

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

# METODOLOGÍA DE LA PROGRAMACIÓN

## - ÁREA DE PROGRAMACIÓN -

---

UNIDAD 3. DISEÑO MODULAR  
PROBLEMARIO 2 : LLAMADO A MÓDULOS CON PASO DE PARÁMETROS



Docente:  
Prof.<sup>a</sup> Guillermina Sánchez Román

Alumno:  
Jesús Huerta Aguilar

Matricula:  
202041509

NRC: 31673  
Sección: 005

**PRIMER SEMESTRE**

Puebla, Pue.

Fecha de entrega: 30/04/2021

## Ejercicios:

Realizar la siguiente lista de ejercicios utilizando el diseño modular. Y para tener derecho al 3er. examen parcial debes de realizar al menos el 70% de la siguiente lista de ejercicios.

1. Hacer un pseudocódigo para crear una matriz de  $5 \times 7$ , donde 5 representa la semana de un mes y 7 los días de la semana. La estructura registrara la temperatura diaria de una cabina De radio, estas oscilan entre los 7 y 38 grados. El pseudocódigo debe llenar la información para el mes de abril (suponemos que tiene 31 días). El algoritmo debe hacer los siguiente.
  - a. Obtener la temperatura más alta y baja de la semana y que día se produjo.
  - b. Obtener la temperatura promedio de la semana.
  - c. Obtener la temperatura más alta del mes y el día en que se produjo.
2. Hacer un pseudocódigo para sumar dos matrices de enteros cuadradas de  $N \times N$  ( $N > 1$ ).
3. Hacer un pseudocódigo para sumar dos matrices de reales rectangulares de  $N \times M$  ( $N, M > 1$ ).
4. Hacer un pseudocódigo para verificar si una matriz es la matriz identidad.
5. Hacer un pseudocódigo para verificar si una matriz cuadrada ( $N \times N$ ) es triangular superior y/o triangular inferior.
6. Hacer un pseudocódigo para obtener la matriz transpuesta de una matriz dada
7. Hacer un pseudocódigo para leer números en una matriz de  $n \times m$ , y almacenar en un arreglo los resultados de sumar los elementos por columna y desplegar resultados, y finalmente realizar la suma de los elementos del arreglo de resultados y presentar resultados.
8. Hacer un pseudocódigo para determinar el número de números primos existentes en la diagonal de una matriz.

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

DIAGRAMA MODULAR [1] (Pt. I):

TEMPERATURA				
VARIABLES GLOBALES:		op,op2,dmnr,dmyr,mnr,myr,recd,recs,rect,ps,ptem,dia,dx,c,d,s		
1	PROCESO PRINCIPAL			
	Subproceso	Subproceso	Subproceso	Subproceso
	instruccioninicial	registrodiario	registrotemp	matriz
VARIABLES LOCALES			d,s,c	s,d
PARAMETROS ENTRADA		c,dia,dx,s,d	recd,recs,rect,dmyr,myr,dmnr,mnr,dia,dx	dia
OPERACIONES	Escribir nota inicial y aviso al usuario.	Dentro de la estructura “Repetir”, leer los días y con un condicional simple invocar al subproceso Error, fuera de la estructura, agregar variable para marcar los días.	Diseñar una estructura “para” para las semanas, dentro de esta, agregar otra estructura “para” para los días, dentro de esta, implementar asignaciones para los días e invocaciones de subprocesos necesarios	Mostrar los días para la matriz, diseñar una estructura “para” para las semanas y dentro de esta, diseñar otra estructura “Para” para los días, dentro de esta, hacer operaciones de escritura para mostrar los valores en forma de matriz.
SALIDA		c	Dia,dx	

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

DIAGRAMA MODULAR [1] (Pt. II):

TEMPERATURA				
VARIABLES GLOBALES:		op,op2,dmnr,dmyr,mnr,myr,recd,recs,rect,ps,ptem,dia,dx,c,d,s		
2	PROCESO PRINCIPAL			
	Subproceso	Subproceso	Subproceso	Subproceso
	definirMayorMenor	mayormenor	promsem	altomes
VARIABLES LOCALES		s,d	s,d,ptem	
PARAMETROS ENTRADA	dmyr,myr,dmnr,mnr,dia,dx,s,d	dmyr,myr,dmnr,mnr	ps,dia	recd,recs,rect
OPERACIONES	Con una la estructura “si”, detectar que valores de Dia son mayores y registrarlos en una variable para compararlos hasta que todos los valores de entrada sean registrados y hacer lo mismo para los casos menores.	Mostrar en pantalla el aviso de la sección indicada, después, dentro de una estructura “Para” diseñada para semanas, mostrar los valores máximos y menores registrados por semana, mostrando la temperatura y el día registrado.	Mostrar en pantalla el aviso de la sección indicada, dentro de una estructura “Para” diseñada para semanas, agregar otra estructura “Para” diseñada para los días, dentro de esta, hacer la suma de los valores de Dia por semana y hacer un promedio de los mismos, el cual, será guardado en una nueva variable de arreglo para después, mostrarla justo al número de semana respectivo.  Mostrar en pantalla el aviso de la sección indicada,	Mostrar en pantalla el aviso de la sección indicada, escribir el día, día de la semana, semana y la temperatura mayor respectivamente.'
SALIDA	dmyr,myr,dmnr,mnr		ps	recd,recs,rect

DIAGRAMA MODULAR [1] (Pt. III):

TEMPERATURA				
VARIABLES GLOBALES:		op,op2,dmnr,dmyr,mnr,myr,recd,recs,rect,ps,ptem,dia,dx,c,d,s		
3	PROCESO PRINCIPAL			
	Subproceso altoTemp	Subproceso menuprincipal	Subproceso error	Subproceso despedida
	VARIABLES LOCALES			
PARAMETROS ENTRADA	rect,recd,recs,dia,dx,s,d	op		
OPERACIONES	Con una estructura “Si” definir la temperatura mayor de todo el mes, guardando la temperatura, el día y la semana en respectivas variables.	Escribir las opciones ofrecidas para el usuario e ingreso de su decisión.	Aviso de error al usuario, solicita que verifique su información.	Mostrar agradecimiento, y firma del autor e institucional.
SALIDA	rect,recd,recs			

1. Hacer un pseudocódigo para crear una matriz de  $5 \times 7$ , donde 5 representa la semana de un mes y 7 los días de la semana. La estructura registrara la temperatura diaria de una cabina De radio, estas oscilan entre los 7 y 38 grados. El pseudocódigo debe llenar la información para el mes de abril (suponemos que tiene 31 días). El algoritmo debe hacer los siguiente.
  - a. Obtener la temperatura más alta y baja de la semana y que día se produjo.
  - b. Obtener la temperatura promedio de la semana.
  - c. Obtener la temperatura más alta del mes y el día en que se produjo

PSEUDOCÓDIGO:

```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Instrucción inicial y aviso
5 SubProceso instruccioninicial
6     Escribir '--REGISTRO DE TEMPERATURAS DURANTE EL MES DE ABRIL--'
7     Escribir ''
8     Escribir 'NOTA: Suponemos que el primer dia del mes es lunes.'
9     Escribir ''
10 FinSubProceso
11
12 //Registro diario
13 SubProceso registrodiario(c Por Referencia,dia Por Referencia,dx,s,d)
14     Repetir
15         Escribir 'Dia ',c,' | ',dx,':'
16         leer dia[s,d]
17         //Aviso por errores fuera del rango
18         Si dia[s,d] > 38 o dia[s,d] < 7 Entonces
19             error
20         FinSi
21         Hasta Que dia[s,d] ≥ 7 Y dia[s,d] ≤ 38
22         c ← c + 1
23 FinSubProceso
24
25 //Registro de temperaturas
26 SubProceso registrotemp(recd Por Referencia,recs Por Referencia,rect Por
Referencia,dmyr,myr,dmnr,mnr,dia,dx)
27     Definir c,s,d Como Entero
28     c ← 1
29     rect ← 0
30     //Cambio de semanas
31     Para s ← 1 Hasta 5 Con Paso 1 Hacer
32         mnr[s] ← 38
```

```

33      Escribir '||||||| SEMANA ',s,' |||||'
34      Para d ← 1 Hasta 7 Con Paso 1 Hacer
35          Si s = 5 y d = 4 Entonces
36              dia[5,4] ← 0
37              dia[5,5] ← 0
38              dia[5,6] ← 0
39              dia[5,7] ← 0
40              d ← 7
41      SiNo
42          //Asignación de los dias
43      Segun d Hacer
44          1:   dx ← 'Lunes'
45          2:   dx ← 'Martes'
46          3:   dx ← 'Miercoles'
47          4:   dx ← 'Jueves'
48          5:   dx ← 'Viernes'
49          6:   dx ← 'Sabado'
50          7:   dx ← 'Domingo'
51      Fin Segun
52      registrodiario(c,dia,dx,s,d)
53
54      altoTemp(rect,recd,recs,dia,dx,s,d)
55
56      definirMayorMenor(dmyr,myr,dmnr,mnr,dia,dx,s,d)
57      FinSi
58      Fin Para
59      Fin Para
60  FinSubProceso
61
62  //Creación de la matriz
63  SubProceso matriz(dia)
64      definir s,d Como Entero
65      Escribir ''
66      Escribir '      L      M      M      J      V      S      D'
67      para s←1 hasta 5 Con Paso 1 Hacer
68          Escribir 'S',s,' ' Sin Saltar
69          para d←1 hasta 7 con paso 1 Hacer
70              Si dia[s,d] < 10 Entonces
71                  Escribir ' | ',dia[s,d] Sin Saltar
72              SiNo
73                  Escribir ' | ', dia[s,d] Sin Saltar

```

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

```
74         FinSi
75     FinPara
76     Escribir " | "
77     FinPara
78     Escribir ' '
79 FinSubProceso
80
81 //Definir temperaturas mayores y menores
82 SubProceso definirMayorMenor(dmyr Por Referencia,myr Por Referencia,dmnr
83 Por Referencia,mnr Por Referencia,dia,dx,s,d)
84     Si dia[s,d] > myr[s] Entonces
85         myr[s] ← dia[s,d]
86         dmyr[s] ← dx
87     SiNo
88         Si dia[s,d] ≤ mnr[s] Entonces
89             mnr[s] ← dia[s,d]
90             dmnr[s] ← dx
91     FinSi
92 FinSubProceso
93
94 //1) Mayor y Menor semanal
95 SubProceso mayormenor(dmyr,myr,dmnr,mnr)
96     Definir s,d Como Entero
97     Escribir "-- TEMPERATURA MAS ALTA Y BAJA DE CADA SEMANA --"
98     Para s ← 1 hasta 5 Con Paso 1 Hacer
99         Escribir ' '
100        Escribir '> Semana ',s
101        Si myr[s] < 10 Entonces
102            Escribir 'Maxima: ',myr[s],'° | Registrada el dia: ',dmyr[s]
103        SiNo
104            Escribir 'Maxima: ',myr[s],'° | Registrada el dia: ',dmyr[s]
105        FinSi
106        Si mnr[s] < 10 Entonces
107            Escribir 'Minima: ',mnr[s],'° | Registrada el dia: ',dmnr[s]
108        SiNo
109            Escribir 'Minima: ',mnr[s],'° | Registrada el dia: ',dmnr[s]
110        FinSi
111    FinPara
112 FinSubProceso
113
```



Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

```
114 //2) Promedio semanal
115 SubProceso promsem(ps Por Referencia,dia)
116     Definir s,d Como Entero
117     Definir ptem Como Real
118     ptem ← 0
119     Escribir "-- TEMPERATURA PROMEDIO DE CADA SEMANA --"
120     Escribir ""
121     para s ← 1 hasta 5 con paso 1 Hacer
122         para d ← 1 hasta 7 Con Paso 1 Hacer
123             ptem ← ptem + dia[s,d]
124         FinPara
125         ps[s] ← ptem/7
126         ptem←0
127         Escribir "Semana ",s,': ', redon(ps[s]*100)/100,'°'
128     FinPara
129 FinSubProceso
130
131 //3) Temperatura mas alta del mes y dia especifico
132 SubProceso altomes(recd,recs,rect)
133     Escribir '--TEMPERATURA MAS ALTA DEL MES Y DIA ESPECIFICO--'
134     Escribir ''
135     Escribir 'El dia ',recd,' de la semana ',recs,' se registro la
temperatura de ',rect,'°.'
136 FinSubProceso
137
138 //temperatura mas alta del mes
139 SubProceso altoTemp(rect Por Referencia,recd Por Referencia,recs Por
Referencia,dia,dx,s,d)
140     Si dia[s,d] > rect Entonces
141         rect ← dia[s,d]
142         recd ← dx
143         recs ← s
144     FinSi
145 FinSubProceso
146
147 //MenuPrincipal
148 SubProceso menuprincipal(op Por Referencia)
149     Escribir '¿Que desea hacer?'
150     Escribir '1- Obtener la temperatura mas alta y baja de cada semana y el
dia en que se produjo'
151     Escribir '2- Obtener la temperatura promedio de cada semana'
```

