

ÁREA II – CCEN / UFPE

Disciplina: **IF 165 – Computação Eletrônica** – 2ª chamada 2009.2 – **Gabarito** – Data: 01 / 12 / 09

1º quesito: Tem-se um conjunto de N pontos no plano, definidos por suas coordenadas cartesianas (X e Y). As coordenadas X e Y são armazenadas em 2 vetores. Tem-se também um ponto Q definido pelas coordenadas XQ e YQ. Pede-se fazer um programa em PASCAL que siga os seguintes passos:

- a) Ler um inteiro N ($3 \leq N \leq 90$) correspondente ao número de pontos; Se o usuário não digitar N no interalo, peça novamente até um valor correto ser digitado.
- b) Ler as N coordenadas X e Y do conjunto de pontos, utilizando um procedimento LERVET a ser incluído no programa; **este procedimento deve ler um vetor cada vez que é chamado;**
- c) Ler as coordenadas do ponto Q (XQ e YQ);
- d) Construir um vetor DIST, contendo os valores das distâncias do ponto Q aos N pontos do conjunto de pontos lidos em (b), utilizando um procedimento CALCDIST;
- e) Calcular o valor da média das distâncias calculadas em (d), utilizando uma função MEDIA;
- f) Imprimir as coordenadas dos pontos cujas distâncias ao ponto Q é menor que a distância média, utilizando um procedimento;

Obs.: **O programa principal deve conter os procedimentos e funções descritas abaixo, onde não deve ser utilizado variáveis globais:**

- I. Um procedimento LERVET para ler um vetor de elementos reais;
- II. Um procedimento CALCDIST para construir o vetor DIST descrito em (d);
- III. Uma função, MEDIA para encontrar a média das distâncias como solicitado em (e);
- IV. Um procedimento IMPPT, para imprimir a lista de pontos com distância ao ponto Q menor que a distância média.

PROGRAM PONTOS;

TYPE VET = ARRAY[1 .. 90] OF REAL;

PROCEDURE LERVET(K : INTEGER; VAR V : VET);

VAR CONT : INTEGER;

BRGIN

FOR CONT := 1 TO K DO

READLN(V[CONT]);

END;

PROCEDURE CALCDIST(K : INTEGER; X, Y : VET; QX, QY : REAL; VAR D : VET);

VAR CONT : INTEGER;

BEGIN

FOR CONT := 1 TO K DO

D[CONT] := SQRT(SQR(X[CONT] – QX[CONT] + SQR(Y[CONT] – QY[CONT])));

END;

FUNCTION MEDIA(K : INTEGER; D : VET) : REAL;

VAR CONT : INTEGER;

S : REAL;

BEGIN

S := 0;

FOR CONT := 1 TO K DO

S := S + D[CONT];

MEDIA := S / K;

END;

PROCEDURE IMPPT(K : INTEGER; M : REAL; X, Y, D : VET);

VAR CONT : INTEGER;

BEGIN

WRITELN(‘Coordenadas dos pontos’);

FOR CONT := 1 TO K DO

IF D[CONT] <= M THEN WRITELN(‘ Ponto ‘, I:2, ‘ : ‘, ‘ X=’, X[CONT]:5:1, ‘ Y=’, Y[CONT]:5:1);

END;

END;

VAR X, Y, DIST : VET;

N : INTEGER;

XQ, YQ, MED : REAL;

BEGIN

REPEAT

WRITE(‘ N=? ‘); READLN(N);

UNTIL (N >= 3) AND (N <= 90);

WRITELN(‘ Digite as coordenadas X’); LERVET(N, X);

WRITELN(‘ Digite as coordenadas Y’); LERVET(N, Y);

WRITELN(‘ Digite as coordenadas X e Y do ponto Q ’); READ(XQ, YQ);

CALCDIST(N, X, Y, XQ, YQ, DIST);

MED := MEDIA(N, DIST);

IMPPT(N, MED, X, Y, DIST);

END.

Critério de correção

- Procedimento LERVET 1,0
- Procedimento CALCDIST. .. . 1,0
- Função MEDIA 1,0
- Procedimento IMPPT. 1,0
- Programa principal 1,0

2º quesito.

O arquivo VOTOS.ARQ, já gravado no disco, contem os votos para deputado estadual e deputado federal de uma eleição, conforme descrição de registro abaixo:

Título do Eleitor	Número do Candidato votado	
	Deputado estadual	Deputado federal
String[15]	Inteiro	Inteiro

Obs.: os votos não válidos (nulos ou em branco) são armazenados com o número de candidato zero.

Fazer um programa em PASCAL para:

Ler o arquivo VOTOS.ARQ e gravar dois outros arquivo (DEPEST.ARQ e DEPFED.ARQ) com os votos válidos para deputado estadual e os votos válidos para deputado federal, respectivamente, cujos registros têm a seguinte estrutura:

Título do Eleitor	Número do Candidato
String[15]	Inteiro

```
PROGRAM ELEICAO;
TYPE REG1 = RECORD
    TIT: STRING[15];
    VDE, VDF: INTEGER;
END;
REG2= RECORD
    TIT: STRING[15];
    NV: INTEGER;
END;
VAR VOTO: REG1;
    VOTOS: FILE OF REG1;
    VD: REG2;
    VDE, VDF: FILE OF REG2;
BEGIN
    ASSIGN( VOTOS, 'VOTOS.ARQ' );
    ASSIGN( VDE, 'DEPEST.ARQ' );
    ASSIGN( VDF, 'DEPFED.ARQ' );

    RESET(VOTOS);
    REWRITE(VDE);
    REWRITE(VDF);

    WHILE NOT EOF(VOTOS) DO
    BEGIN
        READ( VOTOS, VOTO);
        VD.TIT := VOTO.TIT;
        IF VOTO.VDE > 0 THEN
        BEGIN
            VD.NV:= VOTO.VDE;
            WRITE(VDE, VD);
        END;
        IF VOTO.VDF > 0 THEN
        BEGIN
            VD.NV:= VOTO.VDF;
            WRITE(VDF, VD);
        END;
    END;
    CLOSE(VOTOS);
    CLOSE(VDE);
    CLOSE(VDF);
END.
```

Critério de correção

- Definição dos arquivos. 1,0
- ASSIGN's e abertura dos arquivos. . . 1,0
- Leitura do arquivo VOTOS.ARQ 1,0
- Passagem dos campos 1,0
- Gravação dos arquivos 1,0

Feliz Natal!