

MAC 110 – Introdução à Computação – BM e BMA

Terceiro Exercício-Programa (EP3): Lojas PontoChic
Informações adicionais

Data de entrega: **12 de junho de 2012**

1 Introdução

No EP3 você fará uso dos seguintes novos tópicos: variáveis dos tipos `char` (para caracteres) e `float` ou `double` (para números reais), comando `switch`, funções, leitura e criação de arquivos como descrito na seção seguinte. Além disso, você implementará o chamado método de Newton para ‘determinar raízes de funções’. Ao fazer o EP3 você irá aprender vários conceitos novos, todos muito importantes.

2 Entrada e Saída – uso de arquivos

No EP3 é pedido que você faça leitura e criação de arquivos de entrada e de saída. A seguir descrevemos como você pode fazer isso.

O seu programa deverá ler os dados de um arquivo de entrada. Para isso, você pode criar, no mesmo diretório onde se encontra o seu programa, um arquivo chamado “`entrada.txt`”, com o formato de conteúdo como especificado abaixo (e também no enunciado do EP3). Para criar esse arquivo, você pode usar o próprio ambiente do `CodeBlocks`. Para tanto, abra um arquivo novo, digite os dados abaixo (ou outros que você quiser), e salve-o com o nome “`entrada.txt`”.

```
19
v 70.00 3.2 15
p 850.05 3.2 15
n 850.05 70.0 3.2
j 850.05 70.0 15
v 70 1 15
p 980.26 1 15
n 980.25 70.0 1
j 980.25 70.0 15
v 90 50 2
p 150 50 2
n 150 90 50
j 150 90 2
v 1000 200 10
p 1499.97 200 10
n 1499.97 1000 200
j 1499.97 1000 10
n 1200 100 10
n 20810 1000 5
n 20805.8 1000 5
```

Para fazer leitura dos dados que estão no arquivo `entrada.txt` que você criou, você deve usar o seguinte programa. Além de ler um arquivo de nome `entrada.txt` este programa cria no disco um arquivo chamado `saida.txt` (com a saída/resposta do seu programa).

```
#include <stdio.h>

#define ENTRADA "entrada.txt"
#define SAIDA "saida.txt"

int main()
{
    FILE *entrada, *saida;
    int n;
    float v, p, p_max, j;
    char codigo;

    entrada = fopen(ENTRADA,"r"); /* Abre o arquivo entrada.txt para leitura */
    if (entrada == NULL){        /* 0 arquivo existe? */
        printf("ERRO: arquivo de entrada nao encontrado\n");
        return -1;              /* Indica que houve erro */
    }

    saida = fopen(SAIDA,"w");    /* Abre o arquivo saida.txt para escrita */
    if (saida == NULL){
        printf("ERRO: arquivo de saida nao pode ser gravado!\n");
        return -1;
    }

    /* escreva um cabeçalho de identificacao do tipo abaixo no arquivo de saida */

    fprintf(saida, "*****\n");
    fprintf(saida, "***                               ***\n");
    fprintf(saida, "***   MAC0110 - Introducao a Computacao - 1o. semestre 2012   ***\n");
    fprintf(saida, "***   Terceiro Exercicio-Programa (EP3)                   ***\n");
    fprintf(saida, "***                               ***\n");
    fprintf(saida, "***   Aluno :                               ***\n");
    fprintf(saida, "***   Numero USP :                            ***\n");
    fprintf(saida, "***   Curso :                               ***\n");
    fprintf(saida, "***                               ***\n");
    fprintf(saida, "*****\n\n\n");

    while (!feof(entrada)){      /* Enquanto o fim de arquivo ("end of file")
                                   não foi encontrado... */
        fscanf(entrada, "%c", &codigo);

        switch (codigo) {
            case 'v':
                fscanf(entrada, "%f %f %d ", &p, &j, &n);
                fprintf(saida, "Queremos calcular o valor a vista!\n");
                fprintf(saida, "Valor da prestacao: %.2f\n", p);
                fprintf(saida, "Valor dos juros: %.2f%\n", j);
                fprintf(saida, "Numero de meses: %d\n\n", n);
                break;
        }
    }
}
```

```

case 'p':
    fscanf(entrada, "%f %f %d ", &v, &j, &n);
    fprintf(saida, "Queremos calcular o valor das prestacoes!\n");
    fprintf(saida, "Valor a vista: %.2f\n", v);
    fprintf(saida, "Valor dos juros: %.2f%\n", j);
    fprintf(saida, "Numero de meses: %d\n\n", n);
    break;

case 'j':
    fscanf(entrada, "%f %f %d ", &v, &p, &n);
    fprintf(saida, "Queremos calcular o valor dos juros!\n");
    fprintf(saida, "Valor a vista: %.2f\n", v);
    fprintf(saida, "Valor da prestacao: %.2f\n", p);
    fprintf(saida, "Numero de meses: %d\n\n", n);
    break;

case 'n':
    fscanf(entrada, "%f %f %f ", &v, &p_max, &j);
    fprintf(saida, "Queremos calcular o valor da prestacao e o numero de meses!\n");
    fprintf(saida, "Valor a vista: %.2f\n", v);
    fprintf(saida, "Valor maximo da prestacao: %.2f\n", p_max);
    fprintf(saida, "Valor dos juros: %.2f%\n\n", j);
    break;

default:
    /* Leia e ignore os caracteres ate' o fim da linha. */
    while (codigo != '\n' && !feof(entrada))
        fscanf(entrada, "%c", &codigo);
    break;
}
}

fclose(entrada); /* Libera os recursos do sistema que estavam sendo usados */
fclose(saida); /* para controlar o acesso aos arquivos. */

return 0; /* Terminou normalmente. */
}

```

3 Observações Gerais

1. Só use os recursos da linguagem C que foram vistos em aula e os que fazem parte desse enunciado.
2. Você pode considerar que os valores de entrada (ou seja, os valores à vista, das prestações, dos juros e o número de prestações) são sempre positivos, e dentro dos intervalos descritos no enunciado.
3. Para facilitar os testes a serem feitos durante a correção dos programas, deixe a saída idêntica (no conteúdo e na forma) ao do exemplo do enunciado.
4. Não esqueça de seguir as observações muito importantes que estão em **Informações sobre a entrega dos EPs**.