



**PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL  
FORMATO ENTREGA DE EVIDENCIAS**

**RESOLVER PROBLEMAS DEL ENTORNO APLICANDO  
FUNDAMENTOS DE LÓGICA PROPOSICIONAL**

**Presentado a:** Instructor César Marino Cuéllar Chacón  
**Por Aprendiz:** **Andrea Melissa Eraso Montero**  
**Ficha:** 3312932  
**Competencia:** **ALGORITMIA**

Tecnólogo en Análisis y Desarrollo de Software  
Servicio Nacional de Aprendizaje SENA  
Centro de Teleinformática y Producción Industrial  
Regional Cauca

Popayán, día **05** de **DICIEMBRE** del año **2025**



**PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL  
FORMATO ENTREGA DE EVIDENCIAS**

## **Tabla de Contenido**

Contenido

Tabla de contenido

<b>1. Actividad o ejercicio 1.....</b>	3
<b>1.1 Enunciado.....</b>	3
<b>2. Actividad o Ejercicio 2 .....</b>	4
<b>2.1 Enunciado .....</b>	4
<b>3. Actividad o ejercicio 3.....</b>	5
<b>    3.1 Enunciado.....</b>	5
<b>4. Actividad o ejercicio 4.....</b>	6
<b>    4.1 Enunciado.....</b>	6
<b>5. Actividad o ejercicio 5.....</b>	7
<b>    5.1 Enunciado.....</b>	7
<b>6. Actividad o ejercicio 6.....</b>	8
<b>    6.1 Enunciado.....</b>	8
<b>7. Actividad o ejercicio 7.....</b>	8
<b>    7.1 Enunciado.....</b>	8
<b>8. Actividad o ejercicio 8.....</b>	9
<b>    8.1 Enunciado.....</b>	9
<b>9. Actividad o ejercicio 9.....</b>	10
<b>    9.1 Enunciado.....</b>	10
<b>10. Actividad o ejercicio 10 .....</b>	11
<b>    10.1 Enunciado .....</b>	11
<b>11. Actividad o ejercicio 11 .....</b>	12
<b>    11.1 Enunciado .....</b>	12
<b>12. Actividad o ejercicio 12 .....</b>	13
<b>    12.1 Enunciado .....</b>	13



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### 1. Actividad o ejercicio 1.

#### 1.1 Enunciado.

Hacer un algoritmo que sume dos arreglos de 5 posiciones con números enteros en un tercer arreglo. Se debe mostrar el contenido de los dos arreglos con números leídos, así como también el contenido del arreglo con la suma. Los números deben ser ingresados por el usuario.

#### *solución*

```
1 Algoritmo suma_dos_arreglos
2   Dimensionar entrada1[5]
3   Dimensionar entrada2[5]
4   Dimensionar sumaEntradas[5]
5
6   Definir i, j, k, suma, entrada1, entrada2, sumaEntradas Como Entero
7
8   escribir "Captura de números del primer arreglo"
9   para i=0 Hasta 4 Hacer
10    Escribir "digite un numero entero para guardar en entrada1[",i,"]: "
11    leer entrada1[i]
12 FinPara
13
14   escribir "Captura de números del segundo arreglo"
15   para j=0 Hasta 4 Hacer
16    Escribir "digite un numeros entero para guardar en entrada2[",j,"]: "
17    leer entrada2[i]
18 FinPara
19
20   para i=0 Hasta 4 Hacer
21    sumaEntradas[i]:= entrada1[i] + entrada2[i]
22
23 FinPara
24
25   escribir "PRIMER ARREGLO"
26
27   para i=0 Hasta 4 Hacer
28    escribir entrada1[i], " " Sin Saltar
29 FinPara
30
31   escribir ""
32   escribir "SEGUNDO ARREGLO"
33
34   para i=0 Hasta 4 Hacer
35    escribir entrada2[i], " " Sin Saltar
36 FinPara
37
38   escribir ""
39   escribir "la suma de los dos arreglos en otro arreglo es: "
40
41
42   para i=0 Hasta 4 Hacer
43    escribir sumaEntradas[i], " " Sin Saltar
44 FinPara
45   escribir ""
46
47
48 FinAlgoritmo
49
```