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

```
152     Escribir '3- Obtener la temperatura mas alta del mes y el dia que se
153     produjo'
154     Leer op
155 FinSubProceso
156 //Error
157 SubProceso error
158     Escribir '(!) ERROR: VERIFICA TU INFORMACIÓN (!)'
159     Escribir ''
160 FinSubProceso
161
162 //Despedida
163 SubProceso despedida
164     Escribir ''
165     Escribir 'Gracias por usar nuestros servicios.'
166     Escribir "Jesús Huerta Aguilar      BUAP - FCC"
167     Escribir ""
168 FinSubProceso
169
170 //PROCESO PRINCIPAL
171 Proceso temperatura
172     //definir variables
173     Definir op2,dmnr,dmyr,dx,recd Como Caracter
174     Definir op Como Entero
175     Definir ps,dia,mnr,myr,recs,rect Como Real
176     Dimension dia[5,7]
177     Dimension ps[5]
178     Dimension myr[5]
179     Dimension mnr[5]
180     Dimension dmyr[5]
181     Dimension dmnr[5]
182     //Instrucciones y panel de registro de temperaturas
183     instruccioninicial
184     registrotmp(recd,recs,rect,dmyr,myr,dmnr,mnr,dia,dx)
185     matriz(dia)
186     //menuprincipal y elección
187     Repetir
188         Repetir
189             menuprincipal(op)
190             Limpiar Pantalla
191             Segun op Hacer
```

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

192	1: mayormenor(dmyr,myr,dmnr,mnr)
193	2: promsem(ps,dia)
194	3: altomes(recd,recs,rect)
195	De Otro Modo:
196	error
197	Fin Segun
198	Escribir ''
199	Escribir '¿Desea obtener algo mas? (S/N) '
200	Leer op2
201	Limpiar Pantalla
202	Hasta Que op > 0 y op ≤ 3
203	Hasta Que op2 ≠ 's' y op2 ≠ 'S'
204	despedida
205	FinProceso

## EJECUCIÓN:

```
PSelnt - Ejecutando proceso TEMPERATURA
*** Ejecución Iniciada. ***
--REGISTRO DE TEMPERATURAS DURANTE EL MES DE ABRIL--

NOTA: Suponemos que el primer día del mes es lunes.

||||| SEMANA 1 |||||
Día 1 | Lunes:
> 12
Día 2 | Martes:
> 13
Día 3 | Miércoles:
> 15
Día 4 | Jueves:
> 16
Día 5 | Viernes:
> 14
Día 6 | Sábado:
> 13
Día 7 | Domingo:
> 2
[!] ERROR: VERIFICA TU INFORMACIÓN [!]

Día 7 | Domingo:
> 14
||||| SEMANA 2 |||||
Día 8 | Lunes:
> 15
Día 9 | Martes:
> 16
Día 10 | Miércoles:
> 14
Día 11 | Jueves:
> 15
Día 12 | Viernes:
> 16
Día 13 | Sábado:
> 17
Día 14 | Domingo:
> 19
||||| SEMANA 3 |||||
Día 15 | Lunes:
> 20
Día 16 | Martes:
> 21
Día 17 | Miércoles:
> 23
Día 18 | Jueves:
> 22
```

```
PSelnt - Ejecutando proceso TEMPERATURA
> 20
Día 16 | Martes:
> 21
Día 17 | Miércoles:
> 23
Día 18 | Jueves:
> 22
Día 19 | Viernes:
> 23
Día 20 | Sábado:
> 21
Día 21 | Domingo:
> 23
||||| SEMANA 4 |||||
Día 22 | Lunes:
> 24
Día 23 | Martes:
> 25
Día 24 | Miércoles:
> 26
Día 25 | Jueves:
> 25
Día 26 | Viernes:
> 26
Día 27 | Sábado:
> 27
Día 28 | Domingo:
> 28
||||| SEMANA 5 |||||
Día 29 | Lunes:
> 29
Día 30 | Martes:
> 28
Día 31 | Miércoles:
> 28

    L  M   M   J   V   S   D
S1 | 12 | 13 | 15 | 16 | 14 | 13 | 14 |
S2 | 15 | 16 | 14 | 15 | 16 | 17 | 19 |
S3 | 20 | 21 | 23 | 22 | 23 | 21 | 23 |
S4 | 24 | 25 | 26 | 25 | 26 | 27 | 28 |
S5 | 29 | 28 | 28 | 0 | 0 | 0 | 0 |

¿Que desea hacer?
1- Obtener la temperatura mas alta y baja de cada semana y
el día en que se produjo
2- Obtener la temperatura promedio de cada semana
3- Obtener la temperatura mas alta del mes y el día que se
produjo
> 1 |
```

```
PSelnt - Ejecutando proceso TEMPERATURA
-- TEMPERATURA MAS ALTA Y BAJA DE CADA SEMANA --
línea 200 instrucción 1

> Semana 1
Maxima: 16° | Registrada el dia: Jueves
Minima: 13° | Registrada el dia: Sabado

> Semana 2
Maxima: 19° | Registrada el dia: Domingo
Minima: 14° | Registrada el dia: Miercoles

> Semana 3
Maxima: 23° | Registrada el dia: Miercoles
Minima: 21° | Registrada el dia: Viernes

> Semana 4
Maxima: 28° | Registrada el dia: Domingo
Minima: 25° | Registrada el dia: Jueves

> Semana 5
Maxima: 29° | Registrada el dia: Lunes
Minima: 28° | Registrada el dia: Miercoles

¿Desea obtener algo mas? (S/N)
> s
```

```
PSelnt - Ejecutando proceso TEMPERATURA

¿Que desea hacer?
1- Obtener la temperatura mas alta y baja de cada semana y el dia en que se produjo
2- Obtener la temperatura promedio de cada semana
3- Obtener la temperatura mas alta del mes y el dia que se produjo
> 2 |
```

```
PSelnt - Ejecutando proceso TEMPERATURA
-- TEMPERATURA PROMEDIO DE CADA SEMANA --

Semana 1: 13.86°
Semana 2: 16°
Semana 3: 21.86°
Semana 4: 25.86°
Semana 5: 12.14°

¿Desea obtener algo mas? (S/N)
> s |
```

```
PSelnt - Ejecutando proceso TEMPERATURA
¿Que desea hacer?
1- Obtener la temperatura mas alta y baja de cada semana y el dia en que se produjo
2- Obtener la temperatura promedio de cada semana
3- Obtener la temperatura mas alta del mes y el dia que se produjo
> 3 |
```

```
PSelnt - Ejecutando proceso TEMPERATURA
--TEMPERATURA MAS ALTA DEL MES Y DIA ESPECIFICO--

El dia Lunes de la semana 5 se registro la temperatura de 29°.

¿Desea obtener algo mas? (S/N)
> n |
```

```
PSelnt - Ejecutando proceso TEMPERATURA

Gracias por usar nuestros servicios.
Jesús Huerta Aguilar      BUAP - FCC

*** Ejecución Finalizada. ***
```

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

DIAGRAMA MODULAR [2] (Pt. I):

MATRICES CUADRADAS				
VARIABLES GLOBALES:		val,n,res,reclongval,textval		
1	PROCESO PRINCIPAL			
	Subproceso	Subproceso	Subproceso	Subproceso
	tamMatriz(n)	registroMatriz	crearMatriz	Espacios
VARIABLES LOCALES			r,c,i,textvalf,l	e,switch,iz,dr,x
PARAMETROS ENTRADA	n	val,reclongval,n	z,val,reclongval,n	textvalf,reclongval,z,res,val,r,c,i
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para las matrices A y B, dentro de la estructura, diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Mostrar en pantalla la sección de resultados, Diseñar una estructura “Para” para las matrices A y B, dentro de la estructura, diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, invocar al subproceso designado para la ayuda de impresión de las matrices e imprimir los valores.	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.
SALIDA	n .	val, reclongval	z	textvalf

DIAGRAMA MODULAR [2] (Pt. II):

MATRICES CUADRADAS			
VARIABLES GLOBALES:		val,n,res,reclongval,txtval	
2	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	sumarMatriz	crearMatrizAB	error
VARIABLES LOCALES	r,cswitch,longval	r,c,txtvalf	
PARAMETROS ENTRADA	res,reclongval,n,val	val,reclongval,n	
OPERACIONES	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, sumar los valores respectivamente de la matriz A y B. para guardarlas en una variable nueva de arreglo, además de determinar las longitudes mayores de cada valor.	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Aviso de error al usuario, solicita que verifique su información.
SALIDA	res,reclongval	z	



2. Hacer un pseudocódigo para sumar dos matrices de enteros cuadradas de NxN ( $N > 1$ ).

PSEUDOCÓDIGO:

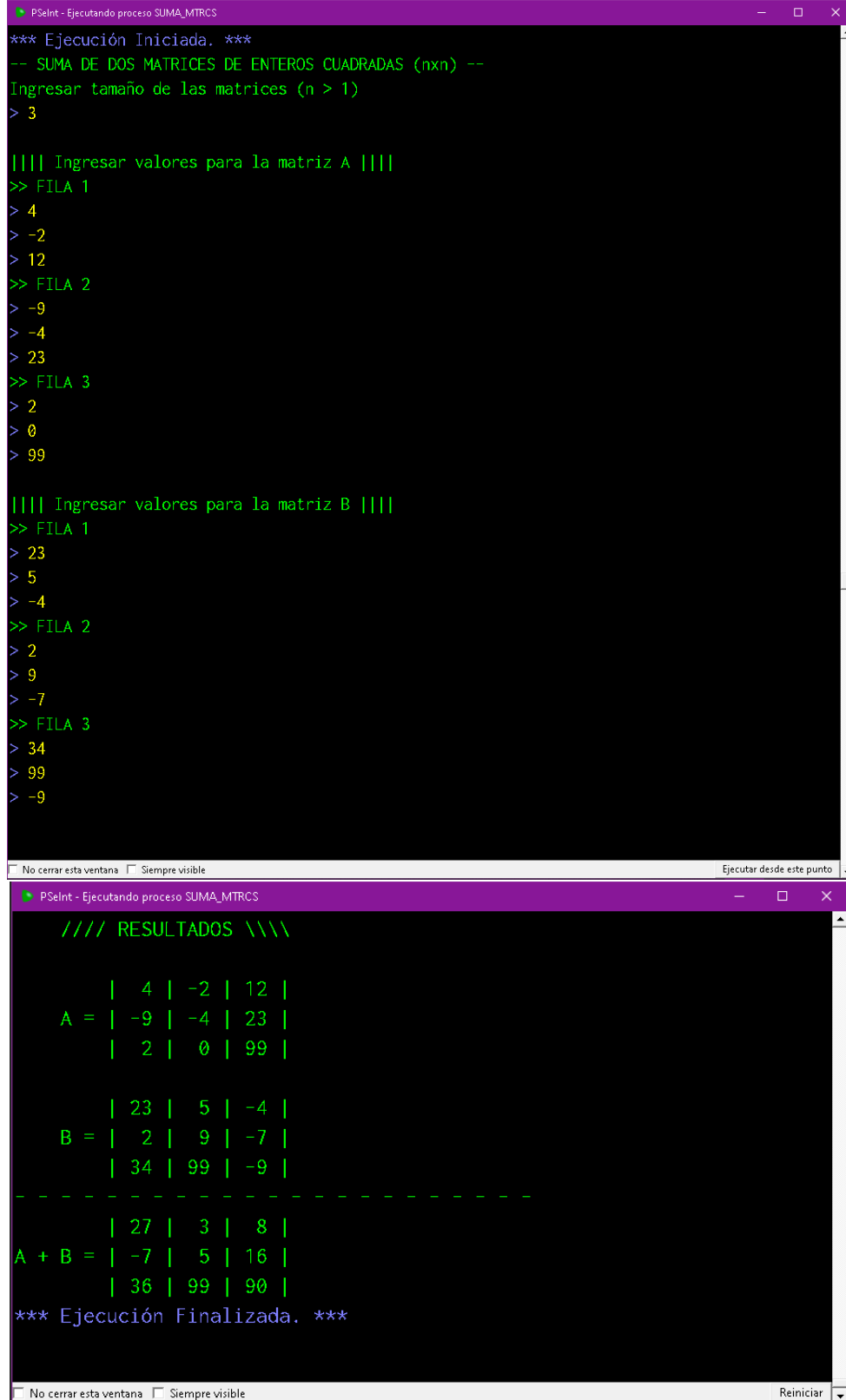
```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de las matrices
5 SubProceso tamMatriz(n Por Referencia)
6     Escribir '-- SUMA DE DOS MATRICES DE ENTEROS CUADRADAS (nxn) --'
7     Escribir 'Ingresar tamaño de las matrices (n > 1)'
8     Repetir
9         Leer n
10        Si n ≤ 1 Entonces
11            error
12        FinSi
13    Hasta Que n > 1
14    Escribir ''
15 FinSubProceso
16
17 //Ingreso de valores para cada matriz
18 SubProceso registroMatriz(val Por Referencia, reclangval Por Referencia, n)
19     Definir r, c, i, logval Como Entero
20     reclangval ← 0
21     Escribir '|||| Ingresar valores para la matriz A ||||'
22     para i ← 1 hasta 2 con paso 1 Hacer
23         Si i = 2 Entonces
24             Escribir '|||| Ingresar valores para la matriz B ||||'
25         FinSi
26     para r ← 1 hasta n Con Paso 1 Hacer
27         Escribir '>> FILA ', r
28         para c ← 1 hasta n con paso 1 Hacer
29             leer val[r, c, i]
30             //longitud del valor
31             longval ← Longitud(ConvertirATexto(val[r, c, i]))
32             si longval > reclangval Entonces
33                 reclangval ← longval
34         FinSi
35     FinPara
36 FinPara
37 Escribir ''
38 FinPara
```

```
39 FinSubProceso
40
41 //Creacion de matriz A y B
42 SubProceso crearMatriz(z Por Referencia, val, reclangval, n)
43     Definir r, c, i Como Entero
44     definir textvalf Como caracter
45     Escribir '      /// RESULTADOS \\\\'
46     para i ← 1 hasta 2 con paso 1 Hacer
47         Escribir ''
48         Segun i
49             1:
50                 l ← 'A'
51             2:
52                 l ← 'B'
53         FinSegun
54         para r ← 1 hasta n con paso 1 Hacer
55             si r = redon(n/2) Entonces
56                 Escribir '      ', l, ' = ' Sin Saltar
57             SiNo
58                 Escribir '      ' Sin Saltar
59             FinSi
60             para c ← 1 hasta n con paso 1 Hacer
61                 z ← 0
62                 espacios(textvalf, reclangval, z, res, val, r, c, i)
63                 Escribir '|', textvalf Sin Saltar
64             FinPara
65             Escribir "|"
66         FinPara
67     FinPara
68     Escribir '-----'
69 FinSubProceso
70
71 //concatenación de espacios
72 SubProceso espacios(textvalf Por Referencia, reclangval, z, res, val, r, c, i)
73     Definir e, switch Como Entero
74     Definir iz, dr, x Como Caracter
75     iz ← ' '; dr ← ' '; x ← ' '
76     switch ← 0
77     para e ← 1 hasta reclangval+2 con paso 1 Hacer
78         Si z = 0 Entonces
79             textval ← ConvertirATexto(val[r, c, i])
```

