

ESPIRAL DE ULAM EN LENGUAJE ENSAMBLADOR

Marcela Margarita Estrada Rodríguez

Carné: 1010419

GENERACIÓN E IMPRESIÓN DE COORDENADAS Y MATRIZ

Lógica y código

PATRÓN PARA LA GENERACIÓN DE COORDENADAS

100	99	98	97	96	95	94	93	92	91
65	64	63	62	61	60	59	58	57	90
66	37	36	35	34	33	32	31	56	89
67	38	17	16	15	14	13	30	55	88
68	39	18	5	4	3	12	29	54	87
69	40	19	6	1	2	11	28	53	86
70	41	20	7	8	9	10	27	52	85
71	42	21	22	23	24	25	26	51	84
72	43	44	45	46	47	48	49	50	83
73	74	75	76	77	78	79	80	81	82

- Primer caso:
 - Disminuye x hasta llegar a $-x$
 - Y permanece igual
- Segundo caso:
 - Disminuye y hasta llegar a $-y$
 - X permanece igual
- Tercer caso:
 - Aumenta $-x$ hasta llegar a x
 - Y permanece igual
- Cuarto caso
 - Primero
 - X aumenta uno
 - Y permanece igual
 - Segundo
 - Aumenta y hasta llegar a $y = x$
 - X permanece igual

CÓDIGO PARA LOS CASOS

```
;PRIMER CASO
primerCasoResta proc near
    XOR AX,AX
    MOV AL, coordenadaX
    SUB AL, 01h
    DIV diez
    MOV coordenadaX, AH

    CALL restarUno
    CALL agregarMatriz
    CALL imprimirCoordenada
    CALL restaTotal
    JLE llamarFinal

    CMP coordenadaX, 01h
    JL primerCasoSuma
    JMP primerCasoResta
primerCasoResta endp
```

```
primerCasoSuma proc near
    XOR AX,AX
    MOV AL, coordenadaX
    ADD AL, 01h
    DIV diez
    MOV coordenadaX, AH

    CALL restarUno
    CALL agregarMatriz
    CALL imprimirCoordenadaN
    CALL restaTotal
    JLE llamarFinal

    XOR AX, AX
    MOV AL, bandera
    CMP coordenadaX, AL
    JE segundoCasoResta
    JMP primerCasoSuma
primerCasoSuma endp
```

MÉTODO PARA IMPRESIÓN DE COORDENADAS

```
;IMPRIMIR NUMEROS Y PARENTESIS
imprimirCoordenada proc near
    CALL imprimirEnter
    CALL imprimirActual
    ;parentesis
    XOR DX, DX
    MOV DL, 28h
    MOV AH, 02h
    INT 21h

    XOR DX, DX
    MOV DL, coordenadaX
    ADD DL, 30h
    MOV AH, 02h
    INT 21h

    ;Coma
    XOR DX, DX
    MOV DL, 2Ch
    MOV AH, 02h
    INT 21h
```

```
XOR DX, DX
MOV DL, coordenadaY
ADD DL, 30h
MOV AH, 02h
INT 21h

;parentesis
XOR DX, DX
MOV DL, 29h
MOV AH, 02h
INT 21h

ret
imprimirCoordenada endp
```

```
;ENTER
imprimirEnter proc near
    MOV DL, 0Ah
    MOV AH, 02h
    INT 21h
ret
imprimirEnter endp
```

```
imprimirActual proc near
    XOR AX, AX
    MOV AL, contador
    MOV BL, 64h
    DIV BL

    MOV residuoA, AH

    MOV DL, AL
    ADD DL, 30h
    MOV AH, 02h
    INT 21h

    XOR AX, AX
    MOV AL, residuoA
    MOV BL, diez
    DIV BL

    MOV residuoA, AH

    MOV DL, AL
    ADD DL, 30h
    MOV AH, 02h
    INT 21h

    MOV DL, residuoA
    ADD DL, 30h
    MOV AH, 02h
    INT 21h
```