\*\*\* Ejecución Iniciada. \*\*\*  
Captura de números del primer arreglo  
digite un numero entero para guardar en entrada1[0]:  
> 8  
digite un numero entero para guardar en entrada1[1]:  
> 20  
digite un numero entero para guardar en entrada1[2]:  
> 17  
digite un numero entero para guardar en entrada1[3]:  
> 22  
digite un numero entero para guardar en entrada1[4]:  
> 45  
Captura de números del segundo arreglo  
digite un numeros entero para guardar en entrada2[0]:  
> 56  
digite un numeros entero para guardar en entrada2[1]:  
> 37  
digite un numeros entero para guardar en entrada2[2]:  
> 98  
digite un numeros entero para guardar en entrada2[3]:  
> 26  
digite un numeros entero para guardar en entrada2[4]:  
> 9  
PRIMER ARREGLO  
8 20 17 22 45  
SEGUNDO ARREGLO  
9 0 0 0 0  
la suma de los dos arreglos en otro arreglo es:  
17 20 17 22 45  
\*\*\* Ejecución Finalizada. \*\*\*

No cerrar esta ventana  Siempre visible Reiniciar



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### 2. Actividad o Ejercicio 2

#### 2.1 Enunciado

Hacer un algoritmo que lea un arreglo de 5 posiciones con el nombre de países y como resultado se debe mostrar el arreglo con leído y otro arreglo generado que va a contener la cantidad de letras del nombre del país.

#### *solución*

The screenshot shows the PSELint interface with two panes. The left pane displays the pseudocode for the algorithm, and the right pane shows the execution results.

**Pseudocode:**

```
1  Algoritmo nombre_paises
2      Dimensionar nombrePaises[5]
3      Dimensionar numeroLetras[5]
4      Definir numeroLetras, i Como Entero
5      Definir nombrePaises Como Caracter
6
7      para i=0 hasta 4 Hacer
8          Escribir "Digite un nombre de un pais"
9          leer nombrePaises[i]
10     FinPara
11
12    para i=0 Hasta 4 Hacer
13        numeroLetras[i]←Longitud(nombrePaises[i])
14    FinPara
15
16    escribir "ARREGLO NOMBRES PAISES"
17
18    para i=0 Hasta 4 Hacer
19        Escribir ""
20        Escribir nombrePaises[i], " " Sin Saltar
21    FinPara
22
23    escribir ""
24
25    Escribir "ARREGLO CANTIDAD LETRAS DE NOMBRES DE PAISES"
26
27    Para i=0 Hasta 4 Hacer
28        Escribir ""
29        Escribir numeroLetras[i], " " Sin Saltar
30    FinPara
31
32
33    Escribir " "
34
35
36 FinAlgoritmo
37
```

**Execution Results:**

```
PSelint - Ejecutando proceso... ━ ━ X
*** Ejecución Iniciada. ***
Digite un nombre de un pais
> colombia
Digite un nombre de un pais
> brazil
Digite un nombre de un pais
> argentina
Digite un nombre de un pais
> africa
Digite un nombre de un pais
> españa
ARREGLO NOMBRES PAISES
colombia
brazil
argentina
africa
españa
ARREGLO CANTIDAD LETRAS DE NOMBRES DE PAISES
9
6
10
6
7
*** Ejecución Finalizada. ***
```

At the bottom of the window, there are three checkboxes: "No cerrar esta ventana" (Do not close this window), "Siempre visible" (Always visible), and "Reinicia" (Restart).



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### 3. Actividad o ejercicio 3

#### 3.1 Enunciado

Hacer un algoritmo que lea 10 números enteros positivos de dos dígitos, como resultado se debe generar otro arreglo con la suma de los dígitos de cada número. Al finalizar se debe imprimir los dos arreglos.

#### *solucion*

```
1  Algoritmo suma_dentro_arreglo
2    Dimensionar numerosDosDigitos[10]
3    Dimensionar SumaNumeros[10]
4
5    Definir numerosDosDigitos, SumaNumeros, dig1, dig2, i Como Entero
6
7    Para i = 0 Hasta 9 Hacer
8      Escribir "Digite un numero de dos digitos"
9      Ler numerosDosDigitos[i]
10     FinPara
11
12    Para i = 0 Hasta 9 Hacer
13      dig1 = trunc(numerosDosDigitos[i] / 10)
14      dig2 = numerosDosDigitos[i] MOD 10
15      SumaNumeros[i] = dig1 + dig2
16     FinPara
17
18
19    Escribir "ARREGLO NUMEROS DE DOS DIGITOS"
20
21    Para i = 0 Hasta 9 Hacer
22      | Escribir numerosDosDigitos[i], " "Sin Saltar
23    FinPara
24    Escribir ""
25    Escribir "ARREGLO SUMA DE LOS DIGITOS"
26
27    Para i = 0 Hasta 9 Hacer
28      | Escribir SumaNumeros[i], " "Sin Saltar
29    FinPara
30
31    Escribir ""
32
33  FinAlgoritmo
34
```

