ADS 2014 - FATEC SP

ALGORITMOS E LÓGICA DE PROGRAMAÇÃO

PROJETO - **SIMULADOR DE FILA BANCÁRIA**

Bruno Alves Munhoz - 13200519

Felipe

Alisson

program SimBanco;

TYPE

Operation = record

time : integer;

ID : string;

end;

Client = record

ID : integer;

op : Operation;

end;

CList = array[0..99]of Client;

cxa = array[1..3] of integer;

operations = array[0..4] of Operation;

VAR

Ops : Operations;

clientFile : text;

caixa : cxa;

clientList : CList;

index, time, rand, clientCount: integer;

totalWait : integer = 0;

clientWait : array[0..99] of Integer;

taxaMin, avgWait : real;

procedure advanceList;

{\*Avança a lista de clientes, removendo o que estiver na primeira posição

e avançando cada um dos outros uma posição à frente.\*}

var temp : integer = 0;

begin

for index := 0 to 99 do

begin

if index = 99 then

clientList[index].ID := 0

else

clientList[index] := clientList[index + 1];

if clientList[index].ID = 0 then break;

end;

end;

procedure writeFile(var f : text; c: Client);

{\*Autoexplicativo. Escreve os dados dos clientes no arquivo 'Clientes.txt'.\*}

begin

assign(f, 'Clientes.txt');

append(f);

writeln(f, '-------------------------Cliente-----------------------------');

writeln(f, '-------------------------------------------------------------');

writeln(f, 'Entrada: ', c.ID);

writeln(f, 'Saida: ', time, ' segundos');

writeln(f, 'Espera total: ', time - c.ID, ' segundos');

Writeln(f, 'Operacao realizada: ', c.op.ID);

writeln();

close(f);

end;

procedure checkCaixa;

{\*Verifica se os caixas estão vazios. Para cada um que estiver,

remove o primeiro cliente da fila e passa o valor em segundos

da transação realizada ao caixa, por fim avançando a fila.\*}

begin

for index := 1 to 3 do

if caixa[index] = 0 then

if clientList[0].ID > 0 then

begin

caixa[index] := clientList[0].op.time;

writeFile(clientFile, clientList[0]);

totalWait := totalWait + (time - clientlist[0].ID);

{\*Realiza um backup do tempo de espera de cada cliente para ser usado no desvio padrão mais tarde\*}

clientWait[index] := totalWait;

advanceList;

end;

end;

function clienteChegou() : integer;

{\*Determina se um cliente chegou (1) ou não (0).\*}

var

pps:real;

rand:integer;

begin

pps := 1/(taxaMin \* 60) \* 100;

rand := random(10000);

if rand <= (pps\*100) then

ClienteChegou := 1

else

ClienteChegou := 0;

end;

procedure tick;

{\*Simula a passagem de um segundo\*}

begin

time := time+1;

if clienteChegou() = 1 then

begin

for index := 0 to 99 do

if clientList[index].ID = 0 then

begin

clientCount := clientCount +1;

clientlist[index].ID := time;

clientList[index].op := ops[random(5)];

break;

end;

end;

checkcaixa;

for index := 1 to 3 do

begin

if caixa[index] > 0 then

caixa[index] := caixa[index] -1;

end;

end;

procedure initOps;

{\*Inicializa o array de operações \*}

begin

ops[0].time := 35;

ops[0].ID := 'Verificacao de Saldo';

ops[1].time := 45;

ops[1].ID := 'Saque';

ops[2].time := 50;

ops[2].ID := 'Impressao de Extrato';

ops[3].time := 60;

ops[3].ID := 'Aplicacao em Renda Fixa';

ops[4].time := 70;

ops[4].ID := 'Transferencia Entre Contas';

end;

procedure dPadrao;

{\*Calcula o desvio padrão \*}

var

media1:real;

variancia: real;

desviomedio1: real;

desviomedio2: real = 0;

begin

{\* calcula a media de espera dos clientes \*}

media1:= totalwait/clientcount;

{\*retorna o backup dos tempos de cada cliente, calcula-se o desvio medio de cada cliente e ja prepara para a realização do calcula da Variancia\*}

for index := 0 to 99 do

Begin

desviomedio1:= sqr(clientWait[index] - media1);

desviomedio2:= desviomedio1+desviomedio2;

end;

{\*termina o calculo da variancia\*}

variancia:= desviomedio2/clientcount;

{\*Escreve no programa o Desvio Padrao\*}

writeln('Desvio Padrao: ', (sqrt(variancia)), ' segundos.');

end;

Begin {\*----MAIN----\*}

Randomize();

{Inicializações de variáveis...}

clientCount := 0; time := 0; initOps;

Writeln('Digite o tempo medio de chegada de clientes, em minutos:');

readln(taxaMin);

while time < 28800 do {Expediente equivalente a 8 horas}

tick;

writeln('Clientes atendidos: ',clientCount);

writeln('Espera total: ', totalWait, ' segundos.');

writeln('Media de espera: ', (totalWait / clientCount), ' segundos.');

dPadrao;

readln();

end.