrada, utilize a função `sqrt` da biblioteca padrão da linguagem C.

- (b) Escreva um programa C completo que capture as coordenadas de dois pontos, fornecidas pelo usuário via teclado, e calcule e imprima a distância entre eles. Este programa deve fazer uso da função do item anterior.

INF 1620 – P1 - 16/09/06	Questão 2
Nome:	
Matrícula:	Turma

Considere uma função `distribuicao` que calcula a distribuição das notas de uma turma em três faixas de valores definidas a seguir:

aprovados:[5.0,10.0]

prova_final:[3.0,5.0)

reprovados:[0.0,3.0)

O programa abaixo ilustra o uso dessa função, imprimindo os valores 3, 2 e 1, que representam neste exemplo as quantidades de alunos aprovados, em prova final e reprovados, respectivamente.

```
int main (void) {
    float v[6] = {7.5, 3.2, 5.0, 4.7, 9.6, 1.5}; /*notas*/
    int aprov, prova_final, reprov;
    distribuicao(6, v, &aprov, &prova_final, &reprov);
    printf("%d %d %d\n", aprov, prova_final, reprov);
    return 0;
}
```

Implemente a função `distribuicao` de tal forma que o exemplo acima funcione adequadamente. A solução deve funcionar para qualquer vetor de números de ponto flutuante (`float`) com valores entre 0.0 e 10.0.

INF 1620 – P1 - 16/09/06	Questão 3
Nome:	
Matrícula:	Turma

Implemente uma função que receba como parâmetros um vetor de números inteiros v e seu número de elementos n , e verifique se os elementos do vetor estão armazenados em ordem crescente, isto é, se o primeiro elemento é menor ou igual ao segundo, o segundo é menor ou igual ao terceiro, e assim sucessivamente. Caso os elementos do vetor estejam ordenados, a função deve retornar 1. Caso o vetor não esteja ordenado, a função deve retornar 0. Considere que vetores com 1 ou 0 elementos estão ordenados. Essa função deve ter o seguinte protótipo:

```
int ordenado (int n, int* v);
```

INF 1620 – P1 - 16/09/06	Questão 4
Nome:	
Matrícula:	Turma

Escreva uma função em C que receba como parâmetros duas strings e retorne uma nova string, alocada dinamicamente, com a concatenação das duas strings separadas por um caractere '-'. Por exemplo, se forem passadas como parâmetros as cadeias de caracteres "PUC" e "Rio", a função deve retornar a cadeia "PUC-Rio". As strings passadas como parâmetros não podem ser alteradas. Essa função deve ter o seguinte protótipo:

```
char* concatena (char* s1, char* s2);
```

RASCUNHO

Respostas nesta folha não serão consideradas.

Protótipos de funções que podem ser úteis:

stdio.h:

```
int scanf (char* formato, ...);
int printf (char* formato, ...);
FILE* fopen (char* nome, char* modo);
int fclose (FILE* fp);
int fscanf (FILE* fp, char* formato, ...);
int fprintf (FILE* fp, char* formato, ...);
char* fgets(char* str, int size, FILE* fp));
int sscanf(char* str, char* formato, ...);
```

math.h:

```
double sqrt (double x);
double pow (double x, double exp);
double cos (double radianos);
double sin (double radianos);
```

string.h:

```
int strlen (char* s);
int strcmp (char* s, char *t);
char* strcpy (char* destino, char* fonte);
char* strcat (char* destino, char* fonte);
```

stdlib.h:

```
void* malloc (int nbytes);
void free (void* p);
void qsort (void *vet, int n, int tam, int (*comp) (const void*, const void*));
```

Não separe as folhas deste caderno.