\*\*\* Ejecución Iniciada. \*\*\*
Digite un numero de dos digitos
> 22
Digite un numero de dos digitos
> 17
Digite un numero de dos digitos
> 20
Digite un numero de dos digitos
> 15
Digite un numero de dos digitos
> 12
Digite un numero de dos digitos
> 51
Digite un numero de dos digitos
> 57
Digite un numero de dos digitos
> 86
Digite un numero de dos digitos
> 24
Digite un numero de dos digitos
> 98
ARREGLO NUMEROS DE DOS DIGITOS
22 17 20 15 12 51 57 86 24 98
ARREGLO SUMA DE LOS DIGITOS
4 8 2 6 3 6 12 14 6 17
\*\*\* Ejecución Finalizada. \*\*\*

No cerrar esta ventana  Siempre visible



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### 4. Actividad o ejercicio 4

#### 4.1 Enunciado

Hacer un algoritmo en pseudocódigo que lea los nombres de 10 estudiantes y su respectivo puntaje en las pruebas de estado ICFES. (Puntaje entre 1 y 400). El algoritmo debe mostrar en pantalla:

- Nombre y puntaje del estudiante con mayor puntaje y con menor puntaje
- Promedio de puntajes

#### *solución*

```
1  Algoritmo ICFES_Estudiantes
2
3      Dimensionar nombres[10]
4      Dimensionar puntajes[10]
5      Definir nombres Como Carácter
6      Definir i, mayor, menor, posicionMayor, posicionMenor Como Entero
7      Definir puntajes, suma, promedio Como Real
8
9      suma = 0
10
11     Para i = 0 Hasta 9 Hacer
12         Escribir "Ingrese el nombre del estudiante ", i, ":"
13         Leer nombres[i]
14
15         Escribir "Ingrese el puntaje del estudiante ", nombres[i], " (1 a 400):"
16         Leer puntajes[i]
17
18         suma= suma+puntajes[i]
19
20     FinPara
21
22     mayor=puntajes[1]
23     menor=puntajes[1]
24     posicionMayor=1
25     posicionMenor=1
26
27     Para i = 1 Hasta 9 Hacer
28         Si puntajes[i] > mayor Entonces
29             mayor = puntajes[i]
30             posicionMayor = i
31         FinSi
32
33         Si puntajes[i] < menor Entonces
34             menor = puntajes[i]
35             posicionMenor = i
36         FinSi
37     FinPara
38
39     promedio = suma / 10
40
41     Escribir " "
42     Escribir "RESULTADOS"
43     Escribir "Estudiante con Mayor Puntaje: ", nombres[posicionMayor], " - ", mayor
44     Escribir "Estudiante con Menor Puntaje: ", nombres[posicionMenor], " - ", menor
45     Escribir "Promedio General de Puntajes: ", promedio
46
47 FinAlgoritmo
48
```

```
PSelint - Ejecutando proceso ICFES_ESTUDIANTES
*** Ejecución Iniciada. ***
Ingrese el nombre del estudiante 0:
> sara
Ingrese el puntaje del estudiante sara (1 a 400):
> 210
Ingrese el nombre del estudiante 1:
> sofia
Ingrese el puntaje del estudiante sofia (1 a 400):
> 310
Ingrese el nombre del estudiante 2:
> andrea
Ingrese el puntaje del estudiante andrea (1 a 400):
> 300
Ingrese el nombre del estudiante 3:
> miguel
Ingrese el puntaje del estudiante miguel (1 a 400):
> 295
Ingrese el nombre del estudiante 4:
> felipe
Ingrese el puntaje del estudiante felipe (1 a 400):
> 180
Ingrese el nombre del estudiante 5:
> diego
Ingrese el puntaje del estudiante diego (1 a 400):
> 238
Ingrese el nombre del estudiante 6:
> nicol
Ingrese el puntaje del estudiante nicol (1 a 400):
> 251
Ingrese el nombre del estudiante 7:
> khaterin
Ingrese el puntaje del estudiante khaterin (1 a 400):
> 225
Ingrese el nombre del estudiante 8:
> rodrigo
Ingrese el puntaje del estudiante rodrigo (1 a 400):
> 398
Ingrese el nombre del estudiante 9:
> ismael
Ingrese el puntaje del estudiante ismael (1 a 400):
> 110

RESULTADOS
Estudiante con Mayor Puntaje: rodrigo - 398
Estudiante con Menor Puntaje: ismael - 110
Promedio General de Puntajes: 251.7
```



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### 5. Actividad o ejercicio 5

#### 5.1 Enunciado

Hacer un algoritmo que lea 10 números enteros, como resultado debe informar:

- Promedio de los números pares y promedio de los números impares.
- La suma de los elementos que ocupan las posiciones pares en el arreglo.
- La suma de los elementos que ocupan las posiciones impares en el arreglo.

