

Apellido y Nombre: ..... Fecha: ...../...../.....

### CONCEPTOS A TENER EN CUENTA

#### EXPRESIONES

- Las expresiones combinan constantes, variables, símbolos de operación y nombres de funciones especiales. Una expresión consta de *operandos* y *operadores*. Las expresiones se clasifican en *aritméticas*, *alfanuméricas* y *lógicas*.
- Las expresiones aritméticas (matemáticas) utilizan operadores aritméticos y variables y constantes numéricas (reales o enteras).
- Las expresiones alfanuméricas utilizan operadores alfanuméricos y se producen resultados de tipo alfanumérico.
- Las expresiones lógicas usan operadores lógicos y/o relacionales para representar alguna condición. La condición puede ser VERDADERA o FALSA.

#### OPERADORES

Los operadores ejecutan acciones sobre los datos. La siguiente tabla presenta los operadores más utilizados en programación.

Tipo	Símbolo	Nombre	Función
Paréntesis	( )		Anida expresiones
Aritméticos	^ *, / +, - div, mod	Potencia Producto, división Suma, diferencia División entera, resto	Conectan objetos o campos numéricos
Alfanuméricos	+	Concatenación	Conectan campos alfanuméricos
Relacionales	= < <= > >= <>	Igual a Menor que Menor o igual que Mayor que Mayor o igual que Distinto a	Conectan objetos, campos o expresiones de cualquier tipo. Su evaluación da como resultado "Verdadero" o "Falso".
Lógicos	NOT AND OR	Negación Conjunción Disyunción	Conectan expresiones de tipo lógico. Su evaluación da como resultado "Verdadero" o "Falso".

#### REGLAS DE EVALUACIÓN DE EXPRESIONES

- Las expresiones se evalúan de izquierda a derecha, respetando la prioridad de los operadores
- Las subexpresiones entre paréntesis se evalúan primero
- Las subexpresiones con paréntesis anidados se evalúan de adentro hacia afuera (el paréntesis más interno se evalúa primero)

#### PRIORIDAD DE OPERACIONES

Dentro de una misma expresión o subexpresión, los operadores aritméticos se evalúan en el siguiente orden:

Operador	Prioridad
^ *, / div, mod +, -	Más alta (se evalúa primero)
	Más baja (se evalúa al final)
Si se utilizan paréntesis, las expresiones encerradas se evalúan primero.	

**REGLA ASOCIADA IZQUIERDA:** Los operadores en una misma expresión o subexpresión con igual nivel de prioridad (tal como \* y /) se evalúan de izquierda a derecha.

## EXPRESIONES LÓGICAS

## Operadores de relación

Operador	Significado	Ejemplo
<	Menor que	$a < b$
<=	Menor o igual que	$a <= b$
=	Igual a	$a = a$
<>	Distinto a	$a <> b$
>=	Mayor o igual que	$b >= a$
>	Mayor que	$b > a$

Observación expresiones verdaderas en columna ejemplo considerando  $a=10$  y  $b=20$

**OPERADORES DE RELACIÓN:** Los operadores de relación se utilizan para expresar condiciones y describen una relación entre dos valores. El resultado de una expresión lógica es un valor de tipo lógico VERDADERO o FALSO.

## Operadores lógicos

Las expresiones lógicas pueden combinarse para formar expresiones más complejas utilizando operadores lógicos: AND, OR y NOT.

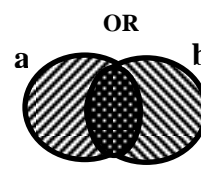
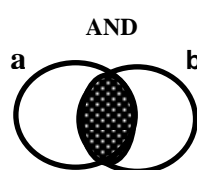
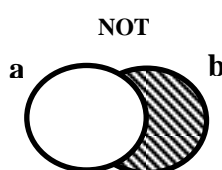
Orden de prioridad NOT, AND, OR.

## OPERADORES LÓGICOS:

**NOT (NO):** niega un valor.

**AND (Y):** Combina dos condiciones simples y produce un resultado verdadero solo si los dos componentes son verdaderos.

