



Universidad de las Fuerzas Armadas ESPE

Departamento: Ciencias de la computación

Carrera: Ingeniería en Electricidad y Automatización

Taller académico N°: 3

1. Información General

Asignatura: Fundamentos de Programación

Apellidos y nombres de los estudiantes: Ana Ulloa, Jordy Cevallos, Bryan Miguitama

NRC: 20823

Fecha de realización: 08/05/2025

2. Objetivo del Taller y Desarrollo

Objetivo del Taller:

Realizar tabla, algoritmo y prueba de escritorio.

Desarrollo:

Ejercicio 1

Problema 1.1 Media de dos valores.
Desarrolle un programa que lea dos números reales del teclado e imprima su media.
La solución consiste en una secuencia de instrucciones, tal y como se refleja en el diagrama de flujo y en el código C. Primero se lee un valor real y se guarda en la variable *x*; a continuación se lee y almacena el segundo valor en *y*. Finalmente la variable *res* recibe la media de ambos valores y se muestra por pantalla.

TABLA

Objeto	Nombre	Valor	Tipo
Dato 1	x, z	variable	real
Dato 2	MA	constante	real
Dato 3	Resultado	variable	real

ALGORITMO

Inicio algoritmo Mediadedosvalores

Definir x, z, resultado como real

Escribir "Ingrese los dos números reales:"

Leer x

Leer z



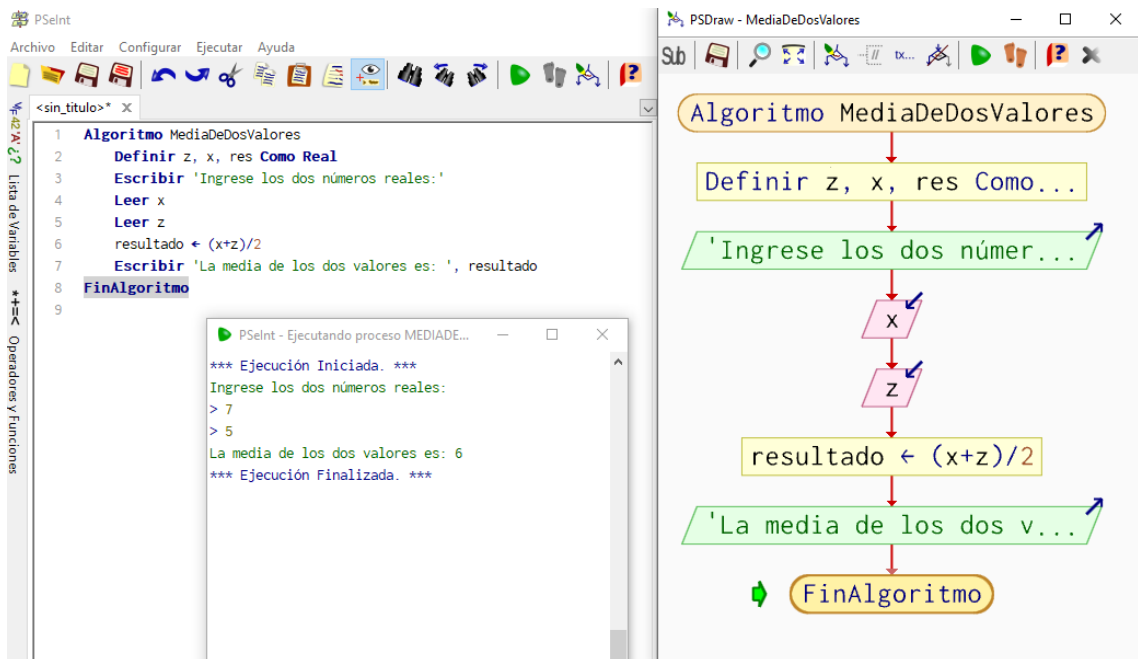
resultado <- (x + z) / 2

Escribir "La media de los dos valores es: ", resultado

FinAlgoritmo

PRUEBA DE ESCRITORIO

Valores	Fórmula	Proceso	Resultado
x, z	x, z/2		
7, 5	7+5/2	12/2	6



Ejercicio 2

Problema 1.2 Valor absoluto de x al cubo.