80	<b>SiNo</b>
81	textval ← ConvertirATexto(res[r,c])
82	<b>FinSi</b>
83	textvalf ← Concatenar(Concatenar(iz,textval),dr)
84	e ← Longitud(textvalf)
85	<b>si</b> e < reclongval+2 <b>y</b> switch = 0 <b>Entonces</b>
86	iz ← Concatenar(iz,x)
87	switch ← 1
88	<b>SiNo</b>
89	dr ← Concatenar(x,dr)
90	switch ← 0
91	<b>FinSi</b>
92	<b>FinPara</b>
93	<b>FinSubProceso</b>
94	
95	<i>//Sumar matrices A + B</i>
96	<b>SubProceso</b> sumarMatriz(res <b>Por Referencia</b> ,reclongval <b>Por Referencia</b> ,n,val)
97	<b>Definir</b> r,c,switch,longval <b>Como Entero</b>
98	switch ← 0
99	reclongval ← 0
100	<b>para</b> r ← 1 <b>hasta</b> n <b>con paso</b> 1 <b>Hacer</b>
101	<b>para</b> c ← 1 <b>hasta</b> n <b>con paso</b> 1 <b>Hacer</b>
102	res[r,c] ← val[r,c,1] + val[r,c,2]
103	<i>//longitud del valor</i>
104	longval ← Longitud(ConvertirATexto(res[r,c]))
105	<b>si</b> longval > reclongval <b>Entonces</b>
106	reclongval ← longval
107	<b>FinSi</b>
108	<b>FinPara</b>
109	<b>FinPara</b>
110	<b>FinSubProceso</b>
111	
112	<i>//Crear matriz A+B</i>
113	<b>SubProceso</b> crearMatrizAB(z <b>Por Referencia</b> ,res,reclongval,n)
114	<b>definir</b> r,c <b>Como Entero</b>
115	<b>Definir</b> textvalf <b>Como Caracter</b>
116	<b>para</b> r ← 1 <b>hasta</b> n <b>Con Paso</b> 1 <b>Hacer</b>
117	<b>si</b> r = redon(n/2) <b>Entonces</b>
118	<b>Escribir</b> 'A + B = ' <b>Sin Saltar</b>
119	<b>SiNo</b>
120	<b>Escribir</b> '                ' <b>Sin Saltar</b>

```
121      FinSi
122      para c ← 1 hasta n con paso 1 Hacer
123          z ← 1
124          espacios(textvalf,reclongval,z,res,val,r,c,i)
125          Escribir '|',textvalf Sin Saltar
126      FinPara
127      Escribir "|"
128  FinPara
129 FinSubProceso
130
131 //Error
132 SubProceso error
133     Escribir '[] ERROR: VERIFICA TU INFORMACIÓN []'
134     Escribir ''
135 FinSubProceso
136
137 //PROCESO PRINCIPAL
138 Proceso matrices_cuadradas
139     Definir val,n,res,reclongval Como Entero
140     Definir textval Como Caracter
141     tamMatriz(n)
142     Dimension val[n,n,2]; Dimension res[n,n]
143     registroMatriz(val,reclongval,n)
144     crearMatriz(z,val,reclongval,n)
145     sumarMatriz(res,reclongval,n,val)
146     crearMatrizAB(z,res,reclongval,n)
147 FinProceso
```

### EJECUCIÓN:



```
PSelnt - Ejecutando proceso SUMA_MTRCS
*** Ejecución Iniciada. ***
-- SUMA DE DOS MATRICES DE ENTEROS CUADRADAS (nxn) --
Ingresar tamaño de las matrices (n > 1)
> 3

[[[ Ingresar valores para la matriz A ]]]
>> FILA 1
> 4
> -2
> 12
>> FILA 2
> -9
> -4
> 23
>> FILA 3
> 2
> 0
> 99

[[[ Ingresar valores para la matriz B ]]]
>> FILA 1
> 23
> 5
> -4
>> FILA 2
> 2
> 9
> -7
>> FILA 3
> 34
> 99
> -9

//// RESULTADOS \\\

      | 4 | -2 | 12 |
A =   | -9 | -4 | 23 |
      | 2 | 0 | 99 |

      | 23 | 5 | -4 |
B =   | 2 | 9 | -7 |
      | 34 | 99 | -9 |

-----
      | 27 | 3 | 8 |
A + B = | -7 | 5 | 16 |
      | 36 | 99 | 90 |

*** Ejecución Finalizada. ***
```

DIAGRAMA MODULAR [3] (Pt. I):

SUMA MATRICES RECTANGULARES				
VARIABLES GLOBALES:		val,n,m,res,reclongval,textval		
1	PROCESO PRINCIPAL			
	Subproceso	Subproceso	Subproceso	Subproceso
	tamMatriz	registroMatriz	crearMatriz	Espacios
VARIABLES LOCALES		r,c,i,longval	r,c,i,textvalf,l	e,switch,iz,dr,x
PARAMETROS ENTRADA	n,m	val,reclongval,n,m	z,val,reclongval,n,m	textvalf,reclongval,z,res,val,r,c,i
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para las matrices A y B, dentro de la estructura, diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Mostrar en pantalla la sección de resultados, Diseñar una estructura “Para” para las matrices A y B, dentro de la estructura, diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, invocar al subproceso designado para la ayuda de impresión de las matrices e imprimir los valores.	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.
SALIDA	n,m	val, reclongval	z	textvalf

DIAGRAMA MODULAR [3] (Pt. II):

SUMA MATRICES RECTANGULARES			
VARIABLES GLOBALES:		val,n,m,res,reclongval,textval	
2	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	sumarMatriz	crearMatrizAB	error
VARIABLES LOCALES	r,cswitch,longval	r,c,textvalf	
PARAMETROS ENTRADA	res,reclongval,n,m,val	val,reclongval,n,m	
OPERACIONES	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, sumar los valores respectivamente de la matriz A y B. para guardarlas en una variable nueva de arreglo, además de determinar las longitudes mayores de cada valor.	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Aviso de error al usuario, solicita que verifique su información.
SALIDA	res,reclongval	z	

3. Hacer un pseudocódigo para sumar dos matrices de reales rectangulares de NxM (N,M>1).

PSEUDOCÓDIGO:

```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de las matrices
5 SubProceso tamMatriz(n Por Referencia,m Por Referencia)
6     Escribir '-- SUMA DE DOS MATRICES RECTANGULARES DE REALES (nxm) --'
7     Escribir 'Ingresar tamaño de las matrices (n,m > 1)'
8     Repetir
9         Escribir ''
10        Escribir 'Cantidad de filas (n)'
11        Leer n
12        Escribir 'Cantidad de columnas (m)'
13        Leer m
14        Si n ≤ 1 o m ≤ 1 Entonces
15            error
16        FinSi
17        Hasta Que n > 1 y m > 1
18        Escribir ''
19 FinSubProceso
20
21 //Ingreso de valores para cada matriz
22 SubProceso registroMatriz(val Por Referencia,reclongval Por
Referencia,n,m)
23     Definir r,c,i,longval Como Entero
24     Escribir '|||| Ingresar valores para la matriz A ||||'
25     para i ← 1 hasta 2 con paso 1 Hacer
26         Si i = 2 Entonces
27             Escribir '|||| Ingresar valores para la matriz B ||||'
28         FinSi
29     para r ← 1 hasta n Con Paso 1 Hacer
30         Escribir '>> FILA ',r
31         para c ← 1 hasta m con paso 1 Hacer
32             leer val[r,c,i]
33             //longitud del valor
34             longval ← Longitud(ConvertirATexto(val[r,c,i]))
35             si longval > reclongval Entonces
36                 reclongval ← longval
37         FinSi
```



```

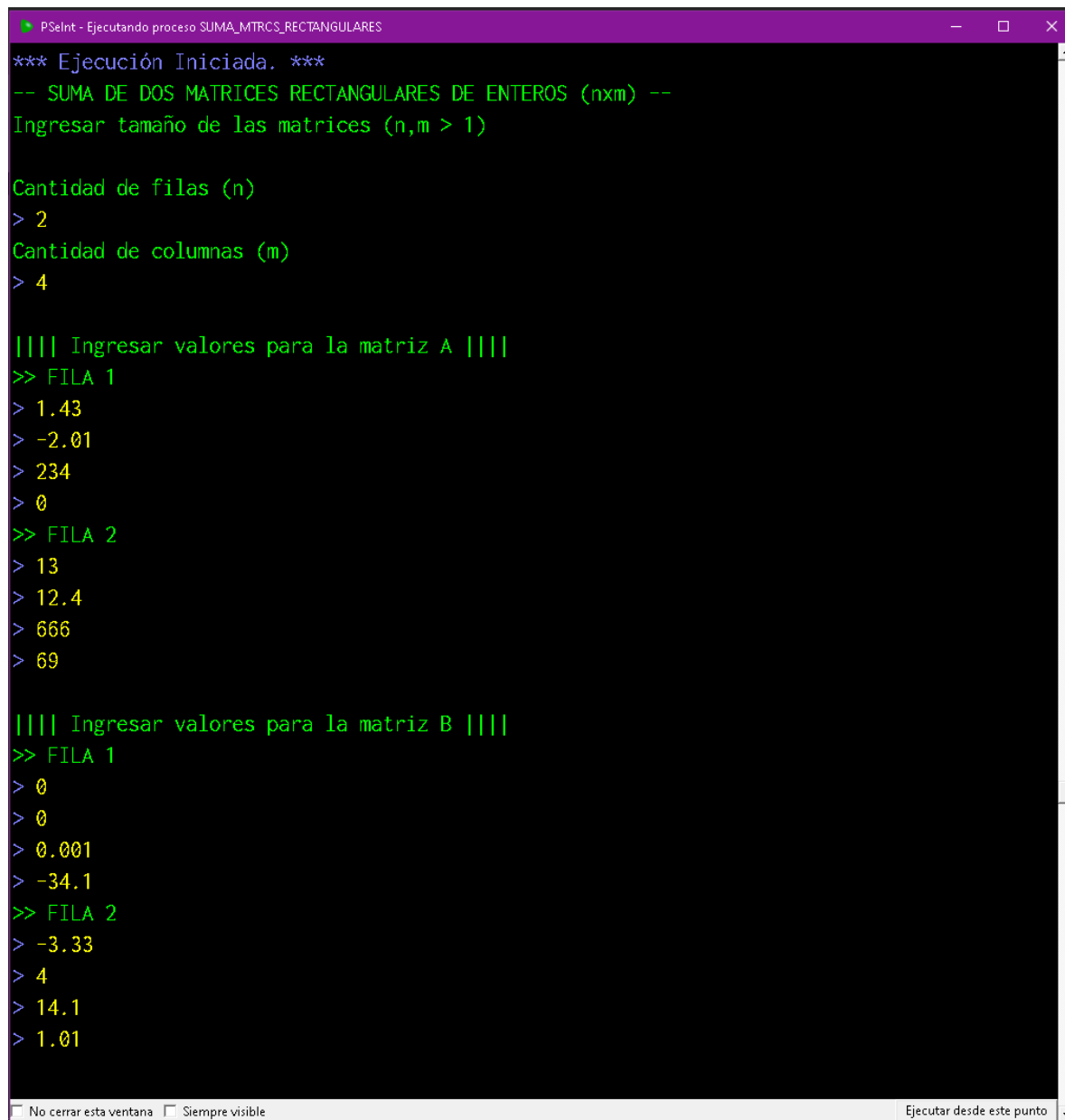
38         FinPara
39     FinPara
40     Escribir ''
41     FinPara
42 FinSubProceso
43
44 //Creacion de matriz A y B
45 SubProceso crearMatriz(z Por Referencia, val, reclangval, n, m)
46     Definir r, c, i Como Entero
47     definir textvalf, l Como caracter
48     Escribir '          \\\\ RESULTADOS \\\\ '
49     para i ← 1 hasta 2 con paso 1 Hacer
50         Escribir ''
51         Segun i
52             1:
53                 l ← 'A'
54             2:
55                 l ← 'B'
56         FinSegun
57         para r ← 1 hasta n con paso 1 Hacer
58             si r = redon(n/2) Entonces
59                 Escribir '          ', l, ' = ' Sin Saltar
60             SiNo
61                 Escribir '          ' Sin Saltar
62             FinSi
63             para c ← 1 hasta m con paso 1 Hacer
64                 z ← 0
65                 espacios(textvalf, reclangval, z, res, val, r, c, i)
66                 Escribir '|', textvalf Sin Saltar
67             FinPara
68             Escribir "|"
69         FinPara
70     FinPara
71     Escribir '-----'
72 FinSubProceso
73
74 //concatenación de espacios
75 SubProceso espacios(textvalf Por Referencia, reclangval, z, res, val, r, c, i)
76     Definir e, switch Como Entero
77     Definir iz, dr, x Como Caracter
78     iz ← ' '; dr ← ' '; x ← ' '

