Departamento de Ciência da Computação – IME-USP

MAC 115 – Introdução à Computação para Ciências Exatas e Tecnologia

Escola Politécnica – Primeiro Semestre de 1998

Terceiro Exercício-Programa — Data de entrega: até 25 de maio de 1998.

Turmas: Civil C, Elétrica B, Elétrica D e Minas/Metal/Materiais.

AGÊNCIA BANCÁRIA

O Banco Itadesco abriu uma nova agência nas proximidades da Cidade Universitária e decidiu contratar os alunos da Escola Politécnica para escrever um programa que possibilite o controle de suas contas correntes.

O programa deve trabalhar com as seguintes informações sobre contas correntes:

Número da conta;
Nome do titular;
Endereço para correspondência;
Data de abertura da conta;
Tipo da conta (Especial ou Comum);
Saldo atual;
Limite do Cheque Especial;
Vencimento do Contrato.

Deve-se permitir ao usuário escolher uma dentre as seguintes operações:

- ler um arquivo com as informações das contas já cadastradas
- listar o número da conta e o nome do titular das contas já cadastradas;
- buscar uma conta a partir do seu número ou a partir do nome do titular (ou parte desta informação);
- inserir informações sobre uma nova conta;
- eliminar uma conta do cadastro;
- alterar as informações sobre uma conta cadastrada.

Para entender melhor como o seu programa deve funcionar, instalamos na rede do CCE um arquivo executável de exemplo. O arquivo se chama `banco.exe` e está instalado no diretório `i:\mac115` da rede do CCE. Também neste diretório, você vai encontrar um arquivo com dados, `banco.txt`, que você pode usar como cadastro inicial.

Execute `banco` e opte por ler um arquivo de dados. Então escolha como arquivo o `banco.txt`. A partir daí, execute o que você quiser e veja o resultado.

O seu programa deve funcionar de forma semelhante. Você deve implementar as opções que você vê quando executa o nosso exemplo.

As funções que fazem leitura ou gravação das informações de um arquivo ou do teclado devem ser escritas de forma consistente. Para ajudar, instalamos, no mesmo diretório mencionado acima, o arquivo `alunos.c`, com um programa em C que manipula dados de alunos (nome do aluno, número de créditos e média). Baseie-se neste programa para escrever as suas funções de leitura, gravação, edição e inserção.

O seu programa deve conter as seguintes funções:

1. ```
   int leitura (char *conta[], char *titular[], char *ender[],
                char *abertura[], char tipo[], char *vencimento[],
                float saldo[], float limite[] );
   /* Lê o nome de um arquivo e, caso o arquivo exista, lê os dados nele gravados
      e os guarda nos vetores conta, titular, ender, abertura, tipo, vencimento,
      saldo, limite. Retorna o número de registros (ou seja, cadastros de contas)
      lidos. Caso o arquivo não exista, o programa imprime uma mensagem e devolve
      0. */
   ```

2. ```
   void gravacao (int n, char *conta[], char *titular[], char *ender[],
                  char *abertura[], char tipo[], char *vencimento[],
                  float saldo[], float limite[] );
   /* Grava em um arquivo em disco as informações que estão na memória,
      nas n primeiras posições dos vetores conta, titular, ender, abertura, tipo,
      vencimento, saldo, limite.  */
   ```

3. ```
   void lista(int i, char *conta[], char *titular[],
              char *ender[], char *abertura[]. char tipo[],
              char *vencimento[], float saldo[], float limite[] );
   /* Lista as informações guardadas na posição i dos vetores conta, titular,
      ender, abertura, tipo, vencimento, saldo, limite. */
   ```

4. ```
   int insercao (int n, char *conta[], char *titular[], char *ender[],
                 char *abertura[], char tipo[], char *vencimento[],
                 float saldo[], float limite[] );
   /* Insere as informações de uma nova conta na posição n dos vetores.
      Nao permite a inserção de um número de conta já existente.  */
   ```

5. ```
   void editar(int i, char *conta[], char *titular[], char *ender[],
               char *abertura[], char tipo[], char *vencimento[],
               float saldo[], float limite[] );
   /* Mostra as informações guardadas na posição i dos vetores conta, titular,
      ender, abertura, tipo, vencimento, saldo, limite, possibilitando ao usuário
      a alteração de  qualquer uma destas informações. */
   ```

6. ```
   `void remover(int i, int *n, char *conta[], char *titular[],
                 char *ender[], char *abertura[], char tipo[],
                 char *vencimento[], float saldo[], float limite[] );
   /* Remove as informações guardadas na posição i dos vetores. */
   ```

7. ```
   int buscapalavra(int n, char *conta[], char *titular[], char *ender[],
                    char *abertura[], char tipo[], char *vencimento[],
                    float saldo[], float limite[]);
   /* Lê uma palavra do teclado e busca esta palavra nas n primeiras
      posições dos vetores titular e ender. Mostra um a um os registros
      contendo a palavra, e, para cada um, permite a opção de editar o tal
      registro. */
   ```

8. ```
   int buscaconta(char *conta[], char *titular[], char *ender[],
                  char *abertura[], char tipo[], char *vencimento[],
                  float saldo[], float limite[]);
   /* Lê um número de  conta do teclado e busca este número no vetor conta.
      Caso tal conta exista, permite a edição do registro correspondente a mesma.
   */
   ```

9. ```
   int listatitulo(int *n, char *conta[], char *titular[],char *ender[],
                   char *abertura[], char tipo[], char *vencimento[],
   ```
   ```

```
                float saldo[], float limite[]);
/* Lista o  número e o nome do titular de todas as contas do cadastro, de tal
   modo que o usuário possa pedir a informação completa de alguns dos clientes.
*/
```

Observações

- Não use variáveis globais.
- Sinta-se à vontade para corrigir os erros ou implementar outras opções que não constem do programa fornecido `banco.exe`.
- Faça declaração de variáveis e alocação de memória similar às feitas no programa fornecido `alunos.c`. De outra forma, não haverá memória disponível no QuickC para compilar seu programa.
- Você deve entregar uma listagem impressa de seu programa e um disquete contendo, além de seu programa em C, uma versão executável do mesmo.
- Todas as observações gerais dadas anteriormente valem para este exercício-programa.
- O quarto exercício-programa vai ser uma extensão deste, portanto FAÇA UMA CÓPIA DO SEU EXERCÍCIO-PROGRAMA ANTES DE ENTREGÁ-LO. Você vai precisar.