

INF 1620 – P4 – 30/06/07	Questão 1
Nome:	
Matrícula:	Turma

O maior divisor comum (MDC) de dois números inteiros é o maior número inteiro que faz uma divisão exata (com resto zero) desses dois números. Por exemplo, o MDC de 32 e 40 é 8, o MDC de 9 e 27 é 9 e o MDC de 13 e 25 é 1.

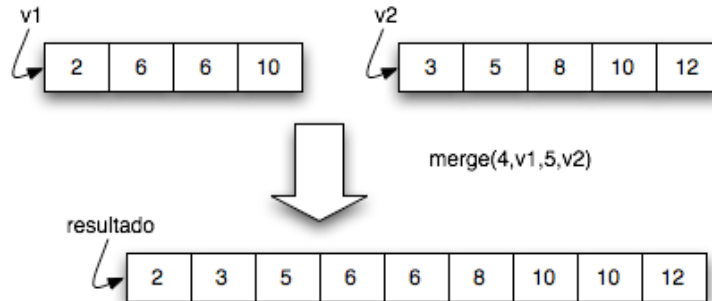
- a) Implemente uma função que receba como parâmetros dois números inteiros x e y e calcule o MDC desses números. Essa função deve ter o seguinte protótipo:

```
int mdc (int x, int y);
```

- b) Dois ou mais números são **primos entre si** quando o máximo divisor comum desses números é **1**. Por exemplo, 13, 25 e 32 são primos entre si. Escreva um programa completo que capture três inteiros, fornecidos pelo usuário via teclado, utilize a função do item anterior para determinar se os três são ou não primos entre si, e imprima uma linha informando o “SIM” se forem primos entre si, ou “NÃO”, caso contrário.

INF 1620 – P4 – 30/06/07	Questão 2
Nome:	
Matrícula:	Turma

Escreva uma função em C que receba dois vetores de números inteiros ordenados em ordem crescente e retorne um novo vetor, alocado dinamicamente, que seja a junção dos dois vetores recebidos como parâmetros, isto é, o novo vetor deve conter todos os elementos dos vetores originais. O vetor retornado também deve estar ordenado em ordem crescente. A figura a baixo ilustra o comportamento da função:



Observe que os vetores podem ter elementos repetidos. O protótipo da função deve ser:

```
int* merge (int n1, int* v1, int n2, int* v2);
```

INF 1620 – P4 – 30/06/07	Questão 3
Nome:	
Matrícula:	Turma

Considere um arquivo texto com as notas dos alunos de uma disciplina. Cada linha do arquivo contém a matrícula de um aluno (cadeia de nove caracteres), seguida pelos valores de suas três notas (P1, P2 e P3). Considere ainda que não existem linhas em branco no arquivo. Um exemplo desse formato é mostrado abaixo.

entrada.txt

9010087-2 2.0 4.3 6.5
8820324-3 7.0 8.2 8.6
9210478-5 6.0 7.5 7.8
9020256-8 3.0 0.5 4.2

Considerando que um aluno está aprovado se tiver média maior ou igual a 5.0, escreva um programa completo que leia o arquivo “entrada.txt”, que contém as notas de uma disciplina no formato descrito acima, e produza dois arquivos texto de saída, os arquivos “aprovados.txt” e “reprovados.txt”. No arquivo texto “aprovados.txt” o programa deve escrever o número de matrícula e a média $((P1+P2+P3)/3)$ somente dos alunos aprovados, enquanto no arquivo “reprovados.txt” o programa deve escrever o número de matrícula e a média $((P1+P2+P3)/3)$ apenas dos alunos reprovados. Para o exemplo acima, o programa deve criar os arquivos “aprovados.txt” e “reprovados.txt” com o seguinte conteúdo:

aprovados.txt

8820324-3 7.9
9210478-5 7.1

reprovados.txt

9010087-2 4.3
9020256-8 2.6

Observe que as médias são escritas nos arquivos de saída com apenas uma casa decimal. Se não for possível abrir qualquer um dos arquivos, o programa deve imprimir a mensagem “ERRO” e terminar sua execução.

INF 1620 – P4 – 30/06/07	Questão 4
Nome:	
Matrícula:	Turma

Considere uma lista encadeada que representa um conjunto de alunos. O tipo que representa cada nó da lista é dado por:

```
struct lista {
    int mat;           /* número de matrícula */
    char nome[81];    /* nome do aluno */
    struct lista* prox; /* ponteiro para próximo elemento
*/
};
typedef struct lista Lista;
```

Escreva uma função que retire um aluno da lista. A função deve receber como parâmetro o ponteiro para o primeiro nó da lista e o nome do aluno a ser retirado, e deve ter como valor de retorno o ponteiro para o primeiro nó da lista após a retirada do aluno. Considere que não há nomes repetidos na lista. O protótipo dessa função é dado por:

```
Lista* retira (Lista* l, char* n);
```

INF 1620 – P4 – 30/06/07	Questão 5
Nome:	
Matrícula:	Turma

Considere uma *árvore binária de busca* que armazena os dados dos alunos de uma turma, usando a média do aluno como critério de ordenação. As médias dos alunos associados aos nós da sub-árvore à esquerda são menores que a média do aluno associado à raiz, e as médias dos alunos associados aos nós da sub-árvore à direita são maiores ou iguais à média do nó da raiz. O tipo que representa um nó da árvore é dado por:

```
struct arv {
    char nome[81];    /* nome */
    float media;     /* media obtida */
    struct arv* esq;
    struct arv* dir;
};
typedef struct arv Arv;
```

Escreva uma função que receba como parâmetros uma árvore definida pela estrutura acima e os limites de um intervalo fechado de notas, e retorne quantos alunos têm médias dentro deste intervalo. O protótipo dessa função é dado por:

```
int retira (Arv* a, float nota_menor, float nota_maior);
```

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);
```

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.