

INF 1620 – P4 - 06/12/02	Questão 1
Nome:	
Matrícula:	Turma:

(a) Escreva uma função em C que receba como parâmetro um número inteiro não negativo e retorne, como resultado da função, o valor do seu fatorial. O cabeçalho desta função deve ser:

```
int fat (int n);
```

(b) Escreva um programa completo que leia um número inteiro fornecido pelo usuário via teclado e imprima o valor do fatorial desse número, usando a função escrita no item acima. Pode-se assumir que o usuário nunca entrará com um valor menor do que zero.

INF 1620 – P4 - 06/12/02	Questão 2
Nome:	
Matrícula:	Turma:

Escreva uma função em C para converter uma cadeia de caracteres para letras maiúsculas. A função deve receber como parâmetro de entrada uma cadeia de caracteres e ter como valor de retorno uma nova cadeia, cujo espaço de memória deve ser alocado dinamicamente pela função, contendo a cadeia original convertida para letras maiúsculas. Por exemplo, se for passada a cadeia de caracteres “Puc-Rio”, a função deve retornar a cadeia “PUC-RIO”. A assinatura da função deve ser:

```
char* converte (char* s);
```

Dica: Se *c* representa um caractere minúsculo, o caractere maiúsculo correspondente pode ser dado por: `c-'a'+'A'`.

Protótipos de funções que podem ser úteis:

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

Não separe as folhas deste caderno. Todas as folhas devem ter seu nome. Responda cada questão na folha correspondente. Use o verso se necessário.

INF 1620 – P4 - 06/12/02	Questão 3
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. O protótipo dessa função é dado por:

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

Protótipos de funções que podem ser úteis:

```
int strcmp (char* s, char *t);
```

Não separe as folhas deste caderno. Todas as folhas devem ter seu nome. Responda cada questão na folha correspondente. Use o verso se necessário.

INF 1620 – P4 - 06/12/02	Questão 4
Nome:	
Matrícula:	Turma:

Considere uma *árvore binária de busca* que armazena números inteiros, onde os valores associados aos nós da sub-árvore à esquerda são menores que o valor associado à raiz e que os valores dos nós da sub-árvore à direita são maiores. O tipo que representa um nó da árvore é dado por:

```
struct arv {
    int val;
    struct arv* esq;
    struct arv* dir;
};
typedef struct arv Arv;
```

Pede-se

- (a) Escreva uma função que insira um novo nó em uma árvore de busca, de tal forma que a propriedade de ordenação dos nós da árvore seja mantida. A função recebe como parâmetro a raiz da árvore e o número inteiro a ser inserido, e deve ter como retorno o valor atualizado da raiz. O protótipo dessa função deve ser:

```
Arv* insere (Arv* a, int n);
```

- (b) Escreva uma função para imprimir os valores associados aos nós da árvore em ordem *crescente*. O protótipo dessa função é dado por:

```
void imprime (Arv* a);
```

INF 1620 – P4 - 06/12/02	Questão 5
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 {
    char mat[10];          /* número de matrícula */
    char nome[81];        /* nome do aluno */
    float cr;             /* coeficiente de rendimento */
    struct lista* prox;   /* ponteiro para o próximo elemento */
};
typedef struct lista Lista;
```

Escreva uma função que salve em um arquivo texto os dados de uma lista encadeada de alunos. Essa função recebe como parâmetros o nome do arquivo onde os dados devem ser salvos e um ponteiro para o primeiro elemento da lista encadeada. Se os elementos forem salvos com sucesso, a função deve retornar o valor 1; caso contrário, se o arquivo de saída não puder ser aberto, a função deve retornar o valor 0. O protótipo da função é dado por:

```
int salva(char *nomearq, Lista* l);
```

Obs.: Cada linha do arquivo criado por essa função deve conter a matrícula de um aluno seguida pelo seu nome e pelo valor do seu CR. Um exemplo desse formato é mostrado a seguir.

```
9010087-2 Apu Silva 6.5
8820324-3 João Prasad 8.7
9210478-5 Manuel Roman 7.2
```

Protótipos de funções que podem ser úteis:

```
FILE* fopen (char* nome, char* modo);
int fclose (FILE* fp);
int fprintf (FILE* fp, char* formato, ...);
```

Não separe as folhas deste caderno. Todas as folhas devem ter seu nome. Responda cada questão na folha correspondente. Use o verso se necessário.

RASCUNHO

Respostas nesta folha não serão consideradas.

Não separe as folhas deste caderno.