Desarrolle un programa que lea un número real x y escriba por pantalla $|x|^3$.

En el diagrama de flujo se hace uso de un módulo para calcular el valor absoluto, aunque no es implementado como tal a nivel de código C.

TABLA

Objeto	Nombre	Valor	Tipo
Dato 1	x	variable	real
Dato 2	Resultado	variable	real

ALGORITMO

Algoritmo valorabsolutocubo

Definir x, resultado como real



Escribir "Ingrese un número real x:"

Leer x

resultado <- abs (x ↑ 3)

Escribir "El valor absoluto al cubo de x es: ", resultado

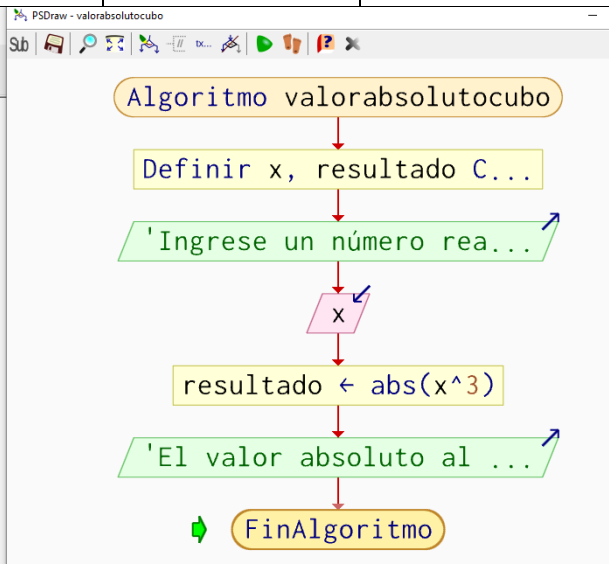
FinAlgoritmo

PRUEBA DE ESCRITORIO

Valores	Fórmula	Proceso	Resultado
x	x^3	$x*x*x$	x
3	3^3	$3*3*3$	27

```
1 Algoritmo valorabsolutocubo
2 Definir x, resultado Como Real
3 Escribir 'Ingrese un número real x:'
4 Leer x
5 resultado ← abs(x^3)
6 Escribir 'El valor absoluto al cubo de x es: ', resultado
7 FinAlgoritmo
```

*** Ejecución Iniciada. ***
Ingrese un número real x:
> 3
El valor absoluto al cubo de x es: 27
*** Ejecución Finalizada. ***



Ejercicio 3

Problema 1.3 Divisible.
Desarrolle un programa que lea dos números enteros por teclado y determine si el primero de ellos es divisible por el segundo. Se mostrará por pantalla el resultado.
La solución utiliza el operador módulo %, que devuelve el resto de la división entera entre números enteros. Si x es divisible entre y , el resto $x\%y$ debe ser 0.

TABLA

Objeto	nombre	valor	tipo
Dato1	Num1	Variable	Entero
Dato2	Num2	Variable	Entero
Dato3	Resultado	Variable	Cadena

ALGORITMO

Algoritmo divisible

Definir Num1, Num2 Como Entero



Escribir "Ingrese el primer número:"

Leer Num1

Escribir "Ingrese el segundo número:"

Leer Num2

Si Num2 \neq 0 Entonces

Si Num1 % Num2 = 0 Entonces

Escribir "El número es divisible"

Sino

Escribir "El número NO es divisible"

FinSi

Sino

Escribir "No se puede dividir por cero"

FinSi

FinAlgoritmo

PRUEBA DE ESCRITORIO

Valores	Fórmula	Proceso	Resultado
Num1 = 12, Num2 = 4	Num1 % Num2 =	12 % 4=0	El número es divisible=0