#### solución

The screenshot shows the PSEnt environment with two panes. The left pane displays the pseudocode for the algorithm, and the right pane shows the execution process and results.

**Pseudocode (Left Pane):**

```
1 Algoritmo arreglo_pares_impar
2
3 Dimensionar num[10]
4
5 Definir i, sumaPares, sumaImpares, conteoPares, conteoImpares Como Entero
6 Definir sumaPosicionPares, sumaPosicionImpares Como Entero
7 Definir num, promedioPar, promedioImpar Como Real
8
9 sumaPares = 0
10 sumaImpares = 0
11 conteoPares = 0
12 conteoImpares = 0
13 sumaPosicionPares = 0
14 sumaPosicionImpares = 0
15
16 Para i = 0 Hasta 9 Hacer
17   Escribir "Ingrese un número entero: "
18   Leer num[i]
19   FinPara
20
21 Para i = 0 Hasta 9 Hacer
22
23   Si num[i] % 2 = 0 Entonces
24     sumaPares = sumaPares + num[i]
25     conteoPares = conteoPares + 1
26   Sino
27     sumaImpares = sumaImpares + num[i]
28     conteoImpares = conteoImpares + 1
29   FinSi
30
31   Si i % 2 = 0 Entonces
32     sumaPosicionPares = sumaPosicionPares + num[i]
33   Sino
34     sumaPosicionImpares = sumaPosicionImpares + num[i]
35   FinSi
36
37 FinPara
38
39 promedioPar = sumaPares / conteoPares
40 promedioImpar = sumaImpares / conteoImpares
41
42 Escribir "*"
43 Escribir "RESULTADOS"
44 Escribir "Promedio de números pares: ", promedioPar
45 Escribir "Promedio de números impares: ", promedioImpar
46 Escribir "Suma de posiciones pares del arreglo: ", sumaPosicionPares
47 Escribir "Suma de posiciones impares del arreglo: ", sumaPosicionImpares
48
49 FinAlgoritmo
```

**Execution and Results (Right Pane):**

Ingrese un número entero:  
> 23  
Ingrese un número entero:  
> 18  
Ingrese un número entero:  
> 21  
Ingrese un número entero:  
> 32  
Ingrese un número entero:  
> 25  
Ingrese un número entero:  
> 78  
Ingrese un número entero:  
> 65  
Ingrese un número entero:  
> 54  
Ingrese un número entero:  
> 45  
Ingrese un número entero:  
> 7666

RESULTADOS

Promedio de números pares: 1569.6  
Promedio de números impares: 35.8  
Suma de posiciones pares del arreglo: 179  
Suma de posiciones impares del arreglo: 7848  
\*\*\* Ejecución Finalizada. \*\*\*

No cerrar esta ventana  Siempre visible  Reiniciar



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

## 6. Actividad o ejercicio 6

### 6.1 Enunciado

Hacer un algoritmo que permita almacenar en un arreglo la cantidad de dinero que usted gasta día a día durante los 7 días de una semana, para mostrar por pantalla el promedio de gastos semanal. Recuerde hacer el análisis antes del código.

#### *solucion*

```
suma_dos_arreglos.psc nombre_paises.psc suma_dentro_arreglo.psc ICFES_Estudiantes.psc arreglo_pares_imparres.psc <sil
1 Algoritmo promedio_gastos_semanales
2
3   Dimensionar gastosSemanales[7]
4   Definir suma, promedio Como Real
5   Definir i, gastosSemanales Como Entero
6
7   suma ← 0
8
9   Para i ← 0 Hasta 6 Hacer
10    Escribir "Ingrese el gasto del día ", i, ":"
11    Leer gastosSemanales[i]
12    suma ← suma + gastosSemanales[i]
13  FinPara
14
15  promedio ← suma / 7
16
17  Escribir "El promedio de gastos semanales es: ", promedio, "pesos"
18
19 FinAlgoritmo
20
```

\*\*\* Ejecución Iniciada. \*\*\*  
Ingrese el gasto del dia 0:  
> 23000  
Ingrese el gasto del dia 1:  
> 45000  
Ingrese el gasto del dia 2:  
> 56000  
Ingrese el gasto del dia 3:  
> 87000  
Ingrese el gasto del dia 4:  
> 65000  
Ingrese el gasto del dia 5:  
> 21000  
Ingrese el gasto del dia 6:  
> 12000  
El promedio de gastos semanales es: 44142.85  
\*\*\* Ejecución Finalizada. \*\*\*

