



DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

ESTRUTURA DE DADOS

Prof. Roberto Freitas Parente

roberto.parente@ufba.br

TRABALHO 01

Instruções:

- O trabalho é em equipe de no máximo **3 alunos**;
- Ao final da entrega dos trabalhos, será aplicado detector de plágio, e uma vez detectado plágio, a nota na disciplina será **zerada**;
- **(Alterado)** Data de entrega: 03 de dezembro (domingo) até 23:59 por **e-mail do professor**;
- A entrega deve ser no formato ZIP com o seguinte conteúdo:
 - + Código-fonte (bem como os arquivos `.c` e `.h` referentes aos TAD utilizados);
 - + Arquivo Makefile associado para compilar o código.
- O código deve estar escrito em Linguagem **C** e as únicas bibliotecas permitidas são: `STDIO.H` e `STDLIB.H`, caso contrário o trabalho será invalidado;
- Para validação do código será utilizado o compilador **GCC 6** com seguintes parâmetros¹:
`gcc -Wall -Wextra -Werror -Wpedantic`
- Trabalho bem documentado gera carinho! 😊😊

¹Veja: <https://gcc.gnu.org/onlinedocs/gcc-6.4.0/gcc/Warning-Options.html>.

Simulação de agência bancária – Parte I

O Banco Pacote Econômico (BPE) está precisando melhorar a eficiência das suas agências bancárias. Para tal, contratou sua equipe para desenvolver um simulador de agência bancária para que assim possam decidir quantos caixas eles precisarão disponibilizar durante o dia de maneira que suas agências não fiquem superlotadas ou evite funcionários ociosos. A simulação deve receber como entrada a quantidade de caixas, quantidade de clientes bem como as informações de suas operações. Ademais, ao final deve gerar um relatório sobre as movimentações.

Simulação

Para cada simulação, informamos que a agência contém M caixas e recebe N clientes durante um dia. Cada cliente tem os seguintes dados: código de identificação, operação e o valor, onde cada dado tem a seguinte forma:

- Código: Inteiro
- Operação: 1 para saque e 0 para depósito
- Operação: valor inteiro

A entrada de cada cliente é gerenciada por uma fila única e o cliente deve ser alocado em um caixa disponível. Cada cliente realiza apenas uma operação, mas não existe restrição da quantidade de vezes o cliente foi atendido. Cada caixa atende a um cliente por vez e armazena as movimentações de cada cliente que atendeu em uma pilha. Ao finalizar os atendimentos, os dados das movimentações de todos os clientes são armazenados em uma lista linear ordenada em que a chave de ordenação é o código do cliente. O relatório final deve informar a quantidade L de clientes distintos que passaram pela agência durante o dia bem como as informações referente ao código do cliente, quantidade de operações e saldo final.

Entrada

A entrada é constituída por uma única simulação enviada pela entrada padrão. A primeira linha contém dois inteiros M e N separados por um espaço em branco, onde $1 \leq M, N \leq 2^{31}$.

M será a quantidade de caixas inicialmente disponíveis e N a quantidade de clientes envolvidos da simulação. As próximas N linhas deve-se conter três inteiros C , O e V separados por um espaço em branco, onde $1 \leq C \leq 2^{31}$ representa o código do cliente, $B \in \{0, 1\}$ representa a operação do cliente e $1 \leq V \leq 2^{31}$ o valor referente a operação. Ademais, teremos a seguinte estrutura:

| Entrada: |
|-------------------|
| $M \ N$ |
| $C_1 \ O_1 \ V_1$ |
| $C_2 \ O_2 \ V_1$ |
| $C_3 \ O_3 \ V_1$ |
| \vdots |
| $C_N \ O_N \ V_N$ |

Saída

A primeira linha contém um inteiro L que representa a quantidade distintas de clientes informados na entrada da simulação, onde $1 \leq L \leq N$. As próximas L linhas devem conter três inteiros C_i , Q_i e S_i separados por um espaço em branco, onde $1 \leq C_i \leq 2^{31}$ representa o código do cliente tal que $C_i < C_j$ para todo $1 \leq i < j \leq L$, Q_i representa a quantidade de operações realizadas pelo cliente C_i e S_i o valor final referente as operações do cliente C_i . Ademais, teremos a seguinte estrutura:

| Saída: |
|-------------------|
| L |
| $C_1 \ Q_1 \ S_1$ |
| $C_2 \ Q_2 \ S_1$ |
| $C_3 \ Q_3 \ S_1$ |
| \vdots |
| $C_L \ Q_L \ S_L$ |

Restrições

Algumas restrições para a implementação das estruturas de dados do sistema:

1. Para organizar a entrada dos clientes, deve-se utilizar uma fila circular utilizando alocação sequencial com um vetor de tamanho $\lfloor \sqrt{N} \rfloor$;

2. A lista ordenada do relatório deve ser implementada utilizando Alocação duplamente encadeada. Ademais, o apontador para a lista deve continuar de onde foi feita a última modificação (inserção/remoção);
3. Cada caixa deve conter sua pilha implementada utilizando alocação sequencial. Cada vetor associado à pilha deve ter tamanho $\lceil N/M \rceil$;
4. O sistema tem que utilizar o conceito de tipo abstrato de dados (TAD) para realizar as operações em suas estruturas de dados, i.e., cada estrutura de dados a equipe deve implementar separadamente suas operações com seus respectivos arquivos .c e .h.

No arquivo da simulação deve-se implementar as seguintes funções:

1. Função para ler linhas da entrada;
2. Função de inserir um cliente na fila;
3. Função de remover usuário na fila;
4. Função de enviar um usuário para atendimento (não deve-se passar por parâmetro o caixa destino, pois essa gerência será feita nessa função);
5. Função para criar relatório;
6. Função para imprimir relatório no formato exigido.