DECLARACIÓN DE LA MATRIZ

```
;Matriz para imprimir
```

```
matriz db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 0Ah  
db 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, 5Fh, '¢'
```

	1	2	3	4	5	6	7	8	9	10
1	100	99	98	97	96	95	94	93	92	91
2	65	64	63	62	61	60	59	58	57	90
3	66	37	36	35	34	33	32	31	56	89
4	67	38	17	16	15	14	13	30	55	88
5	68	39	18	5	4	3	12	29	54	87
6	69	40	19	6	1	2	11	28	53	86
7	70	41	20	7	8	9	10	27	52	85
8	71	42	21	22	23	24	25	26	51	84
9	72	43	44	45	46	47	48	49	50	83
10	73	74	75	76	77	78	79	80	81	82

POSICIONES DENTRO DE LA MATRIZ

	1	2	3	4	5	6	7	8	9	10
1	0	1	2	3	4	5	6	7	8	9
2	11	12	13	14	15	16	17	18	19	20
3	22	23	24	25	26	27	28	29	30	31
4	33	34	35	36	37	38	39	40	41	42
5	44	45	46	47	48	49	50	51	52	53
6	55	56	57	58	59	60	61	62	63	64
7	66	67	68	69	70	71	72	73	74	75
8	77	78	79	80	81	82	83	84	85	86
9	88	89	90	91	92	93	94	95	96	97
10	99	100	101	102	103	104	105	106	107	108

- Aumentar **X**:
 - Se le suma uno a la casilla de la posición.
- Disminuir **X**:
 - Se le resta uno a la casilla de la posición.
- Aumentar **Y**:
 - Se le resta once a la casilla de la posición
- Disminuir **Y**:
 - Se le suma once a la casilla de la posición

MÉTODOS PARA LA ASIGNACIÓN DE VALORES E IMPRESIÓN DE LA MATRIZ

```
sumarUno proc near
    XOR AX,AX
    MOV AH, MatrizC
    ADD AH, 01h
    MOV MatrizC, AH

    XOR AX, AX
    MOV AL, 26
    MOV figura, AL
ret
sumarUno endp
```

```
;Agregar punto a la matriz
agregarMatriz proc near
    XOR AX, AX
    MOV AL, 00h
    ADD AL, MatrizC
    MOV SI, AX
    MOV AL, figura
    MOV matriz[SI], AL
ret
agregarMatriz endp
```

```
;Imprimir la matriz
imprimirMatriz proc near
    MOV DX, offset matriz[0][0]
    MOV AH, 09h
    INT 21h
ret
imprimirMatriz endp
```

	figura	valor
Aumentar X	→	26
Disminuir X	←	27
Aumentar Y	↑	24
Disminuir Y	↓	25

FUNCIONAMIENTO Y VALIDACIONES

Programa ejecutable

INICIO DEL PROGRAMA

Al correr el programa, se le solicita al usuario que ingrese un número entero entre 1 y 100 con el formato ###.

Si el usuario ingresa cualquier número o carácter fuera de este rango, no se imprime ninguna coordenada y la matriz se queda en blanco y lanza el error de número no válido.

```
001
001 > (0,0)

```




```
000
Este numero no es valido

```

```
101
Este numero no es valido

```



MÉTODOS PARA VALIDACIÓN DE NÚMEROS Y MOSTRAR ERROR

```
;VALIDACIÓN DE LOS NÚMEROS
sonNumeros proc near
    CMP AL, 00h
    JL Error
    CMP AL, 09h
    JG Error
ret
sonNumeros endp
```