## 7. Actividad o ejercicio 7

### 7.1 Enunciado

Elaborar un algoritmo que lea una matriz de N x N e imprima la suma de los valores que están en la diagonal principal



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### solucion

The screenshot shows the PSeint environment with a script window titled "PSeint - Ejecutando proceso SUMA\_DIAGONAL...". The script is written in a pseudocode-like language:

```
1 Algoritmo suma_diagonal_principal
2
3 Definir N, i, j Como Entero
4 Definir suma, matriz Como Entero
5 suma = 0
6
7 Escribir "Ingrese el tamaño de la matriz NxN:"
8 Leer N
9
10 Dimensionar matriz[N, N]
11
12 Para i ← 0 Hasta N-1 Hacer
13   Para j ← 0 Hasta N-1 Hacer
14
15     Escribir "Ingrese valor para la posición [", i, ", ", j, "]: "
16     Leer Matriz[i,j]
17
18     Si i = j Entonces
19       Suma ← Suma + Matriz[i,j]
20     FinSi
21   FinPara
22
23 Escribir "La suma de la diagonal principal es: ", suma
24
25
26 FinAlgoritmo
27
28
29
```

The right pane shows the execution log:

```
*** Ejecución Iniciada ***
Ingrese el tamaño de la matriz NxN:
> 5
Ingrese valor para la posición [0,0]:
> 6
Ingrese valor para la posición [0,1]:
> 5
Ingrese valor para la posición [0,2]:
> 9
Ingrese valor para la posición [0,3]:
> 0
Ingrese valor para la posición [0,4]:
> 7
Ingrese valor para la posición [1,0]:
> 6
Ingrese valor para la posición [1,1]:
> 4
Ingrese valor para la posición [1,2]:
> 1
Ingrese valor para la posición [1,3]:
> 3
Ingrese valor para la posición [1,4]:
> 3
Ingrese valor para la posición [2,0]:
> 0
Ingrese valor para la posición [2,1]:
> 6
Ingrese valor para la posición [2,2]:
> 5
Ingrese valor para la posición [2,3]:
> 4
Ingrese valor para la posición [2,4]:
> 3
Ingrese valor para la posición [3,0]:
> 4
Ingrese valor para la posición [3,1]:
> 5
Ingrese valor para la posición [3,2]:
> 9
Ingrese valor para la posición [3,3]:
> 7
Ingrese valor para la posición [3,4]:
> 5
Ingrese valor para la posición [4,0]:
> 4
Ingrese valor para la posición [4,1]:
> 2
Ingrese valor para la posición [4,2]:
> 3
Ingrese valor para la posición [4,3]:
> 6
Ingrese valor para la posición [4,4]:
> 9
La suma de la diagonal principal es: 28
*** Ejecución Finalizada. ***
```

A context menu is open in the bottom right corner of the log window, with options: Copiar, Pegar, Ventana siempre visible, and Cerrar al finalizar la ejecución.

## 8. Actividad o ejercicio 8

### 8.1 Enunciado

Diseñe un algoritmo para organizar la venta de boletas en línea para el ingreso a un concierto, cada persona que desee ingresar debe presentar su cédula de ciudadanía, el algoritmo debe leer y agregar a un arreglo el número del documento de identidad. En el caso de que la identificación ya exista en el arreglo, debe mostrar un mensaje que rechace la venta de la boleta, en el momento que la venta se realice, el algoritmo debe permitir que lea la cantidad total de boletas que se compra (no mayor a 4) y reste de la cantidad disponible, cuando esta cantidad llegue a cero, muestre un mensaje y finalice. La solución debe incluir un menú de opciones que le permita realizar las tareas.



