

DET 109 [Programação de Computadores I]
Assunto: Módulos, procedimentos e funções

```
unit UModulos;
interface
const max = 20;
type TVetor = array[1..max] of integer;
function media(n1,n2,n3: real): real;
procedure eq2Grau(A,B,C: real);
function distancia(x1,y1,x2,y2: integer): real;
procedure exhibePares(valorInicial,valorFinal: integer);
function pesquisa(x: integer; vet: TVetor): integer;

implementation
function media(n1,n2,n3: real): real;
var m: real;
begin
  m := (n1+n2+n3)/3;
  media := m;
end;

procedure eq2Grau(A,B,C: real);
var delta,x1,x2: real;
begin
  delta := sqr(B) - 4 * A * C;
  if (delta >= 0) then
    begin
      x1 := (-B + sqrt(delta))/(2*A);
      writeln('x1 = ',x1:5:2);
      x2 := (-B - sqrt(delta))/(2*A);
      writeln('x2 = ',x2:5:2);
    end
  else
    writeln('Nao existem raizes reais!')
  end;
end;

function distancia(x1,y1,x2,y2: integer): real;
var d: real;
begin
  d := sqrt(sqr(x2-x1) + sqr(y2-y1));
  distancia := d;
end;

procedure exhibePares(valorInicial,valorFinal: integer);
var i: integer;
begin
  for i := valorInicial to valorFinal do
    if (i mod 2 = 0) then
      writeln(i:4)
  end;
end;

function pesquisa(x: integer; vet: TVetor): integer;
var
  posicao: integer;
  encontrou: boolean;
begin
  posicao := 1;
  encontrou := false;
  while (posicao < max) and (encontrou = false) do
    if vet[posicao] = x then
      encontrou := true
    else posicao := posicao + 1;
  if (encontrou = false) then
    posicao := -1;
  pesquisa := posicao
end; // pesquisa

end.
```

```

program PModulos;
{$APPTYPE CONSOLE}
uses
  SysUtils,
  UModulos in 'UModulos.pas';
var
  x,y,z,resultado, d: real;
  x1,y1,x2,y2,vInicial,vFinal,op,i,numeroAPesquisar,posicao: integer;
  numeros: TVetor;
begin
  repeat
    writeln('1. Media');    writeln('2. Equacao do 2o. grau');
    writeln('3. Distancia entre dois pontos');
    writeln('4. Exibe pares em um intervalo');
    writeln('5. Pesquisa elemento');
    writeln('6. Palpite');
    writeln('7. Fim');
    write('Digite opcao: ');
    readln(op);
    case op of
      1: begin
          write('Digite 1o valor: '); readln(x);
          write('Digite 2o valor: '); readln(y);
          write('Digite 3o valor: '); readln(z);
          resultado := media(x,y,z);
          writeln('Media = ',resultado:5:2);
        end;
      2: begin
          write('Digite Coeficiente A: '); readln(x);
          write('Digite Coeficiente B: '); readln(y);
          write('Digite Coeficiente C: '); readln(z);
          Eq2Grau(x,y,z);
        end;
      3: begin
          write('Digite valor de X1: '); readln(x1);
          write('Digite valor de Y1: '); readln(y1);
          write('Digite valor de X2: '); readln(x2);
          write('Digite valor de Y2: '); readln(y2);
          d := distancia(x1,y1,x2,y2);
          writeln('Distancia entre ('x1,',',y1,') e ('x2,',',y2,') = ',d:5:2);
        end;
      4: begin
          write('Digite valor inicial do intervalo: '); readln(vInicial);
          write('Digite valor final do intervalo: '); readln(vFinal);
          exibePares(vInicial,vFinal);
        end;
      5: begin
          writeln('Numeros sorteados');
          writeln('-----');
          for i := 1 to max do
            begin
              numeros[i] := random(100)+1;
              write(numeros[i]:3);
            end;
          writeln;
          write('Digite no. a pesquisar: '); readln(numeroAPesquisar);
          posicao := pesquisa(numeroAPesquisar,numeros);
          if (posicao <> -1) then
            writeln('Numero encontrado na posicao ',posicao)
          else
            writeln('Numero nao encontrado!')
          end;
        end;
    end;//case
    readln;
  until (op = 7)
end.

```