

INF 1007 – EXERCÍCIO PARA P1 – 18/09/12	
Nome:	
Matrícula:	Turma:

Instruções:

1. Esta prova deverá ser resolvida em até 45 minutos
2. A prova é sem consulta e sem perguntas. A interpretação do enunciado faz parte da prova.
3. Seguindo as instruções do fiscal, o aluno deve copiar o arquivo **TURMA_MATRICULA_NOME_EXERCICIO.txt** (**onde deverá ser colocada a solução da prova**) para a sua máquina e **RENOMEÁ-LO**, substituindo *TURMA* pela turma do aluno (com 3 dígitos) , *MATRICULA* pela matrícula do aluno (com 7 dígitos) e *NOME* pelo *último nome do aluno*. Por exemplo, “33B_0720870_MARTINS_ **EXERCICIO.txt**” seria o arquivo contendo a solução da prova de um aluno da turma 33B, cuja matrícula é 0720870 , chamado, por exemplo Pedro da Silva Martins. No dia da prova, faça esta cópia imediatamente, porque o local original do arquivo pode ficar inacessível quando a prova começar e a rede for desligada.
4. O cabeçalho que se encontra dentro deste arquivo .txt deve ser preenchido com nome completo, matrícula e turma do aluno. Informações incorretas ou omitidas neste cabeçalho invalidam a prova.
5. Por segurança, este arquivo .txt deve ser salvo a cada 15 minutos, pelo próprio aluno. Somente esse arquivo será considerado a resposta da prova. EVITAR MANTER MAIS DE UMA CÓPIA, PARA NÃO ALTERAR UM ARQUIVO E ENTREGAR O ERRADO.
6. Durante o tempo de prova, o aluno pode, se desejar, testar suas funções criando um projeto no MS Visual Studio. Mas esses arquivos não serão considerados. O único arquivo considerado resposta é o do item 3.
7. Se em alguma questão o aluno criar e usar uma função auxiliar, essa função deve também ser apresentada no arquivo de respostas.

INF1007- DI- EXERCICIO PARA P1 – 18/09/2012

No Departamento de Transporte, para implementar o cadastro dos carros existe um vetor do tipo estruturado Carro definido como se segue:.

```
struct carro
{   int  registro;
    char placa[8];
    char dono[51];
    int  ano;
    int  numMultas;
};
```

```
typedef struct carro Carro;
```

1) Escreva a função *exibeVetorDeCarros* , que recebe um vetor de carros, o número de carros e exibe os dados de todos os carros do vetor.

2) Escreva a função *obtemRegistrosDeUmAno* que recebe os seguintes dados através de seus argumentos de entrada:

- um vetor (de Carro) com carros da concessionária,
- o número de carros,
- um ano , e
- o endereço de uma variável inteira onde a função deverá disponibilizar o número de carros desse ano;

e retorna um novo vetor de inteiros, ou seja, o endereço do primeiro elemento de um novo vetor, com os registros apenas dos carros daquele ano. Esse novo vetor deve ser alocado dinamicamente com o tamanho exato necessário e deverá conter os registros desejados, na mesma ordem em que eles aparecem no vetor original. A função deve também disponibilizar o número de carros desse ano na variável inteira, cujo endereço foi fornecido como um dos argumentos na chamada da função. Caso **não existam carros desse ano** ou **caso não seja possível criar o novo vetor, a função deve retornar NULL.**

OBS: No caso de você querer testar sua função aqui vai um exemplo de como criar facilmente e inicializar um vetor de Carro a ser usado no seu teste

```
Carro vcarro[6] =    {   {4444, "LAN4532", "ana", 2008, 2},
                        {1111, "KCD1531", "leo", 2012, 6},
                        {6666, "LOB4172", "rui", 2010, 3},
                        {8888, "JKD5539", "bia", 2002, 2},
                        {3333, "LDV4511", "edu", 2008, 8},
                        {2222, "KWX2382", "lia", 2012, 7} };
```

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, ...);
```

stdlib.h:

```
void* malloc (int nbytes);
void* realloc (void *p, int nbytes);
void free (void* p);
```

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);
```