

```
{
  projeto numbers - 2024 (fragmento)
  Projeto de estudo de algoritmos
  Esta obra está licenciado sob Licença Creative Commons Atribuição-CompartilhaIgual 4.0
  Internacional.
  Veja os termos desta licença na página https://creativecommons.org/licenses/by-sa/4.0/
  Copyleft 2024-1 - Evandro Guglielmeli
  versão 0.0.0
}
```

```
program numbers;
```

```
const MAXDIG = 5;
      MAXARR = 120;
      ZPREFIX = '0000';
```

```
// declarações prévias da MV do programa
type placa = array [1..MAXDIG] of integer;
```

```
var arrplaca : array [1..MAXARR] of string;
    linha : integer;
```

```
// rotina para embaralhar os dígitos dados pelo operador
procedure mistura(input : placa; index : integer); // parâmetro
  // input tem a série de dígitos da placa
  // index indica a posição que inicia o arranjo
```

```
var output : placa;
    p, i : LongInt;
    keep, c : string;
```

```
begin
  if index=MAXDIG then
    begin
      keep := ''; c := '';
      p := 0;
      for i := 1 to MAXDIG do
        p := p * 10 + input[i];
      str(p,keep);
      keep := ZPREFIX + keep;
      p := length(keep)-(MAXDIG-1);
      c := copy(keep,p,MAXDIG);
      arrplaca[linha] := c;
      linha := linha + 1;
    end
  else
    begin
      output := input;
      for p := index to MAXDIG do
        begin
          output[index] := input[p];
          for i := index to p - 1 do
            output[i + 1] := input[i];
          for i := p + 1 to MAXDIG do
            output[i] := input[i];
          mistura(output,index + 1); // argumento
        end;
    end;
end;
```

```
var ordenadas : array [1..MAXARR] of string;

procedure ordena();

var swap : string;
    indicea, indiceb : LongInt;
    troca : boolean;

begin
    // faz uma cópia do vetor de séries...
    indicea := 1;
    while not(indicea > MAXARR) do
    begin
        ordenadas[indicea] := arrplaca[indicea];
        indicea := indicea + 1;
    end;
    // repete a varredura do vetor para ver se está ok!
    repeat // laço das corridas pelo vetor...
        indicea := 1;
        troca := FALSE;
        while indicea < MAXARR do // laço de uma corrida pelo vetor...
        begin
            indiceb := indicea + 1;
            if ordenadas[indicea] > ordenadas[indiceb] then
            begin
                swap := ordenadas[indiceb];
                ordenadas[indiceb] := ordenadas[indicea];
                ordenadas[indicea] := swap;
                troca := TRUE; // único lugar que muda a condição do controle de trocas
            end;
            indicea := indicea + 1;
        end;
    until troca = FALSE;
end;

// o programa principal
// versão 0.0.0

var entrada : string;
    input : placa;
    v, w, //para uso no procedimento val()
    p, i : integer; //para controle de laços

begin
    // iniciar a iteração do algoritmo
    //writeln( MAXDIG,' ',MAXARR,' ',ZPREFIX);
    //writeln;
    repeat
        write('Entre o número da placa, na forma de uma sequência de ', MAXDIG,' dígitos: ');
        readln(entrada);
        if entrada='fim' then
            break;
        {
            se compr(entrada) = MAXDIG então
            executa o abaixo
        }
        //writeln(length(entrada));
        //writeln;
```

```
if length(entrada) = MAXDIG then
begin
  for i := 1 to MAXDIG do
  begin
    val(entrada[i],v,w);
    if (w=0) then
      input[i] := v
    else
      break;
  end;
  if (w<>0) then
  begin
    writeln;
    writeln('Apenas dígitos de 0 a 9 são aceitos...');
  end
  else
  begin
    linha := 1;
    mistura(input,1); // argumentos
    for p := 1 to MAXARR do
    begin
      //for i := 1 to MAXDIG do
      //write(arrplaca[p,i]:1);
      //writeln;
      write(arrplaca[p],' / ');
    end;
    writeln;
    ordena;
    for p := 1 to MAXARR do
    begin
      write(ordenadas[p],' / ');
    end;
    writeln;
  end;
end;
until FALSE;
end.
```