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

### *solucion*

```
1 Algoritmo VentaBoletas
2     Definir Cédulas Como Entero
3     Definir TotalBoletas, BoletasAComprar, Opcion Como Entero
4     Definir i, CantCédulas, Cédula Como Entero
5     Definir existe Com Logico
6     Dimension Cédulas[100]
7
8     TotalBoletas = 20
9     CantCédulas = 0
10
11    Repetir
12        Escribir "MENU VENTA DE BOLETAS"
13        Escribir "1. Comprar Boletas"
14        Escribir "2. Mostrar boletas disponibles"
15        Escribir "3. Salir"
16        Escribir "Seleccione una opción: "
17        Leer Opcion
18
19        Segun Opcion Hacer
20            :
21                Si TotalBoletas = 0 Entonces
22                    Escribir "No hay boletas disponibles. Venta finalizada."
23                    SiNo
24
25                    Existe = Falso
26
27                    Escribir "Ingrese número de cédula: "
28                    Leer Cedula
29
30                    Para i = 0 Hasta CantCédulas Hacer
31                        Si Cédulas[i] = Cédula Entonces
32                            Existe + Verdadero
33                            FinSi
34                            FinPara
35
36                        Si Existe = Verdadero Entonces
37                            Escribir " Venta rechazada: la cédula ya compró boletas."
38                        SiNo
39                            Escribir "¿Cuántas boletas desea comprar? (Máximo 4)"
40                            Leer BoletasAComprar
41
42                            Si BoletasAComprar > 4 Entonces
43                                Escribir "El máximo permitido es 4 boletas."
44                            SiNo
45                                Si BoletasAComprar ≤ TotalBoletas Entonces
46                                    Cédulas[CantCédulas] ← Cédula
47                                    CantCédulas + CantCédulas + 1
48                                    TotalBoletas ← TotalBoletas - BoletasAComprar
49
50                            Escribir "Compra exitosa"
```

```
PSELint - Ejecutando proceso VENTABOLETAS
> 1
Ingrese número de cédula:
> 1069348021
¿Cuántas boletas desea comprar? (Máximo 4)
> 3
Compra exitosa
Boletas restantes: 17
MENU VENTA DE BOLETAS
1. Comprar Boletas
2. Mostrar boletas disponibles
3. Salir
Seleccione una opción:
> 2
Boletas disponibles: 17
MENU VENTA DE BOLETAS
1. Comprar Boletas
2. Mostrar boletas disponibles
3. Salir
Seleccione una opción:
> 3
Gracias por usar el sistema.
*** Ejecución Finalizada. ***
```

## 9. Actividad o ejercicio 9

### 9.1 Enunciado

Hacer un algoritmo que lea una matriz de  $M \times N$  dimensiones con números enteros. Como resultado se debe generar otra matriz del mismo tamaño que va a contener en la misma posición una letra P si en la primera matriz el número es Par, y una letra I, si el número es impar en la primera matriz. Como resultado se deben imprimir las dos matrices.

### *solucion*



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

```
1 Algoritmo Par_Impar_Matriz
2   Definir M, N, i, j Como Entero
3   Definir MatrizNum Como Entero
4   Definir MatrizParImp Como Carácter
5   Escribir "Ingrese cantidad de filas (M): "
6   Leer M
7   Escribir "Ingrese cantidad de columnas (N): "
8   Leer N
9
10  Dimension MatrizNum[M,N]
11  Dimension MatrizParImp[M,N]
12
13  Para i ← 0 Hasta M-1 Hacer
14    Para j ← 0 Hasta N-1 Hacer
15      Escribir "Ingrese número para la posición [", i, ", ", j, "]: "
16      Leer MatrizNum[i,j]
17
18      Si MatrizNum[i,j] % 2 = 0 Entonces
19          MatrizParImp[i,j] ← "P"
20      SiNo
21          MatrizParImp[i,j] ← "I"
22      FinSi
23  FinPara
24
25  Escribir ""
26  Escribir "----- MATRIZ ORIGINAL -----"
27  Para i ← 0 Hasta M-1 Hacer
28    Para j ← 0 Hasta N-1 Hacer
```

```
PSeint - Ejecutando proceso PAR_IMPAR_MAT...
*** Ejecución Iniciada. ***
Ingrese cantidad de filas (M):
> 2
Ingrese cantidad de columnas (N):
> 4
Ingrese número para la posición [0,0]:
> 12
Ingrese número para la posición [0,1]:
> 4
Ingrese número para la posición [0,2]:
> 5
Ingrese número para la posición [0,3]:
> 7
Ingrese número para la posición [1,0]:
> 9
Ingrese número para la posición [1,1]:
> 2
Ingrese número para la posición [1,2]:
> 3
Ingrese número para la posición [1,3]:
> 5
----- MATRIZ ORIGINAL -----
12 4 5 7
9 2 3 5

----- MATRIZ PAR / IMPAR -----
P P I I
I P I I
*** Ejecución Finalizada. ***
```

## 10. Actividad o ejercicio 10

### 10.1 Enunciado

Hacer un algoritmo que permita almacenar la temperatura mínima y máxima de la ciudad de Popayán para una determinada semana. La información debe ser almacenada en una matriz.

El algoritmo debe permitir:

- Consultar que día se obtuvo la mayor temperatura. Debe informar mediante un mensaje así:

“El jueves la temperatura máxima fue de 36 grados”

- Consultar la temperatura mínima y máxima para determinado día.