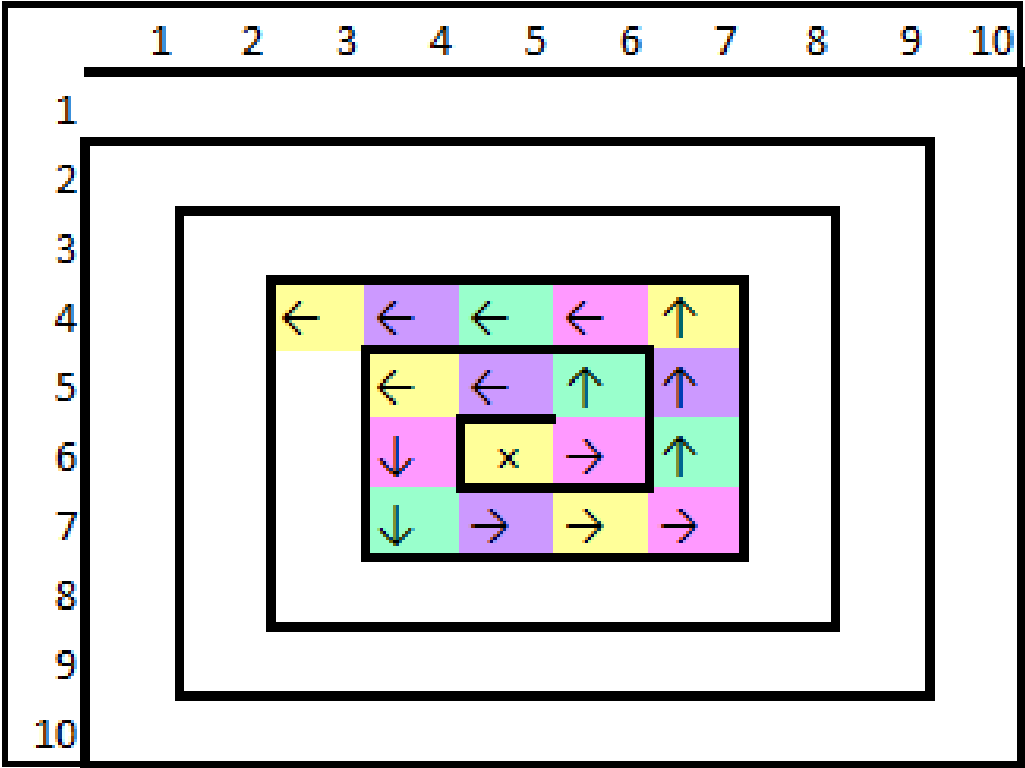
```
;Primero ver si las centenas es 0 o 1
primeraValidacion proc near
    CMP centenas, 01h
    JE validacionCien
    CMP decenas, 01h
    JL validacionUnidades
    JGE llamarFormarNumero
    JMP mostrarError
primeraValidacion endp

;Validación en el caso que las centenas sea 1, las decenas y unidades deben ser 0
validacionCien proc near
    CMP decenas, 01h
    JGE Error
    CMP unidades, 01h
    JGE Error
    JMP formarNumero
validacionCien endp

;Validación en el caso que las centenas y decenas sean 0, unidades tiene que ser mínimo 1
validacionUnidades proc near
    CMP unidades, 01h
    JL Error
    JMP formarNumero
validacionUnidades endp
```

RESULTADO

Cantidad: 17



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

Coordenada plano normal	
x	y
0	0
1	0
1	1
0	1
-1	1
-1	0
-1	-1
0	-1
1	-1
2	-1
2	0
2	1
2	2
1	2
0	2
-1	2
-2	2

PROYECTO 1: Ingrese un numero entre 1 y 100, siguiendo el formato ###

017

001 > <0,0>

002 > <1,0>

003 > <1,1>

004 > <0,1>

005 > <-1,1>

006 > <-1,0>

007 > <-1,-1>

008 > <0,-1>

009 > <1,-1>

010 > <2,-1>

011 > <2,0>

012 > <2,1>

013 > <2,2>

014 > <1,2>

015 > <0,2>

016 > <-1,2>

017 > <-2,2>

←←←←↑

←←↑↑

↓×→↑

↓→→→