<sin_titulo>* X <sin_titulo>* <sin_titulo>*

```
1 Algoritmo divisible
2 Definir Num1, Num2 Como Entero
3 Escribir 'Ingrese el primer número:'
4 Leer Num1
5 Escribir 'Ingrese el segundo número:'
6 Leer Num2
7 Si Num2≠0 Entonces
8     Si Num1 MOD Num2=0 Entonces
9         Escribir 'El número es divisible'
10        SiNo
11            Escribir 'El número NO es divisible'
12        FinSi
13    SiNo
14        Escribir 'No se puede dividir por cero'
15    FinSi
16 FinAlgoritmo
17
```

PSDraw - divisible

```
graph TD
    Start([Algoritmo divisible]) --> Def[Definir Num1, Num2 Como Entero]
    Def --> In1[/Ingrese el primer número/]
    In1 --> Num1[Num1]
    Num1 --> In2[/Ingrese el segundo número/]
    In2 --> Num2[Num2]
    Num2 --> Dec1{Num2 ≠ 0}
    Dec1 -- F --> Out1[/No se puede dividir p.../]
    Dec1 -- V --> Dec2{Num1 MOD Num2 = 0}
    Dec2 -- F --> Out2[/El número NO es divis.../]
    Dec2 -- V --> Out3[/El número es divisible/]
    Out1 --> End([FinAlgoritmo])
    Out2 --> End
    Out3 --> End
```

PSInt - Ejecutando proceso DI...

```
*** Ejecución Iniciada. ***
Ingrese el primer número:
> 12
Ingrese el segundo número:
> 4
El número es divisible
*** Ejecución Finalizada. ***
```



Ejercicio 4

Problema 1.4 Intervalo.

Desarrolle un programa que lea un número real del teclado y determine si pertenece al intervalo $(0, 10]$, indicando por pantalla el resultado.

El número x introducido ha de cumplir $x > 0$ y $x \leq 10$ (ambas condiciones a la vez) para pertenecer al intervalo. Para establecer estas dos condiciones se debe utilizar un operador lógico, en este caso, el operador Y, en diagrama de flujo, y $\&\&$, en C. Es decir, hay que evitar caer en la tentación de escribir $0 < x \leq 10$.

TABLA

Objeto	Nombre	Valor	Tipo
M 1	Lector	variable	real
M 2	Variable X	variable	real
M 3	Comprobador	constante	real
M4	Resultado	variable	real

ALGORITMO

Algoritmo intervalo

Leer un número real

Guardar en variable X

Verificar si $(x > 0)$ and $(x \leq 10)$

Si $(x > 0)$ and $(x \leq 10)$, Escribir: "El número pertenece al intervalo (0;10)"

Sino $(x > 0)$ and $(x \leq 10)$, Escribir: " El número pertenece al intervalo (0;10)"

FinAlgoritmo

PRUEBA DE ESCRITORIO

Lector	Variable X	Comprobador	Resultado
4	4	$(4 > 0)$ y $(4 \leq 10)$	El número pertenece al intervalo (0;10)



