

## Programación Java Tema 2. Conceptos básicos. Relación de ejercicios 3

### Soluciones

1. Dados los siguientes valores para las variables Booleanas X, Y, Z (X=true, Y=false, Z=true), evalúa las expresiones que aparecen a continuación:

a) `(X && Y) || (X && Z)`  
`(T && F) || (T && T)`  
`F || T`  
**T**

b) `(X || !Y) && (!X || Z)`  
`(T || !F) && (!T || T)`  
`(T || T) && (F || T)`  
`T && T`  
**T**

c) `X || Y && Z`  
`T || F && T`  
`T || F`  
**T**

d) `!(X || Y) && Z`  
`!(T || F) && T`  
`!( T ) && T`  
`F && T`  
**F**

e) `X || Y || X && !Z && !Y`  
`T || F || T && !T && !F`  
`T || F || T && F && T`  
`T || F || F && T`  
`T || F || F`  
`T || F`  
**T**

2. Dados los siguientes valores para las variables booleanas W, X, Y, Z (W = false, X=true, Y=true, Z=false), evalúa las expresiones que aparecen a continuación:

a) `W || Y && X && W || Z`  
`F || T && T && F || F`  
`F || T && F || F`  
`F || F || F`  
`F || F`  
**F**

b) `X && !Y && !X || !W && Y`  
`T && !T && !T || !F && T`  
`T && F && F || T && T`  
`F && F || T && T`  
`F || T && T`  
`F || T`  
**T**

c) `int i = !(W || Y && Z) ? 1 : 2;`  
`!(F || T && F)`  
`!(F || F)`  
`!( F )`  
**T**  
**=> i = 1;**

d) `double d = !(W || !Z) && X ? 0.5 : 0.10;`  
`!(F || !F) && T`  
`!(F || T) && T`  
`!( T ) && T`  
`F && T`  
**F**  
**=> d = 0.10;**

e) `int k = Y || Z && W || !X ? 1 : 2;`

`T || F && F || !T`

`T || F && F || F`

`T || F || F`

`T || F`

**T**

**=> k = 1;**

f) `int m = 5, n = !(Y || Z) && !(W || !X) ? ++m : --m;`

`!(T || F) && !(F || !T)`

`!(T || F) && !(F || F)`

`!( T ) && !( F )`

`F && T`

**F**

**n = --m; => primero se realiza --m (m = 4)  
y después se asigna el valor de m a n**

**=> n = 4;**

g) `int p = X && !Z ? (W || Z ? 0 : 1) : 2;`

`T && !F ? (F || F ? 0 : 1) : 2`

`T && T`

**T**

**p = F || F ? 0 : 1**

**F**

**=> P = 1;**

h) `int q = !X || Z ? 0 : (Y && Z ? 1 : 2);`

`!T || F ? 0 : (T && F ? 1 : 2)`

`F || F`

**F**

**q = T && F ? 1 : 2**

**F**

**=> q = 2**

### 3. ¿Qué se muestra por pantalla?

	x	Salida	
<code>int x = 1</code>	1		
<code>System.out.println(x)</code>		1	
<code>x++</code>	2		
<code>System.out.println(x)</code>		2	
<code>System.out.println(++x)</code>	3	3	Primero incrementa x y después se muestra
<code>System.out.println(x++)</code>	4	3	Primero se muestra x y después se incrementa
<code>System.out.println(x)</code>		4	
<code>System.out.println(++x)</code>	5	4	Primero se muestra x y después se incrementa
<code>System.out.println(++x)</code>	6	6	Primero incrementa x y después se muestra
<code>System.out.println(++x)</code>	7	7	Primero incrementa x y después se muestra

#### 4. ¿Qué se muestra por pantalla?

	A	B	Salida	
char A = 'c'	c			
char B				
System.out.println(A++)	d		c	primero se muestra y después se incrementa
System.out.println(A++)	e		d	primero se muestra y después se incrementa
System.out.println(++A)	f		f	primero se incrementa y después se muestra
B = A++	g	f		primero B = A y después A++
System.out.println(++A)	h		h	primero se incrementa y después se muestra
System.out.println(B++)		g	f	primero se muestra y después se incrementa
System.out.println(++B)		h	h	primero se incrementa y después se muestra