```

```
79      switch ← 0
80      para e ← 1 hasta reclangval+2 con paso 1 Hacer
81          Si z = 0 Entonces
82              textval ← ConvertirATexto(val[r,c,i])
83          SiNo
84              textval ← ConvertirATexto(res[r,c])
85          FinSi
86          textvalf ← Concatenar(Concatenar(iz,textval),dr)
87          e ← Longitud(textvalf)
88          si e < reclangval+2 y switch = 0 Entonces
89              iz ← Concatenar(iz,x)
90              switch ← 1
91          SiNo
92              dr ← Concatenar(x,dr)
93              switch ← 0
94          FinSi
95      FinPara
96  FinSubProceso
97
98  //Sumar matrices A + B
99  SubProceso sumarMatriz(res Por Referencia,reclangval Por
Referencia,n,m,val)
100      Definir r,c,switch,longval Como Entero
101      para r ← 1 hasta n con paso 1 Hacer
102          para c ← 1 hasta m con paso 1 Hacer
103              res[r,c] ← val[r,c,1] + val[r,c,2]
104              //longitud del valor
105              longval ← Longitud(ConvertirATexto(res[r,c]))
106              si longval > reclangval Entonces
107                  reclangval ← longval
108          FinSi
109      FinPara
110  FinPara
111  FinSubProceso
112
113  //Crear matriz A+B
114  SubProceso crearMatrizAB(z Por Referencia,res,reclangval,n,m)
115      definir r,c Como Entero
116      Definir textvalf Como Caracter
117      para r ← 1 hasta n Con Paso 1 Hacer
118          si r = redon(n/2) Entonces
```

```
119      Escribir 'A + B = ' Sin Saltar
120  SiNo
121      Escribir '          ' Sin Saltar
122  FinSi
123  para c ← 1 hasta m con paso 1 Hacer
124      z ← 1
125      espacios(textvalf, reclongval, z, res, val, r, c, i)
126      Escribir '|', textvalf Sin Saltar
127  FinPara
128  Escribir "|"
129  FinPara
130 FinSubProceso
131
132 //Error
133 SubProceso error
134     Escribir '[!] ERROR: VERIFICA TU INFORMACIÓN [!]'
135     Escribir ''
136 FinSubProceso
137
138 //PROCESO PRINCIPAL
139 proceso suma_mtrcs_rectangulares
140     Definir n,m,reclongval Como Entero
141     Definir res,val Como Real
142     Definir textval Como Caracter
143     tamMatriz(n,m)
144     Dimension val[n,m,2]; Dimension res[n,m]
145     registroMatriz(val, reclongval, n, m)
146     crearMatriz(z, val, reclongval, n, m)
147     sumarMatriz(res, reclongval, n, m, val)
148     crearMatrizAB(z, res, reclongval, n, m)
149 FinProceso
```

EJECUCIÓN:



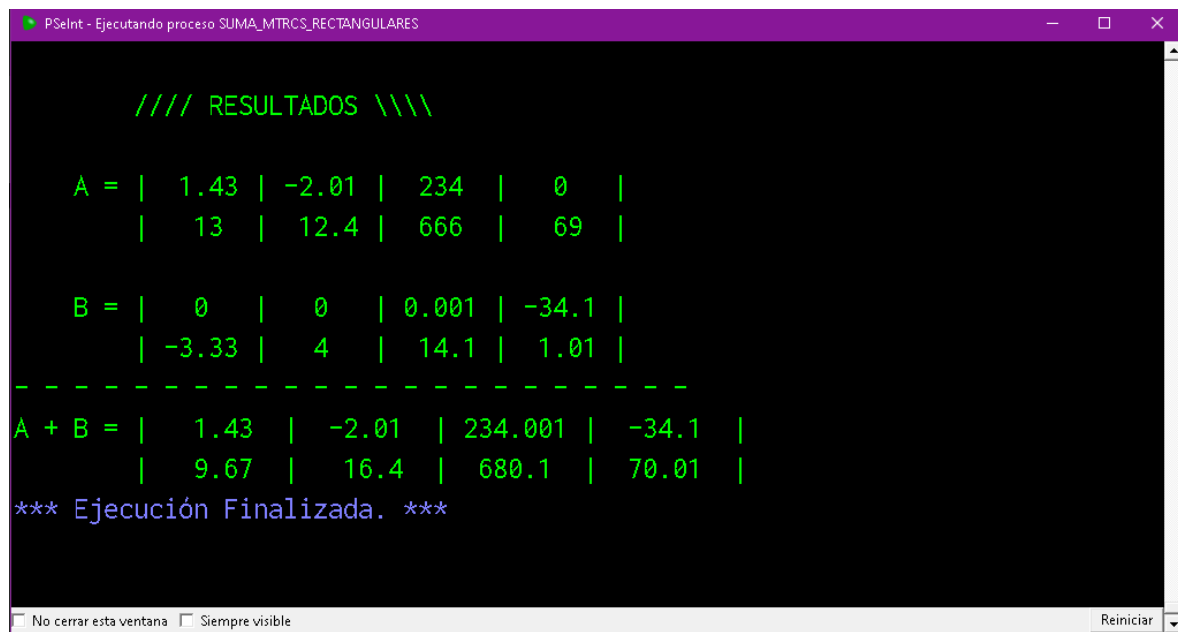
```
*** Ejecución Iniciada. ***
-- SUMA DE DOS MATRICES RECTANGULARES DE ENTEROS (nxm) --
Ingresar tamaño de las matrices (n,m > 1)

Cantidad de filas (n)
> 2
Cantidad de columnas (m)
> 4

|||| Ingresar valores para la matriz A ||||
>> FILA 1
> 1.43
> -2.01
> 234
> 0
>> FILA 2
> 13
> 12.4
> 666
> 69

|||| Ingresar valores para la matriz B ||||
>> FILA 1
> 0
> 0
> 0.001
> -34.1
>> FILA 2
> -3.33
> 4
> 14.1
> 1.01
```

☐ No cerrar esta ventana ☐ Siempre visible Ejecutar desde este punto



```
PSInt - Ejecutando proceso SUMA_MTRCS_RECTANGULARES

//// RESULTADOS \\\

A = |  1.43 | -2.01 |  234 |   0 |
    |  13  |  12.4 |  666 |  69 |

B = |   0 |   0 | 0.001 | -34.1 |
    | -3.33 |  4 |  14.1 |  1.01 |
-----
A + B = |  1.43 | -2.01 | 234.001 | -34.1 |
        |  9.67 |  16.4 |  680.1 |  70.01 |
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

DIAGRAMA MODULAR [4] (Pt. I):

IDENTIDAD			
VARIABLES GLOBALES:		n,reclongval,diag,zero,val	
1	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	tamMatriz	ingresoMatriz	identidad
VARIABLES LOCALES		r,c,longval	
PARAMETROS ENTRADA	n	val,reclongval,diag,zero,n	diag,zero,val,r,c
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Cuando r y c coincidan en valor, comparar el valor de val por una estructura “Si”, en la cual, si val vale 1, su valor será guardado en una nueva variable haciendo una suma recursiva, lo mismo con ceros de caso contrario.
SALIDA	n	val,reclongval,diag,zero	diag,zero

DIAGRAMA MODULAR [4] (Pt. II):

IDENTIDAD			
VARIABLES GLOBALES:		n,reclongval,diag,zero,val	
2	PROCESO PRINCIPAL		
	Subproceso espacios	Subproceso salida	Subproceso error
VARIABLES LOCALES	e,switch,iz,dr,x,textval	i,r,c,l,txtcalf	
PARAMETROS ENTRADA	txtvalf,reclongval,z,res,val,r,c	diag,zero,n,val,reclongval	
OPERACIONES	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.	Comparar las variables donde fueron almacenados los 1 y 0 y determinar si la matriz es identidad o no mediante la estructura “Si”, después de ello, diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Aviso de error al usuario, solicita que verifique su información.
SALIDA	txtvalf		

4. Hacer un pseudocódigo para verificar si una matriz es la matriz identidad.

PSEUDOCÓDIGO:

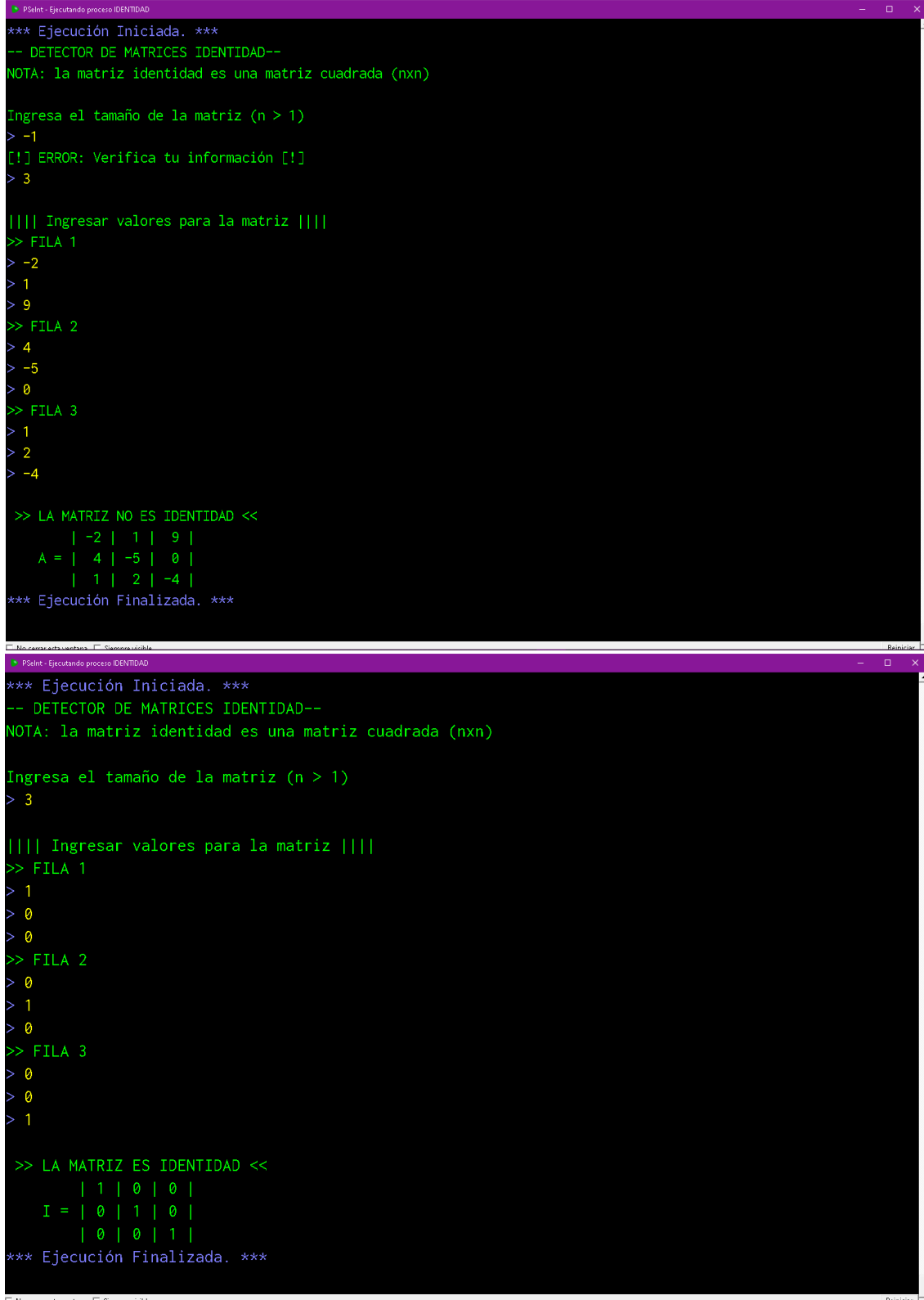
```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de las matrices
5 SubProceso tamMatriz(n Por Referencia)
6     Escribir '-- DETECTOR DE MATRICES IDENTIDAD-- '
7     Escribir 'NOTA: la matriz identidad es una matriz cuadrada (nxn) '
8     Escribir ''
9     Escribir 'Ingresa el tamaño de la matriz (n > 1) '
10    Repetir
11        Leer n
12        Si n ≤ 1 Entonces
13            error
14        FinSi
15    Hasta Que n > 1
16    Escribir ''
17 FinSubProceso
18
19 //Ingreso de valores
20 SubProceso ingresoMatriz(val Por Referencia, reclangval Por Referencia, diag
Por Referencia, zero Por Referencia, n)
21    Definir r, c, longval Como Entero
22    reclangval ← 0
23    Escribir '|||| Ingresar valores para la matriz ||||'
24    para r ← 1 hasta n Con Paso 1 Hacer
25        Escribir '>> FILA ', r
26        para c ← 1 hasta n con paso 1 hacer
27            leer val[r, c]
28            //longitud del valor
29            longval ← Longitud(ConvertirATexto(val[r, c]))
30            si longval > reclangval Entonces
31                reclangval ← longval
32            FinSi
33            identidad(diag, zero, val, r, c)
34        FinPara
35    FinPara
36 FinSubProceso
```