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

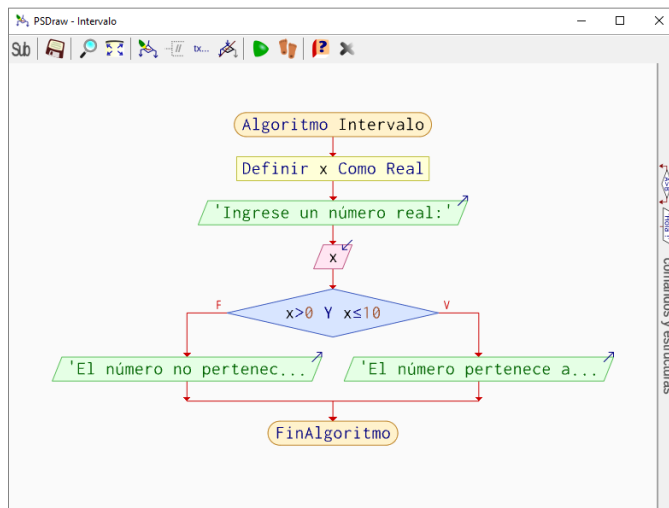


```
Archivo  Editar  Configurar  Ejecutar  Ayuda
<sin_titulo>*  X
1  Algoritmo Intervalo
2      Definir x Como Real
3
4      Escribir "Ingrese un número real:"
5      Leer x
6
7      Si x > 0 Y x ≤ 10 Entonces
8          Escribir "El número pertenece al intervalo (0, 10]"
9      Sino
10         Escribir "El número no pertenece al intervalo (0, 10]"
11      FinSi
12  FinAlgoritmo
13

PSeint - Ejecutando proceso INTERVALO
*** Ejecución Iniciada. ***
Ingrese un número real:
> 2
El número pertenece al intervalo (0, 10]
*** Ejecución Finalizada. ***

<sin_titulo>*  X
1  Algoritmo Intervalo
2      Definir x Como Real
3
4      Escribir "Ingrese un número real:"
5      Leer x
6
7      Si x > 0 Y x ≤ 10 Entonces
8          Escribir "El número pertenece al intervalo (0, 10]"
9      Sino
10         Escribir "El número no pertenece al intervalo (0, 10]"
11      FinSi
12  FinAlgoritmo
13

PSeint - Ejecutando proceso INTERVALO
*** Ejecución Iniciada. ***
Ingrese un número real:
> 32
El número no pertenece al intervalo (0, 10]
*** Ejecución Finalizada. ***
```



Lenguaje de Alto nivel

Tabla

Objeto	nombre	valor	tipo
m1	lector	Variable	Real
m2	Variable x	Variable	Real
m3	Comproedor	constante	Real
m4	Resultado	variable	Real

Algoritmo: Intervalo

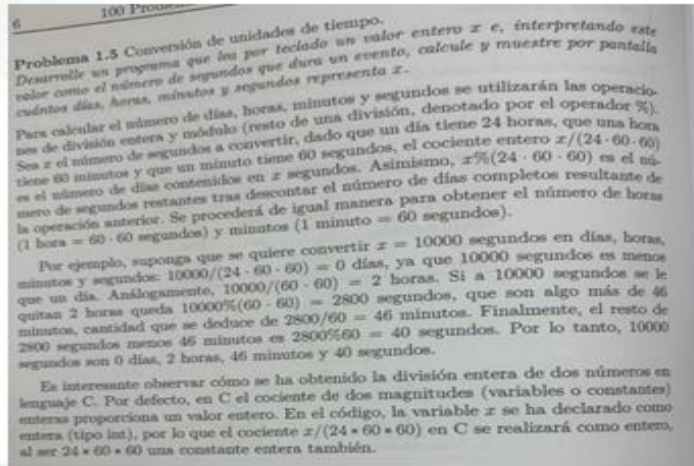
- Inicio
- Leer un número real
- Guardar en variable x y resultado
- Verificar si $(x > 0) \text{ and } (x \leq 10)$
 - Si $(x > 0) \text{ and } (x \leq 10)$, Escribir: "El número pertenece al intervalo (0, 10]"
 - No $(x > 0) \text{ and } (x \leq 10)$, Escribir: "El número no pertenece al intervalo (0, 10]"
- Fin

Prueba de escritorio

lector	Variable x	Comproedor	Resultado
4	4	$(4 > 0) \text{ and } (4 \leq 10)$ Si: El número pertenece al intervalo (0, 10]	
12	12	$(12 > 0) \text{ and } (12 \leq 10)$ No: El número no pertenece al intervalo (0, 10]	



Ejercicio 5



TABLA

Objeto	Nombre	Valor	Tipo
Dato 1	x	variable	entero
Dato 2	Días, horas, minutos, segundos, resto	variable	entero
Dato 3	resultado	variable	entero