#### 5. ¿Qué se muestra por pantalla?

	x	Salida	
char x = 'g'	g		
x++	h		
System.out.println(x)		h	
System.out.println(x++)	i	h	primero se muestra y después se incrementa
System.out.println(x)		i	
System.out.println(--x)	h	h	primero se decrementa y después se muestra
x-=2	f		
System.out.println(x)		f	
x+=3	i		
System.out.println(x++)	j	i	primero se muestra y después se incrementa
System.out.println(++x)	k	k	primero se incrementa y después se muestra
System.out.println(x--)	j	k	primero se muestra y después se decrementa
System.out.println(--x)	i	i	primero se decrementa y después se muestra

#### 6. ¿Qué se muestra por pantalla?

	a	b	c	Salida	
int a = 1, b = 2, c	1	2			
c = a + b++		3	3		b se incrementa después
System.out.println(c)				3	
c = a + ++b		4	5		b se incrementa antes
System.out.println(c)				5	
c = ++a + b	2		6		a se incrementa antes
System.out.println(c)				6	
c = ++a + ++b	3	5	8		a y b se incrementan antes
System.out.println(c)				8	
c = ++a + b++	4	6	9		a se incrementa antes, b después
System.out.println(c)				9	
c = a++ + b	5		10		a se incrementa después
System.out.println(c)				10	
c = a++ + ++b	6	7	12		a se incrementa después, b antes
System.out.println(c)				12	
c = a++ + b++	7	8	13		a y b se incrementan después
System.out.println(c)				13	

### 7. Un programa contiene las siguientes declaraciones y asignaciones de variables:

```
int i = 1, j = 1, k = 1;
```

Determina el valor final de todas las variables que intervienen en cada una de las siguientes expresiones de asignación.

Las instrucciones son independientes unas de otras, es decir, el valor inicial en cada instrucción es

**i = 1, j = 1, k = 1.**

- a) i = ++j;
- b) i = k++;
- c) i = k + ++j;
- d) i = i + j++;
- e) i = (j >= 0 && k > 1) ? ++j : k+1;
- f) i = (j > 1 || k >= 1) ? ++j : k+1;
- g) i = j + ++k;
- h) i = ++k + k;
- i) i = k++ + k;
- j) i += ++k + k;

- a) i = 2    j = 2    k = 1
- b) i = 1    j = 1    k = 2
- c) i = 3    j = 2    k = 1
- d) i = 2    j = 2    k = 1
- e) i = 2    j = 1    k = 1
- f) i = 2    j = 2    k = 1
- g) i = 3    j = 1    k = 2
- h) i = 4    j = 1    k = 2
- i) i = 3    j = 1    k = 2
- j) i = 5    j = 1    k = 2

### 8. Resuelve el ejercicio anterior suponiendo ahora que las instrucciones no son independientes. Se ejecutan una a continuación de la otra tomando cada instrucción los valores producidos por su instrucción anterior.

- a) i = 2    j = 2    k = 1
- b) i = 1    j = 2    k = 2
- c) i = 5    j = 3    k = 2
- d) i = 8    j = 4    k = 2
- e) i = 5    j = 5    k = 2
- f) i = 6    j = 6    k = 2
- g) i = 9    j = 6    k = 3
- h) i = 8    j = 6    k = 4
- i) i = 9    j = 6    k = 5
- j) i = 21   j = 6    k = 6

## 9. ¿Qué se muestra por pantalla?:

```
a) int x = 5, y = 2;  
   System.out.println(x / y >= 2.5 ? "verdadero" : "falso" );
```

5 / 2 >= 2.5

2 >= 2.5

F

=> muestra por pantalla "falso"

```
b) int w = 1, x = 5, y = 2, z = 10;  
   System.out.println(y <= x && x < y * z ? w++ : ++z);
```

2 <= 5 && 5 < 2 \* 10

2 <= 5 && 5 < 10

T && T

T

=> muestra por pantalla el valor actual de w y después w incrementa su valor

=> muestra por pantalla 1

=> el valor final de w es 2

```
c) int x = 5, y = 2;  
   System.out.println(x + y / 2 % 3 >= x + 1 ? x + 2 * y : x / y + 2);
```

5 + 2 / 2 % 3 >= 5 + 1

5 + 1 % 3 >= 5 + 1

5 + 1 >= 5 + 1

6 >= 6

T

=> muestra por pantalla el resultado de: x + 2 \* y

5 + 2 \* 2

5 + 4

9

=> muestra: 9

```
d) int a = 1, b = 1, x = 5, y = 2;  
   a += (x % 2 == 0 && y % 2 == 0 ? b++ : ++b);
```

