

Ejercicios de Programación (valor 5pts)

Materia: Introducción a la Programación

Nombre del estudiante: José Armando Alcántara Zabala

Fecha: 06-10-2025

Definiciones Básicas

1. Condicionales (Si... Entonces... Sino):

Permiten tomar decisiones en el programa dependiendo de si una condición es verdadera o falsa.

2. Bucle Mientras:

Se repite un bloque de instrucciones mientras la condición indicada sea verdadera.

3. Bucles Repetitivos (Para, Repetir Hasta):

Se usan para ejecutar instrucciones varias veces de manera controlada, con un inicio y un fin.

Ejercicios con Condicionales

- Pide la edad al usuario. Si es mayor o igual a 18 muestra 'Eres mayor de edad', sino 'Eres menor de edad'.

The screenshot shows the PSeInt IDE interface with the following details:

- Toolbar:** Archivo, Editar, Configurar, Ejecutar, Ayuda.
- Operadores y Funciones:** A list of operators including arithmetic, logical, relational, and mathematical functions.
- Code Editor:** Displays the following pseudocode:

```
1 Algoritmo Mayor_O_Menor_Edad
2   Definir edad Como Entero
3   Escribir "Introduzca su edad:"
4   Leer edad
5
6   Si edad ≥ 18 Entonces
7     Escribir "Eres mayor de edad"
8   SiNo
9     Escribir "Eres menor de edad"
10  Fin Si
11
12 FinAlgoritmo
```
- Output Window:** Shows the execution results:

```
*** Ejecución Iniciada. ***
Introduzca su edad:
> 27
Eres mayor de edad
*** Ejecución Finalizada. ***
```
- Comandos Panel:** A sidebar with icons for various control structures: Ejecutar paso a paso, Dato1, Asign, Si-Ent, Segui, Mientras, Repetir, Para, and Func.
- Bottom Status Bar:** La ejecución ha finalizado sin errores.

2. Sigue el algoritmo para determinar si un número es positivo, negativo o cero.

```

1. Algoritmo Numero_Positivo,Negativo
2. Definir num Como Real
3. Escribir "Introduzca un numero:"
4. Leer num
5. Si num <= 0 Entonces
6.   Escribir "El numero es positivo"
7. SiNo
8.   Si num < 0 Entonces
9.     Escribir "El numero es negativo"
10.  SiNo
11.    Escribir "El numero es cero"
12. Fin Si
13.
14. FinAlgoritmo
15.
16.

```

The screenshot shows the Psient software interface. On the left is the code editor with the above pseudocode. On the right is the command palette with various icons for operations like If, While, For, and Functions. A small window titled 'Psient - Ejecutando proceso NUMERO_POSITIVO...' is open, showing the execution process and the output: 'El numero es positivo'. At the bottom left, a message says 'La ejecución ha finalizado sin errores.'

3. Pide un número e indica si es par o impar.

```

1. Algoritmo Numero_Par_Impar
2. Definir num Como Entero
3. Escribir "Introduzca un numero:"
4. Leer num
5. Si num MOD 2 = 0 Entonces
6.   Escribir "El numero es par"
7. SiNo
8.   Escribir "El numero es impar"
9. Fin Si
10.
11.

```

The screenshot shows the Psient software interface. On the left is the code editor with the above pseudocode. On the right is the command palette. A small window titled 'Psient - Ejecutando proceso NUMERO_PAR_IMPAR...' is open, showing the execution process and the output: 'El numero es impar'. At the bottom left, a message says 'La ejecución ha finalizado sin errores.'

4. Introduce una nota (0-100). Muestra 'Aprobado con A' si es >=90, 'Aprobado' si es >=70 y 'Reprobado' en caso contrario.

```

1. Algoritmo CALIFICACION
2. Definir nota Como Entero
3. Escribir "Introduzca la nota"
4. Leer nota
5. Si nota >= 90 Entonces
6.   Escribir "Aprobado con A"
7. SiNo
8.   Si nota >= 70 Entonces
9.     Escribir "Aprobado"
10.  SiNo
11.    Escribir "Reprobado"
12. Fin Si
13.
14.
15. FinAlgoritmo
16.