ALGORITMO

Algoritmo CalculoUnidadesTiempo

Definir x, resi, dias, horas, minu, segund Como Real

Escribir 'Ingrese un numero'

Leer x

dias <- x/(24*60*60)

resi <- x MOD (24*60*60)

horas <- resi/(60*60)

resi <- resi MOD (60*60)

minu <- resi/(60)

segund <- resi MOD 60

Escribir 'El valor del numero ', x, ' equivale a ', dias, ' dias, ', horas, ' horas, ', minu, ' minutos y ', segund, ' segundos'

FinAlgoritmo



PRUEBA DE ESCRITORIO

PSInt - Ejecutando proceso CALCULOINIDA...

```
*** Ejecución Iniciada. ***  
Ingrese un numero  
> 849520  
El valor del numero 849520 equivale a 9.832407  
4074 dias, 19.9777777778 horas, 58.6666666667  
minutos y 40 segundos  
*** Ejecución Finalizada. ***
```

PSDraw - calculoUnidadesTiempo

```
Algoritmo calculoUnidadesTiempo  
Definir x, resi, dias, horas, minu, segund Como Real  
Escribir 'Ingrese un numero'  
Leer x  
dias ← x/(24*60*60)  
resi ← x MOD (24*60*60)  
horas ← resi/(60*60)  
resi ← resi MOD (60*60)  
minu ← resi/(60)  
segund ← resi MOD 60  
Escribir 'El valor del numero ', x, ' equivale a ', dias  
FinAlgoritmo
```

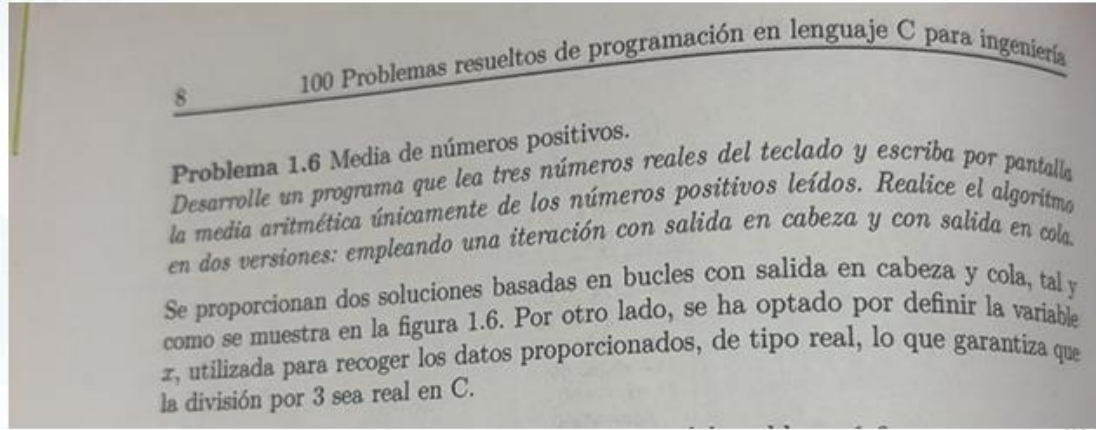
Flowchart:

```
graph TD  
A([Algoritmo calculoUnidadesTiempo]) --> B[Definir x, resi, dias,...]  
B --> C[/Ingrese un numero/]  
C --> D[x]  
D --> E[dias ← x/(24*60*60)]  
E --> F[resi ← x MOD (24*60*60)]  
F --> G[horas ← resi/(60*60)]  
G --> H[resi ← resi MOD (60*60)]  
H --> I[minu ← resi/(60)]  
I --> J[segund ← resi MOD 60]  
J --> K[/El valor del numero '...'/]  
K --> L([FinAlgoritmo])
```

x	dias	resi	horas	resi	minu	segundo	pantalla
451676	5	19676	5	1676	27	56	5 dias 5 horas 27 minutos y 56 segundos.
849520	9	71920	19	3520	58	40	9 dias 19 horas 58 minutos 40 segundos.