**OR (O):** es verdadera si uno de los componentes es verdadero.



## Tablas de verdad

## Operador NO (NOT)

Negación de a

A	NO a
Verdadero	Falso
Falso	Verdadero

## Operador Y (AND)

Conjunción de a y b

A	b	a Y b
Verdadero	Verdadero	Verdadero
Verdadero	Falso	Falso
Falso	Verdadero	Falso
Falso	Falso	Falso

## Operador O (OR)

Disyunción de a y b

A	b	a O b
Verdadero	Verdadero	Verdadero
Verdadero	Falso	Verdadero
Falso	Verdadero	Verdadero
Falso	Falso	Falso

## Expresiones complejas

Poseen expresiones aritméticas, de relación y lógicas.

Operador	Prioridad
NO, ^ *, /, Y div, mod +, -, O	Más alta (se evalúa primero)
<, <=, =, <>, >=, >	Más baja (se evalúa al final)

Si se utilizan paréntesis, las expresiones encerradas se evalúan primero.

## EVALUACIÓN DE EXPRESIONES COMPLEJAS:

1. Las expresiones aritméticas respetando su correspondiente jerarquía de operadores.
2. Las expresiones de relación respetando su correspondiente jerarquía de operadores.
3. Las expresiones lógicas respetando su correspondiente jerarquía de operadores.

## CADENAS DE CARACTERES

Ejemplos de cadenas: "esta es una cadena", "soy cadena".

**Longitud.** Esta función, cuyo argumento es una cadena, retorna un valor entero que indica el número de caracteres de la cadena.

Micadena ← "info"

Longitud(micadena) da como resultado 4.

**Comparación** La comparación de cadenas se basa en el orden numérico del código ASCII.

Ejemplo: Las siguientes comparaciones son verdaderas

'0' < '1', 'A' < 'B', 'M' > 'B', 'a' > 'Z', "diagrama" <= "pseudocodigo", "Pascal" = "Pascal"

**CADENA DE CARACTERES:** Una cadena de caracteres es un conjunto de caracteres (incluido el espacio en blanco) reconocidos por la computadora, los que se almacenan en posiciones de memorias contiguas. La longitud de una cadena es el número de caracteres que ésta contiene. La cadena que no contiene ningún carácter se denomina vacía o nula.

**OPERACIONES:** Longitud, Comparación, Concatenación, Extracción de Subcadenas

**Concatenación** es el proceso de concatenar dos o más cadenas en una sola. El símbolo que representa la concatenación varía de un lenguaje a otro. Los más utilizados son: +, // y &.

Ejemplo: 'sombrero' + ' loco' da como resultado 'sombrero loco'.

**Extracción de subcadenas** La operación de extraer una parte específica de una cadena se representa por la función subcadena. Esta función presenta el siguiente formato:

**subcadena(cadena, p\_inicial, p\_final)**

Ejemplo: Subcadena ('programas ejecutables',5,12) da como resultado 'ramas ej'.

#### SUBCADENA DE CARACTERES:

*Una subcadena es un conjunto de caracteres extraído de una cadena de mayor longitud.*

*Ejemplo: 'soy' es subcadena de 'soy cadena'.*

## CONJUNTOS

**Inicialización:** La inicialización de una variable conjunto se referencia como conjunto vacío.

Ejemplo: La variable letras está definida como un conjunto de tipo conjunto de caracteres. **letras ← { }**

**Asignación** permite cargar elementos en un conjunto. Ejemplo

considerando la variable misletras (de tipo conjunto de caracteres) se puede realizar la siguiente asignación:

**misletras ← {'a', 'm', 'q', 'z'}**

**Pertenencia:** utiliza la función pertenece que determina si un elemento en particular está contenido en un dato tipo conjunto. pertenece (nombre\_conjunto, elemento)

La función utiliza 2 argumentos, el primero indica el nombre de la variable tipo conjunto y el segundo especifica el elemento cuya pertenencia al conjunto se quiere verificar. La función retorna un valor lógico VERDADERO si el elemento especificado pertenece al conjunto indicado, de lo contrario devuelve FALSO.