5 % 2 == 0 && 2 % 2 == 0

1 == 0 && 0 == 0

F && T

F

=> a += ++b;

a = a + ++b;

a = 1 + 2; primero se incrementa el valor de b

a = 3

```
System.out.println(++a);
```

=> muestra por pantalla: 4

=> primero se incrementa el valor de a y después se muestra

e) `int a = 1, b = 2, x = 5, y = 2;`

`b += a + (x % 2 != 0 && y % 3 == 0 ? x++ : ++y);`

`5 % 2 != 0 && 2 % 3 == 0`

`1 != 0 && 2 == 0`

`T && F`

`F`

=> `b += a + ++y;`

`b = b + a + ++y;`

`b = 2 + 1 + 3;` primero se incrementa y después se realiza la suma

`b = 6;`

`b--2;`

=> `b = b - 2;`

`b = 6 - 2;`

`b = 4;`

`System.out.println(++b);`

=> muestra por pantalla: 5

=> primero se incrementa el valor de b y después se muestra

## 10. Determina el valor de cada una de las siguientes expresiones:

a) `2 + 3 < 2 * 3 || 6 < 10`

`2 + 3 < 6 || 6 < 10`

`5 < 6 || 6 < 10`

`T || T`

**T**

b) `8 == 6 * 2 && 12 < 15`

`8 == 12 && 12 < 15`

`F && T`

**F**

c) `8 == 6 * 2 || 12 < 15`

`8 == 12 || 12 < 15`

`F || T`

**T**

d) `!(3 < 5)`

`!T`

**F**

e) `2 + 2 / 3 >= 2 && 13 % 3 * 2 == 2`

`2 + 0 >= 2 && 1 * 2 == 2`

`2 + 0 >= 2 && 2 == 2`

`2 >= 2 && 2 == 2`

`T && T`

**T**

f) `5 == 1 / 5`

`5 == 0`

**F**

g) `5 == 5 % 1`

`5 == 0`

**F**

h) `3 == 10 / 3 && 3 == 10.0 / 3`

`3 == 3 && 3 == 3.333333`

`T && F`

**F**

```
i) 7 % 4 % 2 / 2 + 10 % 3 % 2 * 3 / 2
    3 % 2 / 2 + 10 % 3 % 2 * 3 / 2
    1 / 2 + 10 % 3 % 2 * 3 / 2
    0 + 10 % 3 % 2 * 3 / 2
    0 + 1 % 2 * 3 / 2
    0 + 1 * 3 / 2
    0 + 3 / 2
    0 + 1
    1
```

### 11. Sean las siguientes declaraciones de variables:

```
int i;
long ln;
short s;
float f;
double d;
char c;
byte b;
```

Determina el tipo de dato del resultado de las siguientes expresiones:

- |                                      |                                   |                                |
|--------------------------------------|-----------------------------------|--------------------------------|
| a) $i + ln \Rightarrow long$         | b) $b + c + s \Rightarrow int$    | c) $d + ln \Rightarrow double$ |
| d) $ln + s \Rightarrow long$         | e) $b + f \Rightarrow float$      | f) $f + d \Rightarrow double$  |
| g) $d + c \Rightarrow double$        | h) $i + ln + f \Rightarrow float$ | i) $c + s \Rightarrow int$     |
| j) $s + b \Rightarrow int$           | k) $i + 2L \Rightarrow long$      | l) $.5 * f \Rightarrow double$ |
| m) $2E4f + i + ln \Rightarrow float$ | n) $0B10 + s \Rightarrow int$     | o) $0x2 + ln \Rightarrow long$ |

### 12. A partir de las siguientes declaraciones de variables:

```
var x = 0b11;
var y = x + 23.5;
var z = y + .23F;
var k = x + 'w';
var w = k + y + z;
```

Determina el tipo de las variables x, y, z, k, w.