Ejercicio 6



TABLA

Objeto	Nombre	Valor	Tipo
Dato 1	x, p, z	variable	real
Dato 2	MA	constante	real
Dato 3	Resultado	variable	real

ALGORITMO

Algoritmo media

Definir x,p,z resultado como real

Escribir “Escriba 3 números reales:”, como real

Leer x

Leer p

Leer z

resultado <- ((x+p+z) ↑ 3)

Escribir “la media de de x+p+z: ", resultado

FinAlgoritmo

PRUEBA DE ESCRITORIO

Valores	Fórmula	Proceso	Resultado
x,p,z	$X+z+p/3$		
1,7,2	$1+7+2/3$	10/3	3.333



Ejercicio 7

Problema 1.7 Máximo, mínimo y media de alturas.

Desarrolle un programa que lea por teclado un número indeterminado de alturas y que deje de leer cuando se haya introducido una negativa. El programa debe calcular e imprimir la media, el máximo y el mínimo de las alturas introducidas.

Para resolver este problema es necesario inicializar las variables que almacenan el máximo y el mínimo con un valor válido. Dado que las alturas deben ser todas positivas, el valor máximo inicial podría ser cero, actualizándose con la primera altura introducida. El valor inicial de la altura mínima ha de ser mayor que cualquier valor que se pueda introducir. Esto puede resultar problemático si no estamos seguros del contexto en el que se va a usar el programa. Podría suceder que fijáramos como altura mínima un valor tan bajo que todas las medidas introducidas fuesen superiores a este valor inicial. En este caso, la altura mínima no se actualizaría y el resultado sería erróneo. Una alternativa más segura es tomar como valor inicial para el máximo y mínimo el primer valor introducido por teclado, así no es necesario realizar ninguna hipótesis sobre la naturaleza de los datos que hay que introducir. Esta ha sido la solución adoptada en este ejercicio.

TABLA

Objeto	Nombre	Valor	Tipo
Dato1	Altura1	Variable	Real
Dato2	Altura2	Variable	Real
Dato3	Máximo	Variable	Real
Dato4	Mínimo	Variable	Real
Dato5	Suma	Variable	Real
Dato6	Contador	Variable	Entero
Dato7	Media	Variable	Real