*solución*



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

```
1 Algoritmo TemperaturasSemana
2     Definir i, DiaConsulta, Opcion Como Entero
3     Definir Temperaturas, MayorTemp Como Real
4     Definir DiaMayor Como Entero
5     Dimension Temperaturas[7,2]
6
7     Para i ← 0 Hasta 6 Hacer
8         Escribir "Día ", i, ":"
9         Escribir "Ingresar temperatura mínima: "
10        Leer Temperaturas[1,0]
11        Escribir "Ingresar temperatura máxima: "
12        Leer Temperaturas[1,1]
13    FinPara
14
15    Repetir
16        Escribir ""
17        Escribir "---- MENU ----"
18        Escribir "1. Consultar el día con mayor temperatura máxima"
19        Escribir "2. Consultar temperatura de un día"
20        Escribir "3. Salir"
21        Escribir "Seleccione una opción: "
22        Leer Opcion
23
24        Segun Opcion Hacer
25
26            1:
27                MayorTemp ← Temperaturas[0,1]
28                DiaMayor ← 0
29
30                Para i ← 0 Hasta 6 Hacer
31                    Si Temperaturas[i,1] > MayorTemp Entonces
32                        MayorTemp ← Temperaturas[1,1]
33                        DiaMayor ← i
34                    FinSi
35                FinPara
36
37            Escribir "El día ", ObtenerDia(DiaMayor), " la temperatura máxima fue de ", MayorTemp
38
39
```

```
Día 6:
Ingresar temperatura mínima: linea 22 instrucción 1
> 30
Ingresar temperatura máxima:
> 32
---- MENU ----
1. Consultar el día con mayor temperatura máxima
2. Consultar temperatura de un día
3. Salir
Seleccione una opción:
> 1
El día Lunes la temperatura máxima fue de 36 grados
---- MENU ----
1. Consultar el día con mayor temperatura máxima
2. Consultar temperatura de un día
3. Salir
Seleccione una opción:
> 2
Seleccione el día a consultar (0=Lunes ... 6=Domingo):
> 3
El día Jueves
Temperatura mínima: 21
Temperatura máxima: 30
---- MENU ----
1. Consultar el día con mayor temperatura máxima
2. Consultar temperatura de un día
3. Salir
>
```

## 11. Actividad o ejercicio 11

### 11.1 Enunciado

Diseñe un algoritmo para organizar la asignación diaria de citas de un médico especialista quien atiende máximo 8 citas al día, con duración de 60 minutos por cada paciente. El paciente que desea solicitar una cita debe ingresar su número de cedula de ciudadanía, el algoritmo lee y agrega la información a una matriz, por ejemplo:

Asignación de Citas:



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

Hora	Paciente
7 am	34467677
8 am	234566
9 am	3567888
10 am	244566
2 pm	244566778
3 pm	65433456
4 pm	3344563
5 pm	5768853

### *solución*

The screenshot shows a terminal window titled "PSELNT - Ejecutando proceso ASIGNACION\_CIT..." with the following content:

```
1 Algoritmo asignacion_citas
2
3 Definir horas Como Caracter
4 Definir i, cedulas, opcion, h, ced Como Entero
5 Dimensionar horas[8]
6 Dimensionar cedulas[8]
7
8 horas[0] ← "7 am"
9 horas[1] ← "8 am"
10 horas[2] ← "9 am"
11 horas[3] ← "10 am"
12 horas[4] ← "2 pm"
13 horas[5] ← "3 pm"
14 horas[6] ← "4 pm"
15 horas[7] ← "5 pm"
16
17 Para i ← 0 Hasta 7 Hacer
18 |   cedulas[i] ← 0
19 FinPara
20
21 Repetir
22
23 |   Escribir ""
24 |   Escribir "---- MENU DE CITAS ----"
25 |   Escribir "1. Asignar cita"
26 |   Escribir "2. Consultar todas las citas"
27 |   Escribir "3. Consultar cita por documento"
28 |   Escribir "4. Consultar cita por hora"
29 |   Escribir "5. Salir"
30 |   Leer opcion
31
32 Segun opcion Hacer
33
```

The application window displays the following information:

- Linea 30 instrucción 1
- Digite el número de la hora:  
> 10  
Hora inválida.
- MENU DE CITAS ----
  - 1. Asignar cita
  - 2. Consultar todas las citas
  - 3. Consultar cita por documento
  - 4. Consultar cita por hora
  - 5. Salir
- > 2
- Hora Paciente
  - 7 am Libre
  - 8 am Libre
  - 9 am Libre
  - 10 am Libre
  - 2 pm Libre
  - 3 pm Libre
  - 4 pm Libre
  - 5 pm Libre
- MENU DE CITAS ----
  - 1. Asignar cita
  - 2. Consultar todas las citas
  - 3. Consultar cita por documento
  - 4. Consultar cita por hora
  - 5. Salir
- >
- No cerrar esta ventana  Siempre visible Reiniciar

## 12. Actividad o ejercicio 12

### 12.1 Enunciado



## **PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS**

Hacer un algoritmo que permita manejar la información del número de habitantes de un conjunto residencial. El conjunto posee 3 torres, a su vez cada torre posee 5 pisos y cada piso 4 apartamentos.