```
38
39 //Identificar Identidad
40 SubProceso identidad(diag Por Referencia,zero Por Referencia,val,r,c)
41     si r = c Entonces
42         Si val[r,c] = 1 Entonces
43             diag ← diag + val[r,c]
44         FinSi
45     SiNo
46         zero ← zero + val[r,c]
47     FinSi
48 FinSubProceso
49
50 //concatenación de espacios
51 SubProceso espacios(textvalf Por Referencia,reclongval,z,res,val,r,c)
52     Definir e,switch Como Entero
53     Definir iz,dr,x,textval Como Caracter
54     iz ← ' '; dr ← ' '; x ← ' '
55     switch ← 0
56     para e ← 1 hasta reclongval+2 con paso 1 Hacer
57         textval ← ConvertirATexto(val[r,c])
58         textvalf ← Concatenar(Concatenar(iz,textval),dr)
59         e ← Longitud(textvalf)
60         si e < reclongval+2 y switch = 0 Entonces
61             iz ← Concatenar(iz,x)
62             switch ← 1
63         SiNo
64             dr ← Concatenar(x,dr)
65             switch ← 0
66         FinSi
67     FinPara
68 FinSubProceso
69
70 //Salida Identidad
71 SubProceso salida(diag,zero,n,val,reclongval)
72     Definir i,r,c Como Entero
73     Definir l,textvalf Como Caracter
74     Escribir ''
75     Si diag = n y zero = 0 Entonces
76         Escribir ' >> LA MATRIZ ES IDENTIDAD <<'
77         i = 1
78     SiNo
```

```
79      Escribir ' >> LA MATRIZ NO ES IDENTIDAD <<'
80      i = 2
81  FinSi
82  Segun i
83      1: 1 ← 'I'
84      2: 1 ← 'A'
85  FinSegun
86  para r ← 1 hasta n con paso 1 Hacer
87      si r = redon(n/2) Entonces
88          Escribir '      ',1,' = ' Sin Saltar
89      SiNo
90          Escribir '      ' Sin Saltar
91      FinSi
92      para c ← 1 hasta n con paso 1 Hacer
93          espacios(textvalf,reclongval,z,res,val,r,c)
94          Escribir '|',textvalf Sin Saltar
95      FinPara
96      Escribir "|"
97  FinPara
98  FinSubProceso
99
100 //Error
101 SubProceso error
102     Escribir '[:] ERROR: VERIFICA TU INFORMACIÓN [:]'
103     Escribir ''
104 FinSubProceso
105
106 //Proceso Principal
107 Proceso identidad
108     Definir n,reclongval Como Entero
109     Definir diag,zero,val Como real
110     tamMatriz(n)
111     Dimension val[n,n]
112     ingresoMatriz(val,reclongval,diag,zero,n)
113     salida(diag,zero,n,val,reclongval)
114 FinProceso
```

### EJECUCIÓN:



```
*** Ejecución Iniciada. ***
-- DETECTOR DE MATRICES IDENTIDAD--
NOTA: la matriz identidad es una matriz cuadrada (nxn)

Ingresa el tamaño de la matriz (n > 1)
> -1
[!] ERROR: Verifica tu información [!]
> 3

|||| Ingresar valores para la matriz ||||
>> FILA 1
> -2
> 1
> 9
>> FILA 2
> 4
> -5
> 0
>> FILA 3
> 1
> 2
> -4

>> LA MATRIZ NO ES IDENTIDAD <<
      | -2 | 1 | 9 |
A = | 4 | -5 | 0 |
      | 1 | 2 | -4 |
*** Ejecución Finalizada. ***

*** Ejecución Iniciada. ***
-- DETECTOR DE MATRICES IDENTIDAD--
NOTA: la matriz identidad es una matriz cuadrada (nxn)

Ingresa el tamaño de la matriz (n > 1)
> 3

|||| Ingresar valores para la matriz ||||
>> FILA 1
> 1
> 0
> 0
>> FILA 2
> 0
> 1
> 0
>> FILA 3
> 0
> 0
> 1

>> LA MATRIZ ES IDENTIDAD <<
      | 1 | 0 | 0 |
I = | 0 | 1 | 0 |
      | 0 | 0 | 1 |
*** Ejecución Finalizada. ***
```

DIAGRAMA MODULAR [5] (Pt. I):

MATRIZ TRIANGULAR			
VARIABLES GLOBALES:		n,reclongval,ginf,gsup,tinf,tsup,val,textvalf	
1	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	tamMatriz	ingresoMatriz	triangMatriz
VARIABLES LOCALES		r,c,longval	
PARAMETROS ENTRADA	n	val,reclongval,gsup,tsup,ginf,tinf,n	gsup,tsup,ginf,tinf,val,r,c
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Determinar matemáticamente la triangularía, si $r > c$ comparar si los valores de estos son iguales a cero y guardarlos en una variable. si $r < c$ comparar si los valores de estos son iguales a cero y guardarlos en una variable.
SALIDA	n	val,reclongval,gsup,tsup,ginf,tinf	gsup,tsup,ginf,tinf,val

DIAGRAMA MODULAR [5] (Pt. II):

MATRIZ TRIANGULAR				
VARIABLES GLOBALES:		n,reclongval,ginf,gsup,tinf,tsup,val,textvalf		
2	PROCESO PRINCIPAL			
	<i>Subproceso</i>	<i>Subproceso</i>	<i>Subproceso</i>	<i>Subproceso</i>
	<i>salidaMatriz</i>	<i>espacios</i>	<i>crearMatriz</i>	<i>error</i>
VARIABLES LOCALES		e,switch,iz,dr,x,textval	r,c	
PARAMETROS ENTRADA	gsup,tsup,ginf,tinf	textvalf,reclongval,z,res,val,r,c	n,val	
OPERACIONES	Con ayuda del subproceso <i>triangMatriz</i> , determinar la operación de escritura correspondiente, teniendo como casos adicionales la matriz diagonal o determinar si la matriz no es triangular	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Aviso de error al usuario, solicita que verifique su información.
SALIDA		textvalf		

5. Hacer un pseudocódigo para verificar si una matriz cuadrada (NxN) es triangular superior y/o triangular inferior.

**PSEUDOCÓDIGO:**

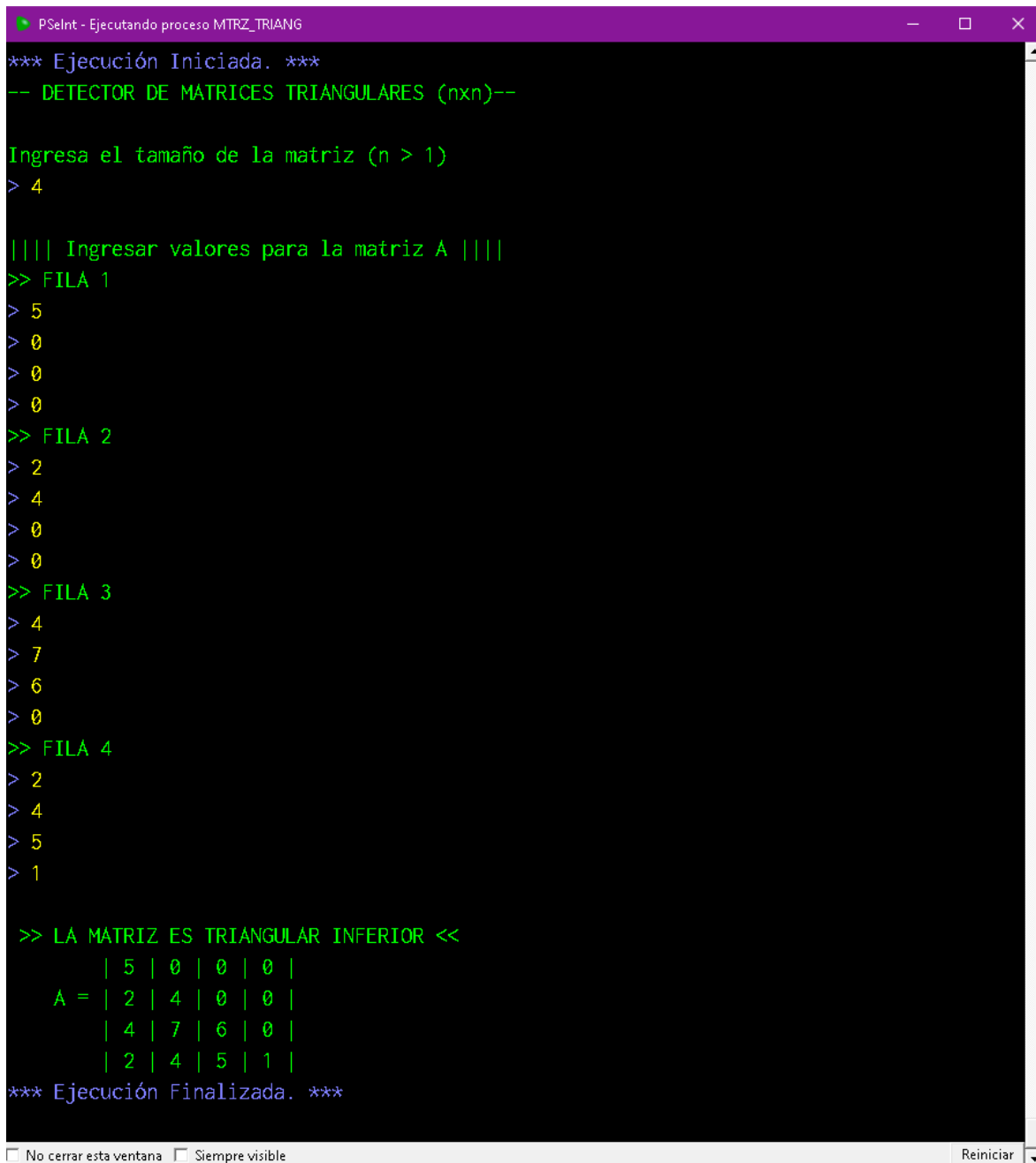
```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de la matriz
5 SubProceso tamMatriz(n Por Referencia)
6     Escribir '-- DETECTOR DE MATRICES TRIANGULARES (nxn)-- '
7     Escribir ''
8     Escribir 'Ingresa el tamaño de la matriz (n > 1)'
9     Repetir
10         Leer n
11         Si n ≤ 1 Entonces
12             error
13         FinSi
14     Hasta Que n > 1
15     Escribir ''
16 FinSubProceso
17
18 //Ingreso de valores
19 SubProceso ingresoMatriz(val Por Referencia, reclangval Por Referencia, gsup
20 Por Referencia, tsup Por Referencia, ginf Por Referencia, tinf Por
21 Referencia, n)
22     Definir r, c, longval Como Entero
23     Escribir '|||| Ingresar valores para la matriz A ||||'
24     para r ← 1 hasta n Con Paso 1 Hacer
25         Escribir '>> FILA ', r
26         para c ← 1 hasta n con paso 1 hacer
27             leer val[r, c]
28             //longitud del valor
29             longval ← Longitud(ConvertirATexto(val[r, c]))
30             si longval > reclangval Entonces
31                 reclangval ← longval
32             FinSi
33         triangMatriz(gsup, tsup, ginf, tinf, val, r, c)
34     FinPara
35 FinSubProceso
36
37 //determinar triangularidad
38 SubProceso triangMatriz(gsup Por Referencia, tsup Por Referencia, ginf Por
39 Referencia, tinf Por Referencia, val, r, c)
40     si r > c Entonces
41         gsup ← gsup + 1
42         si val[r, c] = 0 Entonces
43             tsup ← tsup + 1
```