Ejemplo: **pertenece (misletras, 'b')** retorna **FALSO**

**Unión** es un conjunto compuesto de los elementos de ambos conjuntos. La operación se denota con el signo +. Por ejemplo:

**otrasletras ← {'j', 'k', 'l'}**  
**misletras ← {'a', 'm', 'q', 'z'}**  
**todos ← otrasletras + misletras.**

La variable todos, de tipo conjunto, resulta de la unión de los conjuntos otrasletras y misletras. El conjunto todos contiene los elementos 'j', 'k', 'l', 'a', 'm', 'q' y 'z'.

**Intersección** es el conjunto formado por los elementos comunes a ambos conjuntos. El operador intersección es \*. Por ejemplo:

**simbolos ← {'b', 'm', 'q'}**  
**letras ← {'a', 'm', 'q', 'z'}**  
**interconj ← simbolos \* letras**

La variable interconj, de tipo conjunto, resulta de la intersección de los conjuntos simbolos y letras. El conjunto interconj contiene los elementos 'm' y 'q'.

**Diferencia** es el conjunto formado por los elementos del primero que no pertenecen al segundo. El operador diferencia es -. Por ejemplo:

**simbolos ← {'b', 'm', 'q'}**  
**letras ← {'a', 'm', 'q', 'z'}**

**difconj ← simbolos - letras.** La variable difconj, de tipo conjunto, resulta de la diferencia de los conjuntos símbolos y letras. El conjunto difconj contiene el elemento 'b'.

## ENUMERADO Y SUBRANGO

Un dato de un **tipo enumerado** es aquel que puede tomar por valor uno de los pertenecientes a una lista ordenada de valores definida por el programador.

```
enumerado <nombre_del_tipo> { <constante_1> [ = <valor_1> ],
                               <constante_2> [ = <valor_2> ],...
                               <constante_n> [ = <valor_n> ] }
```

**EJEMPLO** Los colores primarios pueden ser uno de los siguientes:

```
enumerado colores={rojo, azul, amarillo}
```

La lista de constantes está ordenada, y cada una de ellas representa a un valor entero empezando por el 0, e incrementándose de uno en uno. De manera que, las constantes declaradas representan a los valores enteros {0, 1, 2}.

**rojo** representa al valor 0

**azul** representa al valor 1

**amarillo** representa al valor 2

### Variables de tipos enumerados

Una **variable de un tipo enumerado** representa a un espacio de memoria en donde se puede almacenar un dato de un tipo enumerado.

**EJEMPLO** Dadas las declaraciones:

```
enumerado colores={rojo, azul, amarillo}
colores varcolores
```

*varcolores* es una variable del tipo enumerado colores. Por tanto, en el espacio de memoria representado por la variable se podrá almacenar uno de los valores {0, 1, 2}.

Un dato de un **tipo subrango** es aquel que puede tomar por valor uno de los pertenecientes a un subrango definido por el programador.

Matemáticamente, un rango es el conjunto de valores comprendidos entre un valor mínimo y un valor máximo, ambos inclusive.

Por ejemplo, suponiendo que el ordenador –utilizando dieciséis bits– puede representar el siguiente conjunto de valores enteros:

```
{ -32768, -32767, ..., -1, 0, 1, ..., 32766, 32767 }
```

Los valores mínimo y máximo de ese rango son los números -32768 y 32767, respectivamente. Pues bien, un subrango es un subconjunto de valores de un rango.

**EJEMPLO** Del rango { -32768, ..., 0, ..., 32767 } posibles subrangos son:

```
{ 1, 2, 3, 4, 5, 6 }
```

```
{ 0, 1, 2, ..., 8, 9, 10 }
```

```
{ -10, -9, -8, -7 }
```

```
{ -3, -2, -1, ..., 5, 6, 7 }
```

```
{ 1240, 1241, 1243, ..., 2999, 3000, 3001 }
```

**EJEMPLO** Sin embargo, al rango { -32768, ..., 0, ..., 32767 } no pertenecen los siguientes subrangos:

```
{ 0, 1, 2, ... 34998, 34999, 35000 }
```

```
{ -50000, -49999, -49998, ..., 49998, 49999, 50000 }
```

Como se puede observar, el conjunto de valores de un subrango también está comprendido entre un valor mínimo y un valor máximo. Además, en el subrango tienen que estar todos los elementos que están en el rango entre ambos valores.