$x \Rightarrow int$        $y \Rightarrow double$        $z \Rightarrow double$        $k \Rightarrow int$        $w \Rightarrow double$

### 13. Dada una variable entera N, escribe las instrucciones de asignación que realicen lo indicado en cada uno de los apartados siguientes:

#### a) Sumar 5 a N.

$N = N + 5;$       o también  $N += 5;$

#### b) Restar 10 a N.

$N = N - 10;$       o también  $N -= 10;$

#### c) Multiplicar por 3 el valor de N.

$N = N * 3;$       o también  $N *= 3;$

#### d) Si el valor de N es positivo, sumarle 5 sino sumarle 100.

$N = N + (N > 0 ? 5 : 100);$       o también  $N += N > 0 ? 5 : 100;$

#### d) Si el valor de N es negativo, asignarle un 5 sino asignarle un 100.

$N = N < 0 ? 5 : 100;$

14. Escribe las expresiones algorítmicas correspondientes a las siguientes operaciones:

- a) El doble de un número N

$2 * N$

- b) El triple de un número N

$3 * N$

- c) Seis veces la diferencia de dos números enteros A y B

$6 * (A - B)$

- d) La diferencia entre el producto de A por B y la suma de C más D

$A * B - (C + D)$

- e) La mitad de la última cifra de un número entero A

$A \% 10 / 2$

- f) La suma de los dígitos de un número entero N de 3 cifras

dígito central

$N / 100 + N / 10 \% 10 + N \% 10$

dígito de la izquierda                      dígito de la derecha

El dígito central de un número de 3 cifras también se puede obtener de esta forma:

$N \% 100 / 10$

- g) La suma de la última cifra de los números enteros N y M

$N \% 10 + M \% 10$

- h) Comprobar si un número entero N es múltiplo de 2 y de 3

$N \% 2 == 0 \ \&\& \ N \% 3 == 0$

- i) Comprobar si la última cifra de un número entero N es par

$N \% 10 \% 2 == 0$

- j) Comprobar si la primera cifra de un número entero N de 3 cifras es impar

$N / 100 \% 2 != 0$  o también de esta forma:  $N / 100 \% 2 == 1$

- k) Comprobar si la primera cifra de un número entero N de 4 cifras es par

$N / 1000 \% 2 == 0$

- l) Comprobar si una variable A de tipo carácter contiene una letra mayúscula

$A >= 'A' \ \&\& \ A <= 'Z'$

- m) Comprobar si una variable A de tipo carácter contiene una letra mayúscula o minúscula.

$A >= 'A' \ \&\& \ A <= 'Z' \ || \ A >= 'a' \ \&\& \ A <= 'z'$

- n) Comprobar si una variable A de tipo carácter no contiene una letra mayúscula

$A < 'A' \ || \ A > 'Z'$

- o) Comprobar si una variable A de tipo carácter no contiene una letra mayúscula o minúscula.

$!(A >= 'A' \ \&\& \ A <= 'Z' \ || \ A >= 'a' \ \&\& \ A <= 'z')$

También se puede expresar de esta forma:

$(A < 'A' \ || \ A > 'Z') \ \&\& \ (A < 'a' \ || \ A > 'z')$



- p) Comprobar si el contenido de la variable N termina en 0 ó en 7

```
N % 10 == 0 || N % 10 == 7
```

- q) Sumar a una variable N de tipo entero su dígito situado más a la derecha.

```
N = N + N % 10 o también N += N % 10
```

### 15. Escribe las expresiones algorítmicas correspondientes a las siguientes operaciones:

- a) Sumar a una variable N de tipo entero su dígito situado más a la derecha. Por ejemplo si N contiene el valor 123, después de la operación contendrá el valor 126. Si N contiene el valor 7, después de la operación contendrá el valor 14.

```
N = N + N % 10 o también N += N % 10
```

- b) Comprobar si un número entero N de cuatro cifras es capicúa. Un número es capicúa cuando se puede leer igual de derecha a izquierda o de izquierda a derecha. Ejemplos de números de cuatro cifras capicúas: 1221, 3003, 5555.

```
N / 1000 == N % 10 && N / 100 % 10 == N % 100 / 10
```

- c) Una variable entera M contiene un número de mes. Comprobar si corresponde a un mes de 30 días.

```
M == 4 || M == 6 || M == 9 || M == 11
```

- d) Quitarle a un número entero N su última cifra. Supondremos que N tiene más de una cifra. Por ejemplo si N contiene el valor 123, después de la operación contendrá el valor 12.

```
N = N / 10
```

- e) Quitarle a un número entero N de 5 cifras su primera cifra. Por ejemplo si N contiene el valor 12345, después de la operación contendrá el valor 2345.

```
N = N % 10000
```

- f) Comprobar si una variable C de tipo char contiene un dígito. (Carácter entre 0 y 9).

```
C >= '0' && C <= '9'
```