```

The screenshot shows the Psient software interface. On the left is the code editor with the above pseudocode. On the right is the command palette. A small window titled 'Psient - Ejecutando proceso CALIFICACION...' is open, showing the execution process and the output: 'Aprobado con A'. At the bottom left, a message says 'La ejecución ha finalizado sin errores.'

5. Ingresa el monto de una compra. Si es mayor a 500 aplica un 10% de descuento, sino paga precio normal.

The screenshot shows the PSelint IDE interface with the following details:

- Toolbar:** Includes standard file operations (New, Open, Save, Print, etc.) and a search bar.
- Menu Bar:** Archivo, Editar, Configurar, Ejecutar, Ayuda.
- Left Sidebar:**
 - Operadores y Funciones:** Shows operators like + (suma), - (resta), * (multiplicación), / (división), ! (potencia), MOD (resto/modulo), Op. Lógicos (Y (conjunción), O (disyunción), NO (negación)), Op. Relacionales (= (igual), ≠ (distinto), < (menor), ≤ (menor o igual), > (mayor), ≥ (mayor o igual)), and Func. Matemáticas (abs (valor absoluto), trunc (valor truncado), redon (valor redondeado), raiz (raíz cuadrada), sen (seno), cos (coseno)).
 - Llista de Variables:** A list of variables used in the current session.
 - Errores de Sintaxis:** A section for syntax errors.
 - ASIGNAR:** A section for assignments.
- Code Editor:**

```

1 Algoritmo Descuento
2   Definir monto, total Como Real
3   Escribir "Introduzca el monto de la compra"
4   Leer monto
5   Si monto > 500 Entonces
6     total <- monto * 0.9
7     Escribir "Se aplica un 10% de descuento. Total a pagar: ", total
8   SiNo
9     Escribir "No aplica descuento. Total a pagar: ", monto
10 Fin Si
11 FinAlgoritmo
12

```
- Output Window:**

```

*** Ejecución Iniciada. ***
Introduzca el monto de la compra
> 750
Se aplica un 10% de descuento. Total a pagar:
675
*** Ejecución Finalizada. ***

```
- Right Sidebar:** A panel titled "Comandos" containing various control blocks such as Dato, Leer, Si-Entonces, Seguir, Mientras, Repetir, Para, and Función.

La ejecución ha finalizado sin errores.

Ejercicios con Bucle Mientras

1. Muestra los números del 1 al 10 usando mientras.

The screenshot shows the PSeInt IDE interface with the code editor containing the following pseudocode:

```
1 Algoritmo Numero_de_1_a_10
2   Definir i Como Entero
3   i <- 1
4   Mientras i ≤ 10 Hacer
5     Escribir i
6     i <- i + 1
7   Fin Mientras
8
9 FinAlgoritmo
```

The execution window displays the output: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. The status bar at the bottom of the execution window says "La ejecución ha finalizado sin errores." (The execution has ended without errors).

2. Pide números al usuario y suma todos hasta que escriba 0.

The screenshot shows the PSeInt IDE interface with the code editor containing the following pseudocode:

```
1 Algoritmo Sumar_hasta_cero
2   Definir num, suma Como Real
3   suma <- 0
4
5   Escribir "Ingrese un número (0 para terminar): "
6   Leer num
7
8   Mientras num ≠ 0 Hacer
9     suma <- suma + num
10    Escribir "Ingrese otro número (0 para terminar): "
11    Leer num
12  FinMientras
13
14  Escribir "La suma total es: ", suma
15
16 FinAlgoritmo
```

The execution window displays the interaction with the user: "Ingrese un número (0 para terminar):", followed by several inputs like > 50, > 60, > 0, and the final output "La suma total es: 110". The status bar at the bottom of the execution window says "La ejecución ha finalizado sin errores." (The execution has ended without errors).

