



Nome do Aluno: _____

Nro. de Matrícula: _____ - ____

Peso: 4.0

PROVA - GRAU A

Faça um **programa modular**, usando **rotinas genéricas** de manipulação de **estruturas de dados com alocação estática ou dinâmica (pilha, fila circular e/ou deque circular usando arrays ou ponteiros)**, para auxiliar o funcionário do caixa do Banco ACME a atender aos clientes. Este programa deve seguir rigorosamente a especificação fornecida logo abaixo e **não deve usar variáveis globais**.

Resumo: Os clientes e suas transações estão descritos em um arquivo texto em disco, que deve ser lido, formando a fila de clientes a serem atendidos pelo caixa do Banco. Os clientes serão atendidos, de modo a realizar seus depósitos e saques, na ordem definida no arquivo, mas também respeitando a seguinte regra: se não tiver dinheiro em caixa para o saque, o cliente vai para o final da fila (quando chegar de novo a vez dele, quem sabe poderá então ter dinheiro no caixa). Cada operação feita com o cliente é anotada e deve ser colocada em cima de uma pilha de transações efetuadas, sendo acumuladas as diferentes operações de um mesmo cliente – caso existam. Ao final do dia o caixa deve gerar um relatório, gravando em disco um arquivo texto com as movimentações feitas pelos clientes, onde estes não devem aparecer mais de uma vez listados neste arquivo.

- **Ler um arquivo texto**, cujo nome é fornecido pelo usuário, contendo uma lista de nomes de clientes e valores das operações que estes desejam depositar (valor positivo) ou sacar (valor negativo). Criar uma fila que irá armazenar estes dados na mesma ordem em que se encontram no arquivo em disco (use uma rotina do tipo “insere_na_fila”). **A fila comporta no máximo 20 clientes**. O arquivo texto possui os dados em pares de linhas, onde na primeira linha está o nome do cliente (**nomes com no máximo 30 letras**) e na linha seguinte está o valor da sua transação, seguindo-se dos dados dos demais clientes (2 linhas para cada cliente, **contendo sempre nome e valor da transação em cada linha**). O arquivo será similar ao apresentado no exemplo abaixo:

```
Beltrano da Silva
1000.00
Ciclana dos Santos
-350.00
Ciclano da Silva
-800.00
Fulana dos Santos
1400.00
Fulano da Silva
600.00
```

- Uma vez lidos todos os dados do arquivo e inseridos na fila, **exibir na tela os dados contidos nela** (use uma rotina do tipo “exibe_dados_fila”). Os dados devem ser listados do mesmo modo como estão exibidos no exemplo acima (do primeiro ao último da fila).
- Crie uma nova função para manipular a fila de clientes: “atende_cliente”. Esta rotina tem como objetivo, **ir retirando um a um os clientes da fila** (atende na ordem da fila) e **tenta realizar a operação solicitada**. Se o valor indicado for positivo, o cliente estará fazendo um depósito, e conseqüentemente aumentando o dinheiro disponível em caixa (começa sem nada no caixa). Se o valor for negativo, o cliente estará fazendo um saque, e conseqüentemente diminuirá o dinheiro em caixa. Caso **não haja dinheiro** suficiente para realizar a transação (saque), o funcionário solicitará para que o cliente vá para o fim da fila e fique aguardando novamente a sua vez de ser atendido. Durante o expediente de um dia o caixa do banco limitará o número de clientes atendidos (operações “atende_cliente”) a um **máximo de 50 operações**.

- Cada **transação efetivamente realizada** é “**anotada**” em um papel e **colocada na pilha de transações do dia**. Entretanto, antes de colocar na pilha de transações do dia, o funcionário examina todas as anotações da pilha, verificando se encontra uma transação já realizada pelo mesmo cliente. Se o cliente já executou uma transação neste mesmo dia, é armazenado no topo da pilha o total de ambas as transações (a transação anterior é removida – joga fora a anotação anterior). Note que este processo é feito a cada nova transação, e que a **pilha deve e tem que ser considerada como uma pilha**.
- Uma vez que não tenha mais clientes para atender (ou já atendeu o máximo de 50 transações), o funcionário deve **exibir na tela o valor total restante em caixa** e fechar o banco, gerando o relatório de transações.
- **Gravar em disco um arquivo texto com o relatório das transações do dia**. Gravar um arquivo em um formato similar ao formato do arquivo de entrada lido, sob o nome “saida.txt”. Este arquivo deverá conter o nome e o valor das transações de cada cliente (dados da pilha de transações).

Exemplo da tela do programa e da interação com este:

```

>>> BANCO ACME <<<
Arquivo de entrada: clientes.txt
Fila de Clientes:
Beltrano da Silva
1000.00
Ciclana da Silva
-350.00
Ciclano da Silva
-800.00
Fulana da Silva
1400.00
Fulano da Silva
600.00

Saldo no final do dia: 1850.00
Gravando arquivo em disco... saída.txt
FIM.

```

Bom trabalho!

Atenção:

- ⇒ Lembre-se de colocar o seu nome e o número de matrícula como comentários nas primeiras linhas do programa fonte no arquivo “PGA.pas”. Lembre-se também que você deve entregar para o professor a folha da prova identificada com o seu nome e número de matrícula
- ⇒ Envie os programas fonte da prova e do trabalho por e-mail (PGA.pas e TGA.pas) para o professor em *attach* na mensagem (chame o professor quando estiver pronto para enviar o mail). Use o seguinte título na mensagem: “*Subject: Prova GA – 41*”. E-mail: osorio@exatas.unisinos.br
- ⇒ **Lembre-se de entregar o trabalho extra-classe**, junto com a prova ou então enviar por mail até a semana que vem! (Enviar TGA.pas e as Units - .pas e .tpu se existirem – **não envie executável**) Só será considerado entregue o trabalho que receber confirmação do recebimento do mail.
- ⇒ Todo outro uso do correio eletrônico (envio/recepção de mensagens) fica proibido durante a realização da prova. O aluno que enviar mensagens ou ler mensagens recebidas durante a prova ficará com ZERO na nota da prova. Usar o mail somente para o envio do programa ao final da prova.
- ⇒ Prova INDIVIDUAL e COM CONSULTA AO MATERIAL PESSOAL (não é permitido emprestar material ao colega, nem o uso *indiscriminado* de programas alheios). Prova com duração até o final do período.