**EJEMPLO** Así pues, del rango { -32768, ..., 0, ..., 32767 }, los siguientes conjuntos no pueden ser considerados como subrangos:

```
{ 2, 4, 6, 8, 10 }
```

```
{ 1, 2, 4, 8, 16, 32 }
```

Todos los datos de tipos subrangos son ordinales, es decir, solamente pueden tomar por valor elementos de subrangos finitos.

### Declaración de tipos subrangos

En pseudocódigo, para declarar un tipo de dato subrango, se puede utilizar la sintaxis:

```
<nombre_del_tipo> = <valor_mínimo>..<valor_máximo>
```

O también:

```
subrango <nombre_del_tipo> = <valor_mínimo>..<valor_máximo>
```

**EJEMPLO** Suponiendo que la calificación de una asignatura sea dada con un valor perteneciente al subrango { 0, 1, 2, ..., 8, 9, 10 } del rango { -32768, ..., 0, ..., 32767 }, para declarar el tipo subrango calificaciones, se debe escribir:

```
subrango calificaciones = 0..10
```

El indicador: .. separa el valor mínimo y máximo de un subrango.

Y el símbolo reservado igual (=), es el separador del identificador de un tipo de dato subrango y del subrango asignado al mismo.

Variables de tipos subrangos

Una **variable de un tipo subrango** representa a un espacio de memoria en donde se puede almacenar un dato de un tipo subrango.

**EJEMPLO** Dadas las declaraciones:

```
subrango calificaciones = 0..10
calificaciones matemática
```

**matemáticas** es una variable del tipo subrango **calificaciones**. En consecuencia, en el espacio de memoria representado por la variable se podrá almacenar uno de los valores del conjunto { 0, 1, 2, ..., 8, 9, 10 }.

## EJERCICIOS RESUELTOS

- Identifique el tipo de los datos de las siguientes variables, de acuerdo al tipo de Dato que contiene:

Variable	Dato contenido	Tipo de Dato
M	'D'	Carácter
Letra	"verdadero"	Cadena
Es	100	Entero
L2	4.25	Real
Num	FALSO	Lógico

- Escriba las siguientes expresiones algebraicas como expresiones algorítmicas:

*Ejemplo:*  $\left(\left(\frac{x+5y}{x}\right)^2 - \frac{7x+1}{5y}\right)^3$  expresión algorítmica equivalente  $((x+5*y)/x)^2 - (7*x+1)/(5*y)) ^3$

- Considerando las tablas de verdad de los operadores lógicos, resuelva las siguientes expresiones lógicas:

Expresión lógica	Resultado	Observación
(num <= 33) Y (num > 13)	Verdadero	num es una variable numérica que tiene valor 20
(4 <> cuenta) O (5 >= 4)	Verdadero	cuenta es una variable numérica que tiene valor 4
Resta = nro1 - nro2	Falso	nro1 y nro2 son variables numéricas que valen 10 y 18, respectivamente. Resta = 20

- Resuelva las siguientes expresiones teniendo en cuenta la prioridad de los operadores.

Expresión	Resultado	Observación
$5 * x^2 + 5 * y + 12$	52	x=2, y=4
$(12 + 8 * 3) \text{ div } 2$	18	
$64 / x * r - s * r \text{ mod } r^2$	32	x=8, r=4, s=44
$40 / 4 * 5 + 3$	53	

- Analice las siguientes operaciones de asignación y determine los valores finales de las variables A, B, y C.

Asignación	A	B	C
$A \leftarrow 22 * 2 \text{ div } 5$	8		
$B \leftarrow 24 \text{ mod } 2 + 5$		5	
$C \leftarrow A$			8

- Dadas las siguientes cadenas de caracteres, realice sobre ellas las operaciones solicitadas e indique el resultado obtenido.

