**Análisis del problema**

|  |  |  |
| --- | --- | --- |
| **Entrada** | **Proceso** | **Salida** |
| Tamaño de la matriz:  N, M.  Sentido de llenado de la matriz:  Sentido. | Pedir los datos de entrada.  Realizar el llenado de la matriz, dependiendo si es en sentido horario o anti horario.  Mostrar los datos de salida | Mostrar la matriz llena. |

**Pseudocodigo**

**Cabecera:** Llenado en espiral;

**Declaraciones**

Matriz: Arreglo [1..N, 1..M] entero;

Sentido: Carácter;

i,j, Numero, LimI\_N, LimiF\_N, LimI\_M, LimF\_N: Entero;

**Cuerpo**

**Inicio**

i <- 0;

j <- 0;

Numero <- 1

Repetir

Escribir (“Ingrese el tamaño deseado de la matriz. Las filas o las columnas no puede ser mayor de 10: ”);

Leer (N, M);

Si (N>10) o (M>10) Entonces

Escribir (“La cantidad de filas o la cantidad de columnas excede el límite. Vuelva a intentarlo.”);

Fin-Si;

Hasta (N<10) Y (M<10);

Fin-RH;

Para i, 1, N, +1 Hacer

Para j, 1, M, +1 Hacer

Matriz [N,M]<- 0;

Fin-RP;

Fin-RP;

Escribir (“Sentido de llenado de su matriz:”);

Escribir (“1. Sentido horario”);

Escribir (“2. Sentido anti horario”);

Leer (Sentido);

En caso de Sentido Hacer

1:

LimI\_N<- 1;

LimF\_N <- N

LimI\_M <- 1;

LimF\_M <- M;

Repetir Mientras (Numero <> N\*M) Hacer

Repetir para i, LimI\_M, LimF\_M, +1 Hacer

Matriz [LimI\_N, i] <- Numero;

Numero<- Numero + 1;

Fin-RP

LimI\_N <- LimI\_N + 1;

Repetir para i, LimI\_N, LimF\_N, +1 Hacer

Matriz [i, LimF\_N] <- Numero;

Numero<- Numero + 1;

Fin-RP;

LimF\_M <- LimF\_M – 1;

Repetir para i, LimF\_M, LimI\_M, -1 Hacer

Matriz [LimF\_N, i] <- Numero;

Numero<- Numero + 1;

Fin-RP;

LimF\_N<- LimF\_N-1;

Repetir para i, LimF\_N, LimI\_N, -1, Hacer

Matriz [i, LimI\_M] <- Numero;

Numero<- Numero + 1;

Fin-RP;

LimI\_M<- LimI\_M +1;

Fin-RM;

Para i, 1, N, +1 Hacer

Para j, 1, M, +1 Hacer

Write (Matriz [i,j], ‘ ‘);

Fin-RP;

Writeln ();

**Fin.**

**Programa fuente**

**program LlenadodeMatriz;**

**Uses**

**Crt;**

**Var**

**Sentido: Char;**

**i,j, Numero, LimI\_N, LimF\_N, LimI\_M, LimF\_M, N, M: integer;**

**Matriz: array [1..10 , 1..10] of integer;**

**begin**

**i := 0;**

**j := 0;**

**Numero := 1;**

**Repeat**

**clrscr;**

**Writeln ('Ingrese el tamanio deseado de la matriz. Las filas o las columnas no puede ser mayor de 10: ');**

**Readln (N, M);**

**If (N>10) or (M>10) then**

**Writeln ('La cantidad de filas o la cantidad de columnas excede el limite. Vuelva a intentarlo.');**

**until (N <= 10) and (M <= 10);**

**For i:= 1 to N Do**

**For j:= 1 to M Do**

**Matriz [N,M]:= 0;**

**Writeln ('Sentido de llenado de su matriz:');**

**Writeln ('1. Sentido horario');**

**Writeln ('2. Sentido anti horario');**

**Readln (Sentido);**

**Case Sentido of**

**'1':**

**Begin**

**LimI\_N:= 1;**

**LimF\_N := N;**

**LimI\_M := 1;**

**LimF\_M := M;**

**While (Numero <> N\*M) Do**

**Begin**

**if (Numero <> N\*M)then**

**begin**

**For i:= LimI\_M to LimF\_M Do**

**Begin**

**Matriz [LimI\_N, i] := Numero;**

**Numero:= Numero + 1;**

**End;**

**end;**

**LimI\_N := LimI\_N + 1;**

**if (Numero <> N\*M)then**

**begin**

**For i:= LimI\_N to LimF\_N Do**

**Begin**

**Matriz [i, LimF\_M] := Numero;**

**Numero:= Numero + 1;**

**end;**

**end;**

**LimF\_M := LimF\_M - 1;**

**if (Numero <> N\*M)then**

**begin**

**For i:= LimF\_M downto LimI\_M Do**

**Begin**

**Matriz [LimF\_N, i] := Numero;**

**Numero:= Numero + 1;**

**end;**

**end;**

**LimF\_N:= LimF\_N-1;**

**if (Numero <> N\*M)then**

**Begin**

**For i:= LimF\_N downto LimI\_N Do**

**Begin**

**Matriz [i, LimI\_M] := Numero;**

**Numero:= Numero + 1;**

**End;**

**end;**

**LimI\_M:= LimI\_M +1;**

**end;**

**clrscr;**

**For i:= 1 to N Do**

**Begin**

**For j:= 1 to M Do**

**Write (Matriz [i,j], ' ');**

**Writeln ();**

**end;**

**end;**

**end;**

**Readkey;**

**end.**