- g) Dado un número N de cuatro cifras, comprobar si la primera cifra (la más a la izquierda) es impar.

```
N / 1000 % 2 != 0
```

- h) Dado un número N de cinco cifras, comprobar si la primera cifra (la más a la izquierda) es igual a la segunda.

```
N / 10000 == N / 1000 % 10
```

- i) Determinar si un número entero N de 5 cifras es capicúa.

```
N / 10000 == N % 10 && N / 1000 % 10 == N / 10 % 10
```

- j) Comprobar si una variable C de tipo char contiene una vocal mayúscula.

```
C == 'A' || C == 'E' || C == 'I' || C == 'O' || C == 'U'
```

- k) Comprobar si una variable C de tipo char no contiene una vocal mayúscula.

Dos soluciones posibles:

```
!(C == 'A' || C == 'E' || C == 'I' || C == 'O' || C == 'U')
```

```
C != 'A' && C != 'E' && C != 'I' && C != 'O' && C != 'U'
```

- l) Dadas dos variables A y B de tipo char, comprobar si las dos contienen una vocal minúscula

```
(A == 'a' || A == 'e' || A == 'i' || A == 'o' || A == 'u') &&
```

```
(B == 'a' || B == 'e' || B == 'i' || B == 'o' || B == 'u')
```

m) Dada una variable A que contiene un año, determinar si ese año es bisiesto. Un año es bisiesto si es divisible por 4 y no por 100 ó si es divisible por 400

```
A % 4 == 0 && A % 100 != 0 || A % 400 == 0
```

n) Dado un número N de dos cifras, comprobar si las dos cifras son iguales.

```
N / 10 == N % 10
```

o) Dado un número N de dos cifras, comprobar si la suma de sus cifras es un número par.

```
(N / 10 + N % 10) % 2 == 0
```

p) Dado un número N de tres cifras, comprobar si la cifra del centro es la mayor.

```
N / 10 % 10 > N / 100 && N / 10 % 10 > N % 10
```

q) Dado un número N de cuatro cifras, comprobar si alguna de las cifras es un 4.

```
N / 1000 == 4 || N / 100 % 10 == 4 || N / 10 % 10 == 4 || N % 10 == 4
```

r) Dado un número N de 4 cifras, asigna a una variable X las dos primeras cifras del número.

```
X = N / 100
```

s) Dado un número entero N, modifícalo restando a N el valor de su última cifra.

Por ejemplo, si N = 123, el valor final de N debe ser 120.

```
N = N - N % 10 o también N -= N % 10
```

t) Dados dos números enteros N y M, modifica M restándole la última cifra de N.

Por ejemplo si M = 123 y N = 47, el valor final de M debe ser 116.

```
M = M - N % 10 o también M -= N % 10
```

**16. Se tienen dos variables enteras A y B. Escribe las instrucciones necesarias para intercambiar sus valores entre sí. Utiliza una variable auxiliar para realizar el intercambio.**

Por ejemplo, si A contiene un 1 y B contiene un 2 debes escribir las instrucciones de asignación necesarias para que al final el valor de A sea 2 y el valor de B sea 1.

```
AUX = A;
```

```
A = B;
```

```
B = AUX;
```

**17. Se tienen tres variables A, B y C. Escribe las instrucciones necesarias para intercambiar entre sí sus valores de forma que:**

B toma el valor de A

A toma el valor de C

C toma el valor de B

Solo se puede utilizar una variable auxiliar.

```
AUX = B
```

```
B = A
```

```
A = C
```

```
C = AUX
```

### 18. Construye expresiones lógicas correctas:

- a) A es igual a B, pero no es igual a C

`A == B && A != C`

- b) A es positivo pero no es par

`A > 0 && A % 2 != 0`

- c) B es impar y mayor que 5 y A es múltiplo de 3

`B % 2 != 0 && B > 5 && A % 3 == 0`

- d) Alguno de los dos A ó B son pares

`A % 2 == 0 || B % 2 == 0`

- e) A es par pero B no lo es

`A % 2 == 0 && B % 2 != 0`

- f) Con DN, MN, AN día, mes y año de nacimiento de una persona y DA, MA, AA día, mes y año actual, expresar si tiene 18 años cumplidos.

`AA - AN > 18 || AA - AN == 18 && MA > MN || AA - AN == 18 && MA == MN && DA >= DN`