```
42     FinSi
43 SiNo
44     si r < c Entonces
45         ginf ← ginf + 1
46         si val[r,c] = 0 Entonces
47             tinf ← tinf + 1
48         FinSi
49     FinSi
50 FinSi
51 FinSubProceso
52
53 //Salida traingularidad
54 SubProceso salidaMatriz(gsup,tsup,ginf,tinf)
55     Escribir ' '
56     Si tsup = gsup y tinf = ginf Entonces
57         Escribir ' >> LA MATRIZ ES DIAGONAL <<'
58     SiNo
59         Si tsup = gsup Entonces
60             Escribir ' >> LA MATRIZ ES TRIANGULAR SUPERIOR <<'
61         SiNo
62             Si tinf = ginf Entonces
63                 Escribir ' >> LA MATRIZ ES TRIANGULAR INFERIOR <<'
64             SiNo
65                 Escribir ' >> LA MATRIZ NO ES TRIANGULAR <<'
66         FinSi
67     FinSi
68 FinSi
69 FinSubProceso
70
71 //concatenación de espacios
72 SubProceso espacios(textvalf Por Referencia,reclongval,z,res,val,r,c)
73     Definir e,switch Como Entero
74     Definir iz,dr,x,textval Como Caracter
75     iz ← ' '; dr ← ' '; x ← ' '
76     switch ← 0
77     para e ← 1 hasta reclongval+2 con paso 1 Hacer
78         textval ← ConvertirATexto(val[r,c])
79         textvalf ← Concatenar(Concatenar(iz,textval),dr)
80         e ← Longitud(textvalf)
81         si e < reclongval+2 y switch = 0 Entonces
82             iz ← Concatenar(iz,x)
83             switch ← 1
84         SiNo
85             dr ← Concatenar(x,dr)
86             switch ← 0
87         FinSi
88     FinPara
```

```
89 FinSubProceso
90
91 //Imprimir matriz A
92 SubProceso crearMatriz(n,val)
93     Definir r,c Como Entero
94     para r ← 1 hasta n con paso 1 Hacer
95         si r = redon(n/2) Entonces
96             Escribir '      A = ' Sin Saltar
97         SiNo
98             Escribir '          ' Sin Saltar
99         FinSi
100        para c ← 1 hasta n con paso 1 Hacer
101            espacios(textvalf,reclongval,z,res,val,r,c)
102            Escribir '|',textvalf Sin Saltar
103        FinPara
104        Escribir "|"
105    FinPara
106 FinSubProceso
107
108 //Error
109 SubProceso error
110     Escribir '[:] ERROR: VERIFICA TU INFORMACIÓN [:]'
111     Escribir ''
112 FinSubProceso
113
114 //PROCESO PRINCIPAL
115 Proceso mtrz_triangu
116     Definir n,reclongval,ginf,gsup,tinf,tsup Como Entero
117     Definir val Como real
118     Definir textvalf Como Caracter
119     tamMatriz(n)
120     Dimension val[n,n]
121     ingresoMatriz(val,reclongval,gsup,tsup,ginf,tinf,n)
122     salidaMatriz(gsup,tsup,ginf,tinf)
123     crearMatriz(n,val)
124 FinProceso
```



EJECUCIÓN:



```
*** Ejecución Iniciada. ***
-- DETECTOR DE MATRICES TRIANGULARES (nxn)--

Ingresa el tamaño de la matriz (n > 1)
> 4

|||| Ingresar valores para la matriz A ||||
>> FILA 1
> 5
> 0
> 0
> 0
>> FILA 2
> 2
> 4
> 0
> 0
>> FILA 3
> 4
> 7
> 6
> 0
>> FILA 4
> 2
> 4
> 5
> 1

>> LA MATRIZ ES TRIANGULAR INFERIOR <<
      | 5 | 0 | 0 | 0 |
  A = | 2 | 4 | 0 | 0 |
      | 4 | 7 | 6 | 0 |
      | 2 | 4 | 5 | 1 |
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar

```
PSInt - Ejecutando proceso MTRZ_TRIANG
*** Ejecución Iniciada. ***
-- DETECTOR DE MATRICES TRIANGULARES (nxn)--

Ingresa el tamaño de la matriz (n > 1)
> 4

|||| Ingresar valores para la matriz A ||||
>> FILA 1
> 3
> 7
> 6
> 4
>> FILA 2
> 0
> 6
> 5
> 4
>> FILA 3
> 0
> 0
> 1
> 1
>> FILA 4
> 0
> 0
> 0
> 9

>> LA MATRIZ ES TRIANGULAR SUPERIOR <<

  | 3 | 7 | 6 | 4 |
A = | 0 | 6 | 5 | 4 |
    | 0 | 0 | 1 | 1 |
    | 0 | 0 | 0 | 9 |

*** Ejecución Finalizada. ***
☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

```
PSeInt - Ejecutando proceso MTRZ_TRIANG
*** Ejecución Iniciada. ***
-- DETECTOR DE MATRICES TRIANGULARES (nxn)--

Ingresa el tamaño de la matriz (n > 1)
> 4

||| Ingresar valores para la matriz A |||
>> FILA 1
> 2
> -6
> 25
> 2
>> FILA 2
> 0
> 5
> 6
> 34
>> FILA 3
> -5
> -1
> 5
> 49
>> FILA 4
> 5
> -4
> 2
> 0

>> LA MATRIZ NO ES TRIANGULAR <<

      | 2 | -6 | 25 | 2 |
A =   | 0 | 5 | 6 | 34 |
      | -5 | -1 | 5 | 49 |
      | 5 | -4 | 2 | 0 |

*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

```
PSelnt - Ejecutando proceso MTRZ_TRIANG
*** Ejecución Iniciada. ***
-- DETECTOR DE MATRICES TRIANGULARES (nxn)--

Ingresa el tamaño de la matriz (n > 1)
> 4

||| Ingresar valores para la matriz A |||
>> FILA 1
> -4
> 0
> 0
> 0
>> FILA 2
> 0
> 7
> 0
> 0
>> FILA 3
> 0
> 0
> 1
> 0
>> FILA 4
> 0
> 0
> 0
> 12

>> LA MATRIZ ES DIAGONAL <<
      | -4 | 0 | 0 | 0 |
A =   | 0 | 7 | 0 | 0 |
      | 0 | 0 | 1 | 0 |
      | 0 | 0 | 0 | 12 |
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

DIAGRAMA MODULAR [6] (Pt. I):

MATRIZ TRANSPUESTA			
VARIABLES GLOBALES:		n,m,reclongval,val	
1	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	tamMatriz	ingresoMatriz	espacios
VARIABLES LOCALES		r,c,longval	e,switch,iz,dr,x,textval
PARAMETROS ENTRADA	n,m	val,reclongval,n,m	textvalf,reclongval,z,res,val,r,c
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.
SALIDA	n	val,reclongval	textvalf

DIAGRAMA MODULAR [6] (Pt. II):

MATRIZ TRANSPUESTA			
VARIABLES GLOBALES:		n,m,reclongval,val	
2	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	crearMatriz	transMatriz	error
VARIABLES LOCALES	r,c,textvalf	r,c,textvalf	
PARAMETROS ENTRADA	val,reclongval,n,m	val,reclongval,n,m	
OPERACIONES	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para las columnas, dentro de la estructura “Para”, diseñar una estructura “Para” para los renglones, dentro de la estructura, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Aviso de error al usuario, solicita que verifique su información.
SALIDA			

6. Hacer un pseudocódigo para obtener la matriz transpuesta de una matriz dada.

PSEUDOCÓDIGO:

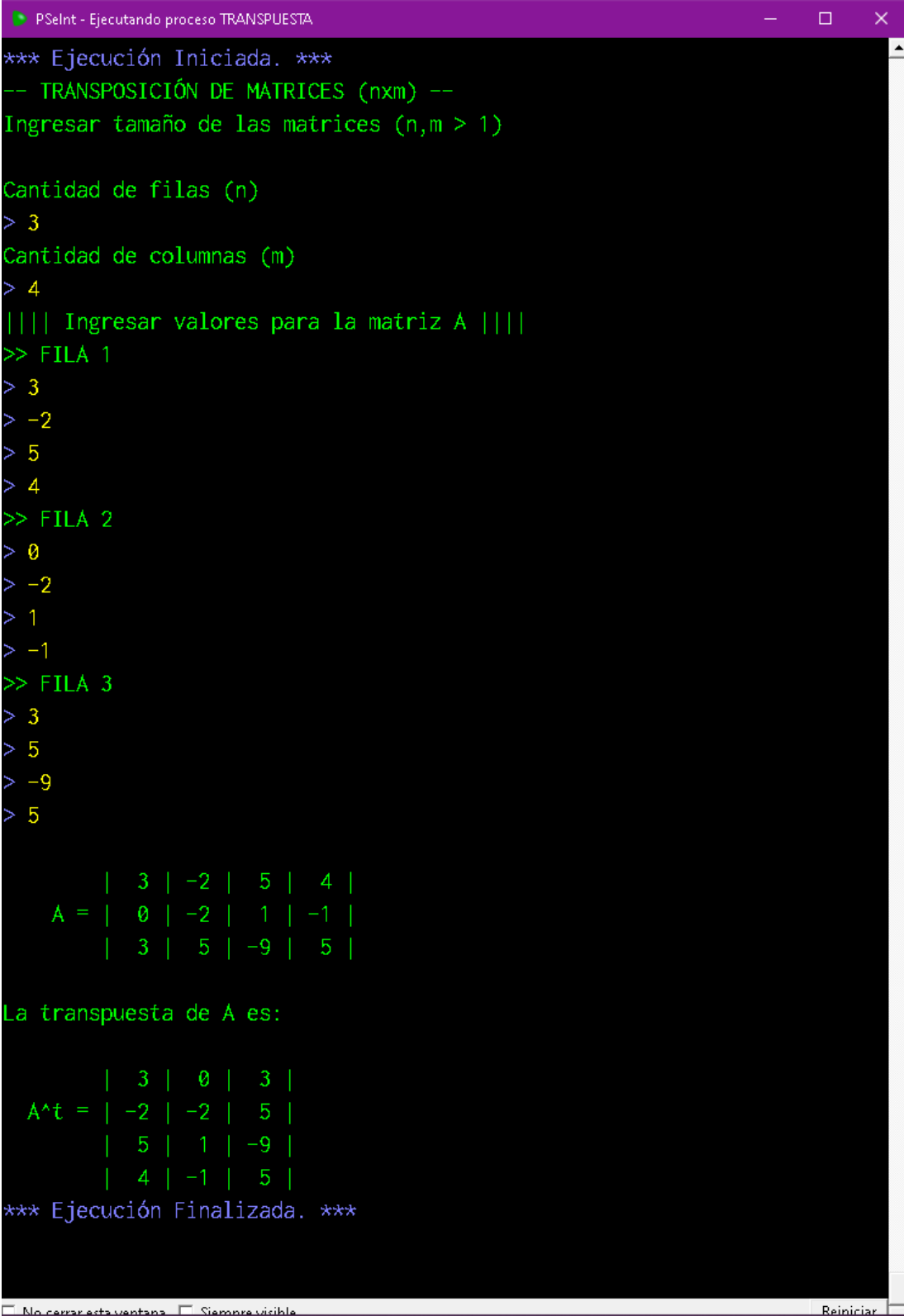
```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de la matriz
5 SubProceso tamMatriz(n Por Referencia,m Por Referencia)
6     Escribir '-- TRANSPOSICIÓN DE MATRICES (nxm) --'
7     Escribir 'Ingresar tamaño de las matrices (n,m > 1)'
8     Repetir
9         Escribir ''
10        Escribir 'Cantidad de filas (n)'
11        Leer n
12        Escribir 'Cantidad de columnas (m)'
13        Leer m
14        Si n ≤ 1 o m ≤ 1 Entonces
15            error
16        FinSi
17        Hasta Que n > 1 y m > 1
18        Escribir ''
19 FinSubProceso
20
21 //Ingreso de valores
22 SubProceso ingresoMatriz(val Por Referencia,reclongval Por Referencia,n,m)
23     Definir r,c,longval Como Entero
24     Escribir '|||| Ingresar valores para la matriz A ||||'
25     para r ← 1 hasta n Con Paso 1 Hacer
26         Escribir '>> FILA ',r
27         para c ← 1 hasta m con paso 1 hacer
28             leer val[r,c]
29             //longitud del valor
30             longval ← Longitud(ConvertirATexto(val[r,c]))
31             si longval > reclongval Entonces
32                 reclongval ← longval
33             FinSi
34         FinPara
35     FinPara
36 FinSubProceso
37
38 //concatenación de espacios
39 SubProceso espacios(textvalf Por Referencia,reclongval,res,val,r,c)
40     Definir e,switch Como Entero
41     Definir iz,dr,x,textval Como Caracter
42     iz ← ' '; dr ← ' '; x ← ' '
43     switch ← 0
44     para e ← 1 hasta reclongval+2 con paso 1 Hacer
```