ALGORITMO

Algoritmo MaxMinMedia

Definir altura, maximo, minimo, suma, contador Como Real

suma <- 0

contador <- 0

Leer altura

maximo <- altura

minimo <- altura

Mientras altura >= 0 Hacer



suma <- suma + altura

contador <- contador + 1

Si altura > maximo Entonces

maximo <- altura

FinSi

Si altura < minimo Entonces

minimo <- altura

FinSi

Leer altura

FinMientras

media <- suma / contador

Escribir "Media: ", media, " Maximo: ", maximo, " Minimo: ", minimo

FinAlgoritmo

PRUEBA DE ESCRITORIO

Valores	Fórmula	Proceso	Resultado
AL1,AL2			
160 175	maximo ← altura minimo ← altura suma ← suma + altura media ← suma / contador	maximo ← 160 minimo ← 160 suma ← 160 + 175 = 335 contador ← 2 media ← 335 / 2 = 167.5	maximo = 175, minimo = 160, media = 335 / 2 = 167.5

```

1  Algoritmo MaxMinMedia
2  Definir altura, maximo, minimo, suma, contador Como Real
3  suma ← 0
4  contador ← 0
5  Leer altura
6  maximo ← altura
7  minimo ← altura
8  Mientras altura ≥ 0 Hacer
9      suma ← suma + altura
10     contador ← contador + 1
11     Si altura > maximo Entonces
12         maximo ← altura
13     FinSi
14     Si altura < minimo Entonces
15         minimo ← altura
16     FinSi
17     Leer altura
18 FinMientras
19 media ← suma / contador
20 Escribir 'Media: ', media, ' Maximo: ', maximo, ' Minimo: ', minimo
21 FinAlgoritmo
22

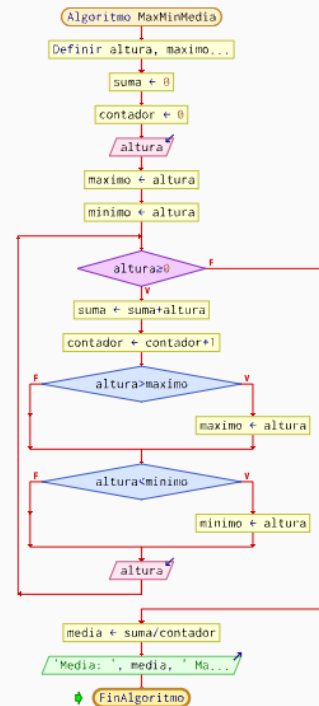
```

PSeInt - Ejecutando pr...

```

*** Ejecución Iniciada. ***
> 160
> 175
> -1
Media: 167.5 Maximo: 175 Minimo: 160
*** Ejecución Finalizada. ***

```



Ejercicio 8

Problema 1.8 Raíz cuadrada entera.

Desarrolle un programa que calcule la raíz cuadrada entera por defecto de un número N positivo dado, que se define como el mayor número entero $R > 0$, tal que $R^2 \leq N$.

Dada una variable entera $i = 1$, se itera incrementando unitariamente su valor mientras $i^2 \leq N$. La aproximación por defecto de la raíz es $R = i - 1$.

TABLA

Objeto	Nombre	Valor	Tipo
Dato 1	Raíz cuadrada	Constante	Entero
Dato 2	X, i	Variable	Real
Dato 3	Resultado	Variable	Entero

ALGORITMO

Algoritmo RaizCuadradaEntera

Definir x, i, num Como Real

Escribir 'Ingrese un numero'

Leer x

$i \leftarrow 1$

Repetir



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



$i <- i+1$

$num <- i*i$

Hasta Que $num \geq x$

$i <- i-1$

Escribir 'La raiz cuadrada de ', x, ' es ', i

FinAlgoritmo

PRUEBA DE ESCRITORIO

The screenshot displays a software environment with three main components:

- Code Editor (PSeInt):** Contains the following algorithm:


```

      1 Algoritmo RaizCuadradaEntera
      2 Definir x, i, num Como Real
      3 Escribir 'Ingrese un numero'
      4 Leer x
      5 i ← 1
      6 Repetir
      7   i ← i+1
      8   num ← i*i
      9 Hasta Que num ≥ x
      10 i ← i-1
      11 Escribir 'La raiz cuadrada de ', x, ' es ', i
      12 FinAlgoritmo
      
```
- Flowchart (PSDraw):** A visual representation of the algorithm. It starts with 'Definir x, i, num Como...', followed by 'Ingrese un numero'. A loop structure is shown with a decision diamond 'num ≥ x'. If 'F' (False), it goes to 'i ← i+1' and then to 'num ← i*i'. If 'V' (True), it goes to 'i ← i-1' and then to the output 'La raiz cuadrada de ...'. The flowchart ends with 'FinAlgoritmo'.
- Execution Console (PSeInt):** Shows the execution results:


```

      *** Ejecución Iniciada. ***
      Ingrese un numero
      > 5
      La raiz cuadrada de 5 es 2
      *** Ejecución Finalizada. ***
      
```

x	i	i+1	num	Num>=x	Pantalla (i-1)
5	1	2	4	F	
	2	3	9	v	2



Ejercicio 9

Problema 1.10 Números primos comprendidos entre dos números dados.
Desarrolle un programa que muestre por pantalla todos los números primos comprendidos entre dos números enteros introducidos por teclado.