Cadenas de caracteres	Operación	Resultado	Almacenado en variable	Tipo de dato de la variable
'Canal' 'de' 'Youtube'	Concatenación	'CanaldeYoutube'	A	Cadena de Caracteres
"alas"	Longitud	4	C	Entero
'canal' <= 'Canal'	Comparación	FALSO	E	Lógico
'Las cadenas son un conjunto de caracteres '	Subcadena posición carácter inicial=10 posición carácter final=20	'as son un c'	H	Cadena de Caracteres

7. Dados los siguientes conjuntos, realice sobre ellos las operaciones solicitadas e indique el resultado obtenido. Además, consigne el tipo del conjunto (entero, carácter, etc.).

Conjuntos	Tipo del Conjunto	Operación	Resultado
A= {3,5,16,24,85}	Entero	Pertenencia de 2 en A elemento a verificar: 2	Falso
A={3,5,16,24,85} B={5,7,9, 11,13,15,17,19,21}	Entero	Unión A+B	A+B = {3,5,7,9, 11,13,15,16,17,19,21,24,85}
A={'a','d','f','g','h'} B={'a','b','c','d','e','f','g'}	Carácter	Intersección A*B	A*B ={'a','d','f','g'}
A={'a','d','f','g','h'} B={'a','b','c','d','e','f','g'}	Carácter	Diferencia B-A	B-A = {'b','c','e' }

8. Entra [aquí](#) para ver ejemplos de problemas sobre conjuntos.

9. Definir tipos enumerados para representar los siguientes datos.

- a. Los días de la semana

**enumerado** dias {lunes, martes, miercoles, jueves, viernes, sabado, domingo}

- b. Las cuatro estaciones del año

**enumerado** estaciones {verano, otonio, invierno, primavera}

10. Definir tipos subrangos para representar datos como los siguientes.

- a. Los números naturales de tres cifras.

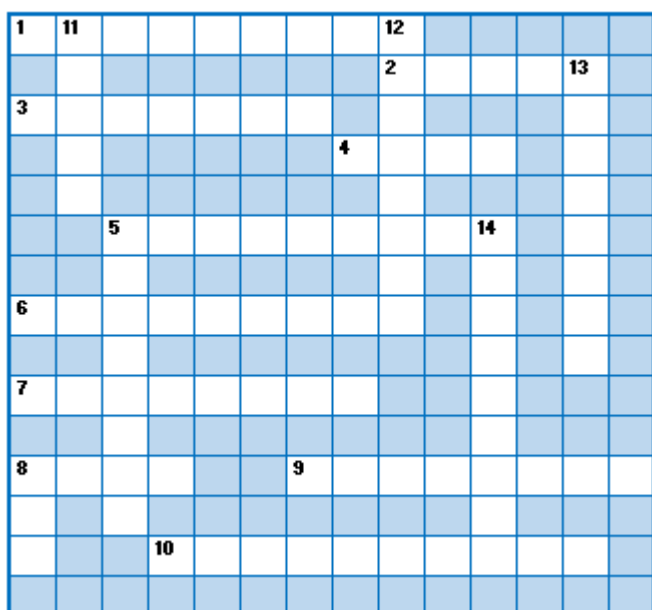
**subrango** natres = 100..999

- b. Las letras mayúsculas

**subrango** letras\_may = 'A'..'Z'

## EJERCICIOS A RESOLVER

1. Complete el siguiente crucigrama



### HORIZONTALES

- 1) Tipos de datos que permiten identificar elementos individuales (antecesor, sucesor, primero, último).
- 2) Operación que combina los elementos de 2 conjuntos.
- 3) Operación que permite almacenar valores en una variable.
- 4) Resultado (en letras) de la operación  $64 \bmod 3 * 2 - 5 * 2 \wedge 3 \div 39$
- 5) Objeto de datos que no modifica su valor durante la ejecución de un programa.
- 6) Resultado (en letras) de la operación  $(13 > 6) \vee (8 \geq 2*4) \vee (7 \bmod 3 < 1)$
- 7) Operación aplicable a datos de tipo numérico
- 8) Operación de entrada/salida
- 9) Operación de cadenas que determina la cantidad de caracteres de éstas.
- 10) Operación de cadenas que permite combinarlas en una sola.