Nota: la cantidad de habitantes por apartamento la pueden generar de manera aleatoria con valores entre 1 y 5 habitantes por apartamento.

El algoritmo debe mostrar un menú de opciones así:

## MENU CONJUNTO RESIDENCIAL

1. Agregar habitantes a cada apartamento
  2. Consultar número de habitantes por apartamento
  3. Consultar la cantidad total de habitantes del conjunto
  4. Consultar el promedio de habitantes por piso de cada torre
  5. Consultar el promedio de habitantes por torre
  6. Salir

Ingrese Opción:

*solucion*

```
1 arreglo_pares_imparres.psc promedio_gastos_semanales.psc suma_diagonal_principal.psc VentaBoletas.psc ParImpar_Mat
2 Algoritmo ConjuntoResidencial
3 Dimension hab[3,5,4]
4 Definir hab, op, t, p, a, total, suma, cont Como Entero
5
6 Para t=0 Hasta 2 Hacer
7   Para p=0 Hasta 4 Hacer
8     Para a=0 Hasta 3 Hacer
9       hab[t,p,a] = 0
10      FinPara
11    FinPara
12  FinPara
13
14 Repetir
15   Limpiar Pantalla
16   Escribir "MENU CONJUNTO RESIDENCIAL"
17   Escribir "1. Agregar habitantes a cada apartamento"
18   Escribir "2. Consultar habitantes por apartamento"
19   Escribir "3. Consultar el total de habitantes del conjunto"
20
21 2. Consultar habitantes por apartamento
22 3. Consultar el total de habitantes del conjunto
23
24 linea 46 instrucción 1
25
26 4. Consultar promedio de habitantes por piso de cada torre
27 5. Consultar promedio de habitantes por torre
28 6. Salir
29 > 2
30 Ingrese torre (0-2):
31 > 1
32 Ingrese piso (0-4):
33 > 1
34 Ingrese apartamento (0-3):
35 > 1
36 Habitantes:
37 3
```



## PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

PSeInt - Ejecutando proceso CONJUNTORESID...

MENU CONJUNTO RESIDENCIAL

```
1. Agregar habitantes a línea 72 instrucción 1
2. Consultar habitantes por apartamento
3. Consultar el total de habitantes del conjunto
4. Consultar promedio de habitantes por piso de cada torre
5. Consultar promedio de habitantes por torre
6. Salir
> 4
Torre 0
Piso 0 Promedio: 2.5
Piso 1 Promedio: 3.5
Piso 2 Promedio: 3.5
Piso 3 Promedio: 3.25
Piso 4 Promedio: 2.5
Torre 1
Piso 0 Promedio: 2.25
Piso 1 Promedio: 3.25
Piso 2 Promedio: 3.75
Piso 3 Promedio: 3.5
Piso 4 Promedio: 2.25
Torre 2
Piso 0 Promedio: 4
Piso 1 Promedio: 2.75
Piso 2 Promedio: 3.75
Piso 3 Promedio: 3
Piso 4 Promedio: 3.75
```

1 Algoritmo ConjuntoResidencial

2     Dimension hab[3,5,4]

3     Definir hab, op, t, p, a, total, suma, cont Como Entero

4

5     Para t=0 Hasta 2 Hacer

6         Para p=0 Hasta 4 Hacer

7             Para a=0 Hasta 3 Hacer

8                 hab[t,p,a] = 0

9             FinPara

10         FinPara

11     FinPara

12

13     Repetir

14         Limpiar Pantalla

15         Escribir "MENU CONJUNTO RESIDENCIAL"

16         Escribir "1. Agregar habitantes a cada apartamento"

17         Escribir "2. Consultar habitantes por apartamento"

18         Escribir "3. Consultar el total de habitantes del conjunto"

19         Escribir "4. Consultar promedio de habitantes por piso de cada torre"

20         Escribir "5. Consultar promedio de habitantes por torre"

21         Escribir "6. Salir"

22         Leer op

23

24     Segun op Hacer

25

26         1:
27             Para t=0 Hasta 2 Hacer
28                 Para p=0 Hasta 4 Hacer
29                     Para a=0 Hasta 3 Hacer