3. Adivina el número secreto (ejemplo: 7).

The screenshot shows the PSeInt IDE interface with the code editor containing the following pseudocode:

```
1 Algoritmo Adivinar_numero
2   Definir numSecreto, intento Como Entero
3   numSecreto <- 7
4
5   Escribir "Adivina el número secreto (entre 1 y 10). "
6   Leer intento
7
8   Mientras intento ≠ numSecreto Hacer
9     Escribir "Incorrecto, intenta otra vez: "
10    Leer intento
11  FinMientras
12
13  Escribir "¡Correcto! El número secreto era ", numSecreto
14
15 FinAlgoritmo
```

The execution window displays the interaction with the user: "Adivina el número secreto (entre 1 y 10).", followed by several inputs like > 7, > 7, and the final output "¡Correcto! El número secreto era 7". The status bar at the bottom of the execution window says "La ejecución ha finalizado sin errores." (The execution has ended without errors).

4. Valida una contraseña. Mientras no sea '1234', vuelve a pedirla.

The screenshot shows the PSeInt IDE interface. On the left, the code editor displays an algorithm to validate a password:

```

1 Algoritmo Validar_Contraseña
2   Definir contraseña Como Cadena
3
4   Escribir "Ingrese la contraseña: "
5   Leer contraseña
6
7   Mientras contraseña ≠ "1234" Hacer
8     Escribir "Contraseña incorrecta. Intente de nuevo: "
9     Leer contraseña
10    FinMientras
11
12   Escribir "¡Acceso autorizado!"
13 FinAlgoritmo
14

```

The execution window on the right shows the process starting and receiving the input "1234". It then outputs "¡Acceso autorizado!" and "*** Ejecución Finalizada. ***".

5. Crea un contador regresivo desde un número dado hasta 1.

The screenshot shows the PSeInt IDE interface. On the left, the code editor displays an algorithm to create a recursive counter:

```

1 Algoritmo Contador_Regresivo
2   Definir num Como Entero
3
4   Escribir "Ingrese un número para comenzar la cuenta regresiva: "
5   Leer num
6
7   Mientras num ≥ 1 Hacer
8     Escribir num
9     num ← num - 1
10    FinMientras
11
12   Escribir "¡GOOO!"
13
14 FinAlgoritmo
15

```

The execution window on the right shows the process starting and receiving the input "3". It then outputs "3", "2", "1", and "¡GOOO!", followed by "*** Ejecución Finalizada. ***".

Ejercicios con Bucles Repetitivos

1. Muestra la tabla de multiplicar de un número ingresado por el usuario.

The screenshot shows the PSeInt IDE interface. On the left, the code editor displays an algorithm named 'Tabla_Multiplicar' with the following pseudocode:

```
1 Algoritmo Tabla_Multiplicar
2   Definir num, i, resultado Como Entero
3
4   Escribir "Ingrese un número para ver su tabla de multiplicar:"
5   Leer num
6
7   Para i ← 1 Hasta 10 Con Paso 1 Hacer
8     resultado ← num * i
9     Escribir num, " x ", i, " = ", resultado
10    FinPara
11 FinAlgoritmo
```

To the right, the execution window shows the output of the program. It starts with 'Ingresar un número para ver su tabla de multiplicar:' followed by a series of multiplication results for the number 5:

```
> 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```

The window also includes a toolbar at the top and a status bar at the bottom indicating 'La ejecución ha finalizado sin errores'.

2. Pide 10 números y calcula la suma total.

The screenshot shows the PSeInt IDE interface again. The code editor contains an algorithm named 'Suma_10_numeros' with the following pseudocode:

```
1 Algoritmo Suma_10_numeros
2   Definir num, suma, i Como Real
3   suma ← 0
4
5   Para i ← 1 Hasta 10 Con Paso 1 Hacer
6     Escribir "Ingrese el número ", i, ": "
7     Leer num
8     suma ← suma + num
9   FinPara
10
11  Escribir "La suma total es: ", suma
12
13 FinAlgoritmo
```

The execution window shows the program prompting for 10 numbers and calculating their sum:

```
Ingresar el número 1:
> 75
Ingresar el número 2:
> 42
Ingresar el número 3:
> 13
Ingresar el número 4:
> 62
Ingresar el número 5:
> 85
Ingresar el número 6:
> 93
Ingresar el número 7:
> 49
Ingresar el número 8:
> 62
Ingresar el número 9:
> 18
Ingresar el número 10:
> 23
La suma total es: 522
```

The status bar at the bottom of the execution window also indicates 'La ejecución ha finalizado sin errores'.

3. Calcula el factorial de un número.

The screenshot shows the PSeInt IDE interface. On the left is the code editor with the following pseudocode:

```

1 Algoritmo Factorial
2   Definir num, i, fact Como Entero
3
4   Escribir "Ingrese un número (entero no negativo): "
5   Leer num
6
7   Si num < 0 Entonces
8     | Escribir "Error: el factorial no está definido para números negativos."
9   Sino
10    | fact ← 1
11    | Para i ← 1 Hasta num Con Paso 1 Hacer
12    |   | fact ← fact * i
13    | FinPara
14    | Escribir "El factorial de ", num, " es: ", fact
15
16 FinAlgoritmo
17

```

To the right is the execution window titled "PSeInt - Ejecutando proceso FACTORIAL". It displays the following output:

```

*** Ejecución Iniciada. ***
Ingrese un número (entero no negativo):
> 8
El factorial de 8 es: 40320
*** Ejecución Finalizada. ***

```

The status bar at the bottom of the execution window says "La ejecución ha finalizado sin errores".

4. Muestra todos los números pares entre 1 y 50.

The screenshot shows the PSeInt IDE interface. On the left is the code editor with the following pseudocode:

```

1 Algoritmo Numero_Pares_1_50
2   Definir i Como Entero
3
4   Escribir "Números pares entre 1 y 50:"
5
6   Para i ← 2 Hasta 50 Con Paso 2 Hacer
7     | Escribir i
8   FinPara
9 FinAlgoritmo
10

```

To the right is the execution window titled "PSeInt - Ejecutando proceso NUMERO_PARES_1_50". It displays the following output:

```

2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
*** Ejecución Finalizada. ***

```

The status bar at the bottom of the execution window says "La ejecución ha finalizado sin errores".

5. Pide 5 notas, calcula la suma y el promedio final.

The screenshot shows the PSelint IDE interface. On the left is the pseudocode editor with the following code:

```
1 Algoritmo Promedio_notas
2   Definir nota, suma, promedio Como Real
3   Definir i Como Entero
4   suma ← 0
5
6   Para i ← 1 Hasta 5 Con Paso 1 Hacer
7     Escribir "Ingrese la nota ", i, ";"
8     Leer nota
9     suma ← suma + nota
10  FinPara
11
12  promedio ← suma / 5
13
14  Escribir "La suma de las notas es: ", suma
15  Escribir "El promedio final es: ", promedio
16
17 FinAlgoritmo
18
```

To the right of the editor is a command window showing the execution process:

```
PSelint - Ejecutando proceso PROMEDIO_NOTAS
*** Ejecución Iniciada. ***
Ingresé la nota 1:
> 95
Ingresé la nota 2:
> 96
Ingresé la nota 3:
> 98
Ingresé la nota 4:
> 99
Ingresé la nota 5:
> 92
La suma de las notas es: 480
El promedio final es: 96
*** Ejecución Finalizada. ***
```

Below the command window is a graphical palette containing various control structures like loops and conditionals.