### VERTICALES

- 5) Tipo de dato simple que permite representar símbolos, dígitos y letras.
- 8) Resultado de la operación `subcadena("espeluznante",5,7)`
- 11) Operación que permite obtener el módulo de un cociente entero.
- 12) Tipo de dato que se define a partir de tipo de dato ordinal.
- 13) Operación lógica que invierte un valor de verdad.
- 14) Combinación de variables, constantes y operadores.

2. Encuentra las 13 palabras (términos de la Unidad II Tipos de Datos) ocultas en esta Sopa de Letras. Ten en cuenta que las palabras pueden estar dispuestas de forma vertical, horizontal, diagonal, de izquierda a derecha o de derecha a izquierda, de abajo hacia arriba o de arriba hacia abajo.

E	S	J	F	D	F	K	B	L	O	C	B	N
T	H	O	W	O	N	R	C	M	U	E	T	H
S	C	R	W	S	C	H	L	O	G	I	C	O
U	P	O	D	V	O	W	N	D	D	O	F	I
B	Q	M	M	F	N	K	C	C	U	Y	E	Z
C	M	E	G	P	J	H	D	S	P	U	U	T
A	L	F	A	N	U	M	E	R	I	C	O	S
D	G	A	H	R	N	E	D	Z	R	N	H	Z
E	U	L	R	O	T	O	S	G	R	B	F	N
N	V	S	Q	R	O	R	E	T	N	E	E	L
A	X	O	T	S	E	U	E	C	O	D	G	K
O	C	Q	K	H	N	G	B	D	B	S	I	M
L	S	A	V	D	S	E	L	O	B	R	A	K
A	G	M	M	H	B	C	Y	O	R	N	X	Q
U	C	O	N	C	A	T	E	N	A	R	E	F

3. Identifique el tipo de los datos de las siguientes variables, de acuerdo al tipo de dato que contiene:

Variable	Contenido	Tipo de Dato
J	FALSO	
C	'c'	
Sum	"123"	
S	0	
T	500.25	
May	"MES"	
Nro	"cuenta"	
F	"01/02/11"	

4. Escriba las siguientes expresiones algebraicas como expresiones algorítmicas:

a)  $\sqrt[3]{x^2 + y^2}$

c)  $C - \frac{(AB^2 + A)^4}{C^3}$

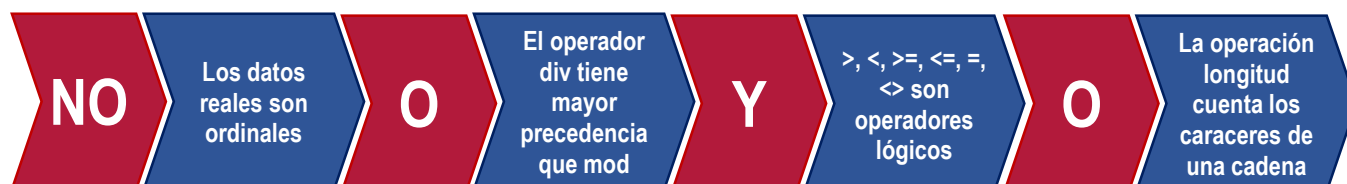
e)  $\frac{4 \times a - \sqrt{5 \times b - 6 \times c}}{\sqrt[3]{a + b + c}} - c^5$

b)  $\left(\frac{x + x^3}{2 \times x} - \frac{3 - x^2}{5}\right)^3$

d)  $\frac{-b - \sqrt{b^2 - 4ac}}{2a}$

f)  $\sqrt[3]{\frac{m^2 - n^3}{\sqrt{q + 1}}} + \frac{m \times q}{n - 2 \times q} - n^5$

5. ¿Cuál es el resultado de las siguientes expresiones lógicas?