```
45     textval ← ConvertirATexto(val[r,c])
46     textvalf ← Concatenar(Concatenar(iz,textval),dr)
47     e ← Longitud(textvalf)
48     si e < reclangval+2 y switch = 0 Entonces
49         iz ← Concatenar(iz,x)
50         switch ← 1
51     SiNo
52         dr ← Concatenar(x,dr)
53         switch ← 0
54     FinSi
55     FinPara
56 FinSubProceso
57
58 //Imprimir matriz A
59 SubProceso crearMatriz(val,reclangval,n,m)
60     Definir r,c Como Entero
61     Definir textvalf Como Caracter
62     Escribir ''
63     para r ← 1 hasta n con paso 1 Hacer
64         si r = redon(n/2) Entonces
65             Escribir '      A = ' Sin Saltar
66         SiNo
67             Escribir '          ' Sin Saltar
68         FinSi
69         para c ← 1 hasta m con paso 1 Hacer
70             espacios(textvalf,reclangval,res,val,r,c)
71             Escribir '|',textvalf Sin Saltar
72         FinPara
73         Escribir "|"
74     FinPara
75 FinSubProceso
76
77 //Imprimir transpuesta de matriz A
78 SubProceso transMatriz(val,reclangval,n,m)
79     Definir r,c Como Entero
80     Definir textvalf Como Caracter
81     Escribir ''
82     Escribir 'La transpuesta de A es:'
83     Escribir ''
84     para c ← 1 hasta m con paso 1 Hacer
85         si c = redon(m/2) Entonces
86             Escribir '  A^t = ' Sin Saltar
87         SiNo
88             Escribir '          ' Sin Saltar
89         FinSi
90         para r ← 1 hasta n con paso 1 Hacer
91             espacios(textvalf,reclangval,res,val,r,c)
```



```
92         Escribir '|',textvalf Sin Saltar
93     FinPara
94     Escribir "|"
95 FinPara
96 FinSubProceso
97
98 //Error
99 SubProceso error
100     Escribir '[] ERROR: VERIFICA TU INFORMACIÓN []'
101     Escribir ''
102 FinSubProceso
103
104 //PROCESO PRINCIPAL
105 Proceso transpuesta
106     Definir n,m,reclongval Como Entero
107     Definir val Como real
108     tamMatriz(n,m)
109     Dimension val[n,m]
110     ingresoMatriz(val,reclongval,n,m)
111     crearMatriz(val,reclongval,n,m)
112     transMatriz(val,reclongval,n,m)
113 FinProceso
```

EJECUCIÓN:



```
PSInt - Ejecutando proceso TRANSPUESTA

*** Ejecución Iniciada. ***
-- TRANSPOSICIÓN DE MATRICES (n x m) --
Ingresar tamaño de las matrices (n, m > 1)

Cantidad de filas (n)
> 3
Cantidad de columnas (m)
> 4
| | | Ingresar valores para la matriz A | | |
>> FILA 1
> 3
> -2
> 5
> 4
>> FILA 2
> 0
> -2
> 1
> -1
>> FILA 3
> 3
> 5
> -9
> 5

      | 3 | -2 | 5 | 4 |
A =   | 0 | -2 | 1 | -1 |
      | 3 | 5 | -9 | 5 |

La transpuesta de A es:

      | 3 | 0 | 3 |
A^t = | -2 | -2 | 5 |
      | 5 | 1 | -9 |
      | 4 | -1 | 5 |

*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible 
```

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

DIAGRAMA MODULAR [7] (Pt. I):

SUMA COLUMNA			
VARIABLES GLOBALES:		n,m,reclongval,val,scol	
1	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	tamMatriz	ingresoMatriz	espacios
VARIABLES LOCALES		r,c,longval	e,switch,iz,dr,x,textval
PARAMETROS ENTRADA	n,m	val,reclongval,n,m	textvalf,reclongval,z,res,val,r,c
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.
SALIDA	n	val,reclongval	textvalf

DIAGRAMA MODULAR [7] (Pt. II):

SUMA COLUMNA				
VARIABLES GLOBALES:		n,m,reclongval,val,scol		
2	PROCESO PRINCIPAL			
	<i>Subproceso</i>	<i>Subproces</i>	<i>Subproces</i>	<i>Subproceso</i>
	<i>crearMatriz</i>	<i>columnaMatriz</i>	<i>totalMatriz</i>	<i>error</i>
VARIABLES LOCALES	r,c,textvalf	r,c	r,c,sfinal	
PARAMETROS ENTRADA	val,reclongval,n,m	scol,val,n,m	scol,n,m	
OPERACIONES	Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para las columnas, dentro de la estructura “Para”, diseñar una estructura “Para” para los renglones, dentro de la estructura, sumar todos los valores de val y guardarlos en una nueva variable para hacer una suma progresiva de la misma, después, imprimir el número de columna y su respectivo valor de suma final.	Mostrar en pantalla el aviso de la sección indicada. . Diseñar una estructura “Para” para las columnas, dentro de la estructura, sumar todos los valores finales de las sumas de cada semana y guardarlos en una nueva variable para hacer una suma progresiva de la misma, después, imprimir el valor final de la suma.	Aviso de error al usuario, solicita que verifique su información.
SALIDA				

7. Hacer un pseudocódigo para leer números en una matriz de nxm, y almacenar en un arreglo los resultados de sumar los elementos por columna y desplegar resultados, y finalmente realizar la suma de los elementos del arreglo de resultados y presentar resultados.

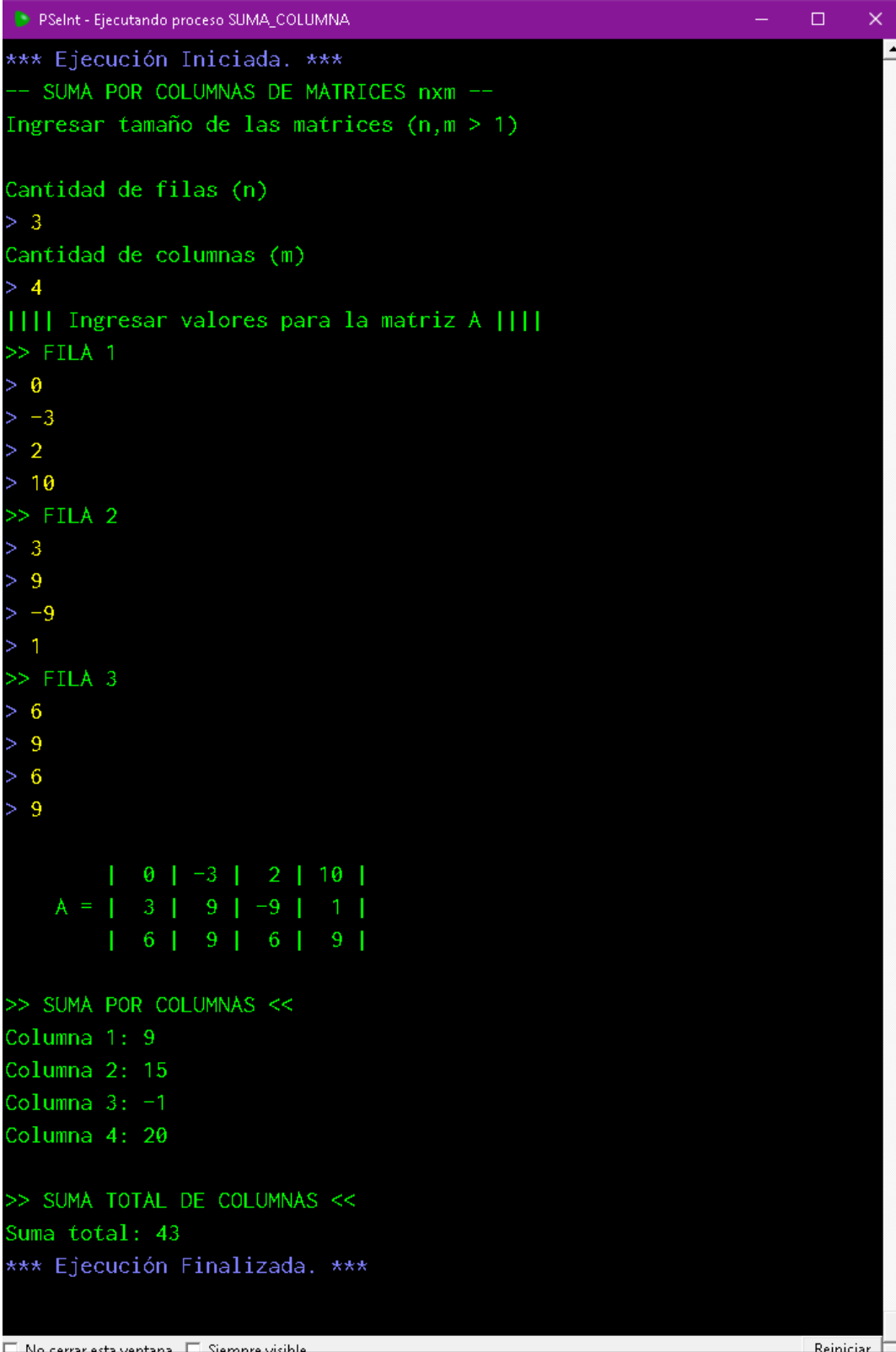
PSEUDOCÓDIGO:

```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de la matriz
5 SubProceso tamMatriz(n Por Referencia,m Por Referencia)
6     Escribir '-- SUMA POR COLUMNAS DE MATRICES nxm --'
7     Escribir 'Ingresar tamaño de las matrices (n,m > 1)'
8     Repetir
9         //tamaño de la matriz
10        Escribir ''
11        Escribir 'Cantidad de filas (n)'
12        Leer n
13        Escribir 'Cantidad de columnas (m)'
14        Leer m
15        Si n ≤ 1 o m ≤ 1 Entonces
16            error
17        FinSi
18        Hasta Que n > 1 y m > 1
19    FinSubProceso
20
21 //Ingreso de valores
22 SubProceso ingresoMatriz(val Por Referencia,reclongval Por Referencia,n,m)
23     Definir r,c,longval Como Entero
24     Escribir '|||| Ingresar valores para la matriz A ||||'
25     para r ← 1 hasta n Con Paso 1 Hacer
26         Escribir '>> FILA ',r
27         para c ← 1 hasta m con paso 1 hacer
28             leer val[r,c]
29             //longitud del valor
30             longval ← Longitud(ConvertirATexto(val[r,c]))
31             si longval > reclongval Entonces
32                 reclongval ← longval
33             FinSi
34         FinPara
35     FinPara
36 FinSubProceso
37
38 //concatenación de espacios
39 SubProceso espacios(textvalf Por Referencia,reclongval,res,val,r,c)
40     Definir e,switch Como Entero
41     Definir iz,dr,x,textval Como Caracter
```

```
42  iz ← ' '; dr ← ' '; x ← ' '
43  switch ← 0
44  para e ← 1 hasta reclangval+2 con paso 1 Hacer
45      textval ← ConvertirATexto(val[r,c])
46      textvalf ← Concatenar(Concatenar(iz,textval),dr)
47      e ← Longitud(textvalf)
48      si e < reclangval+2 y switch = 0 Entonces
49          iz ← Concatenar(iz,x)
50          switch ← 1
51      SiNo
52          dr ← Concatenar(x,dr)
53          switch ← 0
54      FinSi
55  FinPara
56  FinSubProceso
57
58  //Imprimir matriz A
59  SubProceso crearMatriz(val,reclangval,n,m)
60      Definir r,c Como Entero
61      Definir textvalf Como Caracter
62      Escribir ''
63      para r ← 1 hasta n con paso 1 Hacer
64          si r = redon(n/2) Entonces
65              Escribir '      A = ' Sin Saltar
66          SiNo
67              Escribir '          ' Sin Saltar
68          FinSi
69          para c ← 1 hasta m con paso 1 Hacer
70              espacios(textvalf,reclangval,res,val,r,c)
71              Escribir '|',textvalf Sin Saltar
72          FinPara
73          Escribir "|"
74      FinPara
75  FinSubProceso
76
77  //suma de columnas
78  SubProceso columnaMatriz(scol Por Referencia,val,n,m)
79      Definir r,c Como Entero
80      Escribir ''
81      Escribir '>> SUMA POR COLUMNAS <<'
82      para c ← 1 hasta m con paso 1 Hacer
83          para r ← 1 hasta n con paso 1 Hacer
84              scol[c] ← scol[c] + val[r,c]
85          FinPara
86          Escribir 'Columna ',c,': ',scol[c]
87      FinPara
88  FinSubProceso
```