Tal como se vio en el problema 1.9, un número primo es aquel que es divisible únicamente entre él mismo y la unidad. La solución planteada emplea dos módulos: el módulo *Es_primo* para determinar si un número es o no primo, y el módulo principal que recorre cada número en el intervalo de interés para determinar si son o no primos, usando el módulo *Es_primo*.

TABLA

Objeto	Nombre	Valor	Tipo
Dato 1	Números	variable	entero
Dato 2	Proceso	constante	entero
Dato 3	Resultado	variable	entero

ALGORITMO

1. Iniciar el algoritmo.
2. Definir las variables: x, z, i, j, esPrimo como enteros.
3. Escribir “ Ingrese dos números enteros”.
4. Leer los valores de x and z.
5. Si x es mayor que z, intercambiar sus valores para que x sea menor.
6. Escribir “ Los números primos entre x y z:”. respuesta
7. Para cada número i desde x + 1 hasta z - 1 hacer:
 - a. Suponer que i es primo (asignar esPrimo \leftarrow 1).
 - b. Para cada número j desde 2 hasta i - 1 hacer:



8. Si i es divisible por j , asignar $\text{esPrimo} \leftarrow 0$.
- c. Si i es menor o igual que 1, asignar $\text{esPrimo} \leftarrow 0$.
- d. Si esPrimo es igual a 1, mostrar i por pantalla.
9. Finalizar el algoritmo.

PRUEBA DE ESCRITORIO

Valores	Fórmula	Proceso	Resultado
4 , 45			5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43



```
<sin_titulo>* X
2 Definir a, b, i, j, esPrimo Como Entero
3
4 Escribir "Ingrese el primer número entero:"
5 Leer a
6 Escribir "Ingrese el segundo número entero:"
7 Leer b
8
9 // Asegurar que a sea menor que b
10 Si a > b Entonces
11     i ← a
12     a ← b
13     b ← i
14 FinSi
15
16 Escribir "Números primos entre ", a, " y ", b, ":"
17
18 Para i ← a + 1 Hasta b - 1 Hacer
19     esPrimo ← 1
20     Para j ← 2 Hasta i - 1 Hacer
21         Si i MOD j = 0 Entonces
22             esPrimo ← 0
23         FinSi
24     FinPara
25
26     Si i ≤ 1 Entonces
27         esPrimo ← 0
28     FinSi
29
30 Si esPrimo = 1 Entonces
31     Escribir i
32 FinSi
33 FinPara
34 FinAlgoritmo
```

PSelnt - Ejecutando proceso NUMEROSPRIMOSENTREDOS

*** Ejecución Iniciada. ***

Ingrese el primer número entero:

> 4

Ingrese el segundo número entero:

> 45

Números primos entre 4 y 45:

5

7

11

13



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



Objeto	Nombre	Valor	Tipo
Dato 1	Número		
Dato 2	Primo		
Dato 3	Resultado		

1) Table

Objeto	Nombre	Valor	Tipo
Dato 1	Números	variable	real
Dato 2	Proceso	constante	real
Dato 3	Resultado	variable	real

2) Algoritmos

1) ~~Inicio~~ algoritmo

2) Definir las variables: x, z, i, j , esPrimo como enteros

3) Escribir "Ingrese dos números enteros"

4) Leer los valores de x y z

5) Si x es mayor que z , intercambiar sus valores para que sea x menor.

6) Escribir "Los números primos entre x y z :"

7) Para cada número i desde $x + 1$ hasta $z - 1$ hacer:

a. Suponer que i es primo (asignar $\text{esPrimo} \leftarrow 1$).

b. Para cada número j desde 2 hasta $i - 1$ hacer:

8) Si i es divisible j , asignar $\text{esPrimo} \leftarrow 0$

d. Si es Primo es igual a 1, mostrar 1 por pantalla.

9. Fin algoritmo

Prueba de escritorio

Valores

4, 45

Resultados

5, 7, 11, 13, 17, 19, 23, 29, 31, 37
41, 43