6. Considerando las tablas de verdad de los operadores lógicos, resuelva las siguientes expresiones lógicas:

Expresión lógica	Resultado	Observación
$(z \leq 52) \vee (z > 75)$		z es una variable numérica que tiene valor 20
$(14 <> \text{mas}) \vee (15 \geq 24)$		mas es una variable numérica que tiene valor 18
$a \vee b \vee c \vee \text{NO}((b < a) \vee (b > a) \vee \text{NO } b)$		a=Verdadero, b=Falso, c=Verdadero
$(\text{band } \vee (w <> x) \vee \text{NO band } \vee \text{NO}(w < z)) \vee \text{NO}(x = z) \vee (w < 32)$		w=12, x=45, z=65, band=Verdadero
$(11 \leq 1) \vee (\text{NO}(18 > 17) \vee (1 \geq 1))$		
$(m \vee n \vee \text{NO } m \vee \text{NO } n) \vee \text{NO}(m \vee n \vee p \vee \text{NO } p)$		n=Falso, m=Verdadero, p=Verdadero
$a \vee \text{NO } b \vee \text{NO}(c \vee \text{NO } d \vee \text{NO } a \vee b) \vee c \vee d$		a=Verdadero, b=Falso, c=Verdadero, d=Falso
$\text{NO FALSO } \vee (\text{FALSO } <> \text{VERDADERO})$		
$\text{NO}(\text{NO } a \vee b \vee a \vee \text{NO } b) \vee a \vee b \vee c \vee \text{NO } a$		a=Verdadero, b=Falso, c=Verdadero
$\text{NO } w \vee z \vee x \vee w \vee \text{NO}((x = z) \vee (w > x))$		w=Falso, z=Falso, x=Verdadero

7. Complete los casilleros marcados con “?” de la siguiente grilla. Para ello, resuelva las operaciones aritméticas indicadas teniendo en cuenta la precedencia de operadores.

9	+	8	mod	6	/	?	=	?
/		mod		*		*		
3	*	?	div	4	-	5	=	-1
^		*		div		div		
?	^	5	mod	9	/	3	=	?
+		mod		*		mod		
1	-	9	mod	2	*	?	=	0
=		=		=		=		
2		?		?		3		

8. Resuelva las siguientes expresiones complejas teniendo en cuenta la prioridad de los operadores.

Expresión	Resultado	Observación
$(x=8) \vee (1+5 - 2*3 <> 0) \vee (b \text{ div } 2 = z/x)$		X = 8, b = 4, z = 40
$x * 8 ^ (1/3) + y \text{ mod } x^2 - z \text{ div } 2 * x$		X=3, y=14, z=30



Expresión	Resultado	Observación
$(z=8) \vee (x \leq 8) \vee ((6 > z) \vee (b > 1)) \vee (b \text{ div } 2 = x/z)$		$x=8, b=6, z=4$
$\text{No}(x-z*y=y) \vee ((z-1)/w < w)$		$x, y, z$ y $w$ son variables numéricas cuyos valores son 15, 21, 20 y 2, respectivamente.
$a * b * c \text{ div } d + 3$		$a=10, b=24, c=6, d=3$
$m \text{ div } n^2 + p/q^3 \text{ div } 2 - n^2/3$		$m=32, n=8, p=48, q=2$
$(b*a-y*z/2 > 0) \vee ((7/b)*(a-z) < 7) \vee \text{bandera}$		$\text{bandera}=\text{Falso}$ , $a, b, y, z$ valen 4, 8, 12 y 16 respectivamente
$(g/n + m) \text{ mod } 2 + m/g*n$		$m=32, g=8, n=4$
$m \text{ mod } n^2 + p*q^3 \text{ div } 2 - n^3/2$		$m=72, n=5, p=3, q=4$
$p^3/2 - m/n^2 \text{ div } 4 + p \text{ mod } q^3$		$m=81, n=3, p=5, q=2$
$c \text{ mod } a^2 - b*8^{1/3} + c \text{ div } 2*b$		$a=2, b=5, c=110$