```
89
90 //suma total de columnas
91 SubProceso totalMatriz(scol,n,m)
92     Definir r,c Como Entero
93     Definir sfinal Como Real
94     Escribir ''
95     Escribir '>> SUMA TOTAL DE COLUMNAS <<'
96     para c ← 1 hasta m con paso 1 Hacer
97         sfinal ← sfinal + scol[c]
98     FinPara
99     Escribir 'Suma total: ',sfinal
100 FinSubProceso
101
102 //Error
103 SubProceso error
104     Escribir '[] ERROR: VERIFICA TU INFORMACIÓN []'
105     Escribir ''
106 FinSubProceso
107
108 //PROCESO PRINCIPAL
109 Proceso suma_columna
110     Definir n,m,reclongval Como Entero
111     Definir val,scol Como real
112     tamMatriz(n,m)
113     Dimension val[n,m]
114     Dimension scol[m]
115     ingresoMatriz(val,reclongval,n,m)
116     crearMatriz(val,reclongval,n,m)
117     columnaMatriz(scol,val,n,m)
118     totalMatriz(scol,n,m)
119 FinProceso
```

### EJECUCIÓN:



```
PSInt - Ejecutando proceso SUMA_COLUMNA

*** Ejecución Iniciada. ***
-- SUMA POR COLUMNAS DE MATRICES nxm --
Ingresar tamaño de las matrices (n,m > 1)

Cantidad de filas (n)
> 3
Cantidad de columnas (m)
> 4
||| Ingresar valores para la matriz A |||
>> FILA 1
> 0
> -3
> 2
> 10
>> FILA 2
> 3
> 9
> -9
> 1
>> FILA 3
> 6
> 9
> 6
> 9

      | 0 | -3 | 2 | 10 |
A =   | 3 | 9 | -9 | 1 |
      | 6 | 9 | 6 | 9 |

>> SUMA POR COLUMNAS <<
Columna 1: 9
Columna 2: 15
Columna 3: -1
Columna 4: 20

>> SUMA TOTAL DE COLUMNAS <<
Suma total: 43
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible 
```



Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

DIAGRAMA MODULAR [8] (Pt. I):

DIAGONAL PRIMOS			
VARIABLES GLOBALES:		n,m,reclongval,val,tprim,prim	
1	PROCESO PRINCIPAL		
	Subproceso	Subproceso	Subproceso
	tamMatriz	ingresoMatriz	espacios
VARIABLES LOCALES		p,r,c,nprim,longval	e,switch,iz,dr,x,textval
PARAMETROS ENTRADA	n,m	val,reclongval,prim,tprim,n,m	textvalf,reclongval,z,res,val,r,c
OPERACIONES	Mostrar en pantalla el aviso de la sección indicada, Solicitar el ingreso del tamaño de las matrices y guardar el valor en una variable, detectar cuando el valor ingresado sea menor a 1.	Mostrar en pantalla el aviso de la sección indicada. Diseñar una estructura “Para” para los renglones, dentro de la estructura “Para”, diseñar una estructura “Para” para las columnas, dentro de la estructura para ingresar los valores y guardarlos en una variable nueva.	Este subproceso ayudará a imprimir correctamente las matrices, se determinará la longitud mayor de todos los valores ingresados y tendrá un ajuste para la matriz de entrada y la de salida.
SALIDA	n	val,reclongval,prim,tprim	textvalf

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

DIAGRAMA MODULAR [8] (Pt. II):

DIAGONAL PRIMOS			
VARIABLES GLOBALES:		n,m,reclongval,val,tprim,prim	
2	PROCESO PRINCIPAL		
	Subproceso	Subproces	Subproces
	crearMatriz	primos	salidaPrimos
VARIABLES LOCALES	r,c,textvalf	j,nprim	
PARAMETROS ENTRADA	val,reclongval,n,m	prim,tprim,p,val,r,c	
OPERACIONES	<p>Diseñar una estructura “Para” para los renglones, dentro de esta, diseñar una estructura “Para” para las columnas, dentro de esta, invocar el subproceso Espacios para facilitar la impresión de la matriz misma.</p> <p>Mostrar en pantalla el aviso de la sección indicada. Con la estructura “Si” y la información del subproceso <i>primos</i>, Escribir si la matriz tiene o no primos en su diagonal, si los tiene, mostrar cuantos y cuáles son los números en cuestión.</p>	<p>Definir matemáticamente la ubicación de un primo; cuando r y c tengan los mismos valores (con ayuda de la estructura “Si”), Diseñar una estructura “Para”, dentro de esta, pasara cada valor y se definirá su modulo desde 1 hasta el valor val, guardar en una nueva variable recursivamente cuantas veces el módulo del valor es 0. Después, con ayuda de otra estructura “Si”, determinar si dicho número solo tuvo 2 veces el módulo 0, si es verdad, ingresar en una nueva variable el registro del nuevo valor primo, el valor en cuestión y una variable nueva que determine el siguiente probable número.</p>	<p>Mostrar en pantalla el aviso de la sección indicada. Con la estructura “Si” y la información del subproceso <i>primos</i>, Escribir si la matriz tiene o no primos en su diagonal, si los tiene, mostrar cuantos y cuáles son los números en cuestión.</p>
SALIDA		prim,tprim,p	

8. Hacer un pseudocódigo para determinar el número de números primos existentes en la diagonal de una matriz.

PSEUDOCÓDIGO:

```
1 //Nombre: Jesús Huerta Aguilar
2 //Matricula: 202041509
3
4 //Definir tamaño de la matriz
5 SubProceso tamMatriz(n Por Referencia,m Por Referencia)
6     Escribir '-- PRIMOS EXISTENTES EN LA DIAGONAL PRINCIPAL nxm --'
7     Escribir 'Ingresar tamaño de la matriz (n,m > 1)'
8     Repetir
9         Escribir ''
10        Escribir 'Cantidad de filas (n)'
11        Leer n
12        Escribir 'Cantidad de columnas (m)'
13        Leer m
14        Si n ≤ 1 o m ≤ 1 Entonces
15            Escribir '[] ERROR: Verifica tu información []'
16        FinSi
17        Hasta Que n > 1 y m > 1
18    FinSubProceso
19
20 //Ingreso de valores
21 SubProceso ingresoMatriz(val Por Referencia,reclongval Por Referencia,prim
Por Referencia,tprim Por Referencia,n,m)
22     Definir p,r,c,nprim,longval Como Entero
23     tprim ← 0
24     p ← 1
25     Escribir '|||| Ingresar valores para la matriz A ||||'
26     para r ← 1 hasta n Con Paso 1 Hacer
27         Escribir '>> FILA ',r
28         para c ← 1 hasta m con paso 1 hacer
29             nprim ← 0
30             leer val[r,c]
31             //longitud del valor
32             longval ← Longitud(ConvertirATexto(val[r,c]))
33             si longval > reclongval Entonces
34                 reclongval ← longval
35             FinSi
36             primos(prim,tprim,p,val,r,c)
37         FinPara
38     FinPara
39 FinSubProceso
40
41 //concatenación de espacios
42 SubProceso espacios(textvalf Por Referencia,reclongval,res,val,r,c)
```

```

43 Definir e,switch Como Entero
44 Definir iz,dr,x,textval Como Caracter
45 iz ← ' '; dr ← ' '; x ← ' '
46 switch ← 0
47 para e ← 1 hasta reclangval+2 con paso 1 Hacer
48     textval ← ConvertirATexto(val[r,c])
49     textvalf ← Concatenar(Concatenar(iz,textval),dr)
50     e ← Longitud(textvalf)
51     si e < reclangval+2 y switch = 0 Entonces
52         iz ← Concatenar(iz,x)
53         switch ← 1
54     SiNo
55         dr ← Concatenar(x,dr)
56         switch ← 0
57     FinSi
58 FinPara
59 FinSubProceso
60
61 //determinar primos
62 SubProceso primos(prim Por Referencia,tprim Por Referencia,p Por
Referencia,val,r,c)
63     Definir j,nprim Como Entero
64     si r = c Entonces
65         Para j←1 Hasta val[r,c] Hacer
66             Si val[r,c] MOD j=0 Entonces
67                 nprim ← nprim + 1
68         FinSi
69     FinPara
70     FinSi
71     Si nprim=2 Entonces
72         tprim ← tprim + 1
73         prim[p] ← val[r,c]
74         p ← p + 1
75     FinSi
76 FinSubProceso
77
78 //Imprimir matriz A
79 SubProceso crearMatriz(val,reclangval,n,m)
80     Definir textvalf Como Caracter
81     Definir r,c Como Entero
82     Definir textvalf Como Caracter
83     Escribir ''
84     para r ← 1 hasta n con paso 1 Hacer
85         si r = redon(n/2) Entonces
86             Escribir '      A = ' Sin Saltar
87         SiNo
88             Escribir '          ' Sin Saltar

```

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

```
89      FinSi
90      para c ← 1 hasta m con paso 1 Hacer
91          espacios(textvalf, reclangval, res, val, r, c)
92          Escribir '|', textvalf Sin Saltar
93      FinPara
94      Escribir "|"
95  FinPara
96 FinSubProceso
97
98 //Salida primos
99 SubProceso salidaPrimos(prim, tprim)
100     Escribir ''
101     Escribir '>> NUMEROS PRIMOS EN LA DIAGONAL PRINCIPAL <<'
102     Si tprim = 0 Entonces
103         Escribir 'La matriz A no tiene numeros primos en su diagonal
principal'
104     SiNo
105         Escribir 'La matriz A tiene ', tprim, ' numeros primos en su
diagonal principal: {' Sin Saltar
106         para p ← 1 hasta tprim Con Paso 1 Hacer
107             Escribir prim[p] Sin Saltar
108             Si p ≤ tprim - 1 Entonces
109                 Escribir ',' Sin Saltar
110         FinSi
111     FinPara
112     Escribir '}'
113 FinSi
114 FinSubProceso
115
116 //PROCESO PRINCIPAL
117 Proceso diagonal_primos
118     Definir n, m, reclangval, val, tprim, prim Como Entero
119     tamMatriz(n, m)
120     Dimension val[n, m]
121     Dimension prim[n]
122     ingresoMatriz(val, reclangval, prim, tprim, n, m)
123     crearMatriz(val, reclangval, n, m)
124     salidaPrimos(prim, tprim)
125 FinProceso
```

EJECUCIÓN:

```
PSInt - Ejecutando proceso DIAGONAL_PRIMOS
-- PRIMOS EXISTENTES EN LA DIAGONAL PRINCIPAL nxm --
Ingresar tamaño de la matriz (n,m > 1)

Cantidad de filas (n)
> 4
Cantidad de columnas (m)
> 4
||| Ingresar valores para la matriz A |||
>> FILA 1
> 7
> 4
> 6
> 2
>> FILA 2
> 5
> 5
> -9
> 0
>> FILA 3
> -1
> 9
> 8
> 10
>> FILA 4
> 2
> 4
> -7
> 11

      | 7 | 4 | 6 | 2 |
A =   | 5 | 5 | -9 | 0 |
      | -1 | 9 | 8 | 10 |
      | 2 | 4 | -7 | 11 |

>> NUMEROS PRIMOS EN LA DIAGONAL PRINCIPAL <<
La matriz A tiene 3 numeros primos en su diagonal principal: {7,5,11}
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Ejecutar desde este punto
```

```
PSelnt - Ejecutando proceso DIAGONAL_PRIMOS
-- PRIMOS EXISTENTES EN LA DIAGONAL PRINCIPAL nxm --
Ingresar tamaño de la matriz (n,m > 1)

Cantidad de filas (n)
> 4
Cantidad de columnas (m)
> 4
|||| Ingresar valores para la matriz A ||||
>> FILA 1
> 4
> 2
> -6
> 10
>> FILA 2
> 5
> 8
> -11
> 0
>> FILA 3
> 5
> -5
> 16
> -2
>> FILA 4
> 0
> 6
> 9
> 6

      | 4 | 2 | -6 | 10 |
A =   | 5 | 8 | -11 | 0 |
      | 5 | -5 | 16 | -2 |
      | 0 | 6 | 9 | 6 |

>> NUMEROS PRIMOS EN LA DIAGONAL PRINCIPAL <<
La matriz A no tiene numeros primos en su diagonal principal
*** Ejecución Finalizada. ***
☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```