9. Analice las siguientes operaciones de asignación y determine los valores finales de las variables  $x, y$  y  $z$ .

Asignación	X	Y	Z	Observación
$X \leftarrow (9+3) * 5 \text{ div } 4 \text{ mod } 7 + 3$				$z=20$
$Y \leftarrow 3 * x^{(1/2)} + 5 * x + z$				
$x \leftarrow z - y \text{ div } x$				
$y \leftarrow (x^8 + 1)^{(1/2)} + y \text{ mod } (z/x)^2 - z \text{ div } 2*x$				
$z \leftarrow x*z \text{ div } y + 5$				
$x \leftarrow x \text{ div } y^2 + z/x^3 \text{ div } 2 - (y^2-1)/3$				
$z \leftarrow x - z$				

**ASIGNACIÓN:** los valores de las variables pueden cambiar durante la ejecución del programa. Para cambiar el valor de una variable debe asignarse mediante el operador de asignación ( $\leftarrow$ ) o un dispositivo de entrada (lectura). Ejemplo:  $\text{num} \leftarrow 20$  (almacena el valor 20 en la variable num), LEER num (se lee un valor por teclado y se almacena en la variable num). La operación de asignación es destructiva, es decir, el nuevo valor reemplaza al anterior.

10. Defina tipos enumerado o subrango (según corresponda) para representar los siguientes datos:

- |                      |            |                      |
|----------------------|------------|----------------------|
| a. Meses del año     | c. Vocales | e. Colores primarios |
| b. Letras minúsculas | d. Dígitos | f. Notas musicales   |

11. Dados los siguientes conjuntos, realice sobre ellos las operaciones solicitadas e indique el resultado obtenido. Además, consigne el tipo del conjunto (entero, carácter, etc.).

Conjunto	Tipo de Conjuntos	Operación	Resultado	Observación
$A=\{2, 4, 0, 6, 3, 5\}$ $B=\{1, 2, 8, 5, 8, 9, 0, 7\}$		Intersección: $A*B$		
$A=\{'m', 'd', 'r', 't', 'z'\}$ $B=\{'o', 'p', 'q', 'r', 's', 't', 'u', 'v'\}$ $C=\{'m', 'n', 'x', 't', 'w'\}$		Unión: $A+D$		$D=C*B$
$A=\{2, 4, 0, 6, 3, 5\}$ $B=\{1, 2, 8, 5, 8, 9, 0, 7\}$		Diferencia: $A-B$		
$\{15, 25, 35, 55, 65\}$		Pertenencia de $x$		$X=45$
$A=\{'m', 'd', 'r', 't', 'z'\}$ $B=\{'o', 'p', 'q', 'r', 's', 't', 'u', 'v'\}$ $C=\{'m', 'n', 'x', 't', 'w'\}$		$B-A*C$		

12. Dadas las siguientes cadenas de caracteres, realice sobre ellas las operaciones solicitadas e indique el resultado obtenido.

Cadenas de Caracteres	Operación	Resultado	Almacenado en variable	Tipo de dato de la variable	Observación
'mar' = 'MAR'	Comparación		e		
'pez' > 'peces'	Comparación		f		
'DATO' <> "DATO"	Comparación		g		
Subcadena (cad)	Subcadena posición carácter inicial=3 posición carácter final=5		h		cad = "estructurados"
'la variable fue' 'declarada' 'de forma correcta'	Concatenación		b		
'Los programas no compilaron'	Longitud		d		

## ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	A	97	61	141	a
2	2	2		34	22	42	"	66	42	102	B	98	62	142	b
3	3	3		35	23	43	#	67	43	103	C	99	63	143	c
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	'	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	H	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	A	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	B	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	C	14		44	2C	54	,	76	4C	114	L	108	6C	154	l
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56	.	78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	O	111	6F	157	o
16	10	20		48	30	60	0	80	50	120	P	112	70	160	p
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	s
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	v
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	x
25	19	31		57	39	71	9	89	59	131	Y	121	79	171	y
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	

