**Trabajo práctico nro. 1**

|  |  |  |
| --- | --- | --- |
| Logo Departamento 2.jpg | **Asignatura: Laboratorio I** | |
|  | |
| **Cursado:**Primer Trimestre | **Horas** **semanales**: |
|  | **Horas semestrales:**  *Cantidad estimada de horas semestrales/anuales.* |
| **Carrera**: *Tecnicatura Universitaria en Programación* | **Nivel (Año):** |
| **Ciclo Lectivo: 2023** |

**Integrantes de la Cátedra:**

* **DOCENTES:**

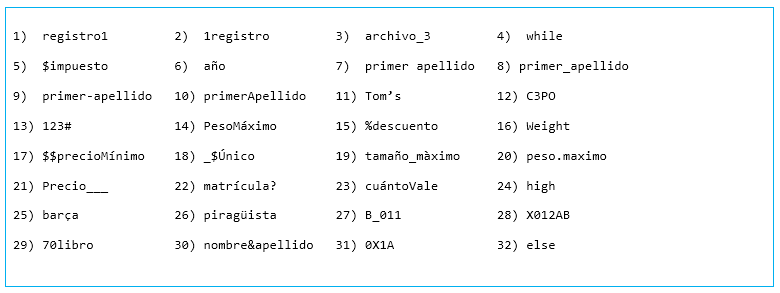
|  |  |  |
| --- | --- | --- |
| **Nombre del Profesor** | **Periodo** | **Cantidad horas materia** |
|  |  | 6 horas |

1. **Tema:**

**Ejercicios Básicos iniciales para principiantes**

1. **Enunciados:**
2. **Ejercicio sobre identificadores: indica qué identificadores Java son válidos.**

Indica cuáles de los siguientes identificadores son válidos en Java. Si el identificador no es válido explica porqué no lo es.



1-Si es valido

2-No, por que empieza con numero

3-Si es valido

4-No, por que es palabra reservada

5-No, empieza con carácter especial

6-Si, pero no es recomendable por la ñ

7- no, esta separado

8­­­­­­-si es valido

9-no, tiene un operador de suma

10-si es valido

11-no, tiene ‘

12- si es valido

13- no, empieza con números

14-si es valido

15-No, empieza con carácter especial

16-si es valido

17- no, contiene carácter especial

18-si es valido

19-si es valido pero no recomendable

20-no , tiene operador \*

21-si es valido

22-si es valido

23-si es valido, no recomendable

24-si es valido

25- no es valido,carácter especial

26-si es valido

27=si es valido

28-si es valido

29-no es valido, empieza con numero

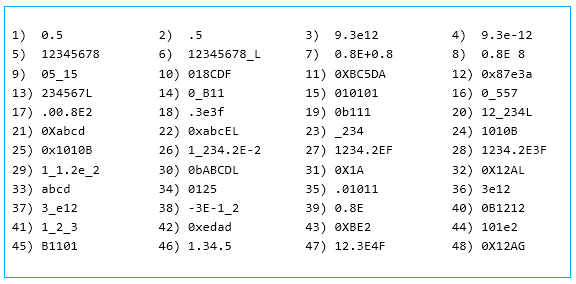
30-si es valido

31-no empieza con numero

32-es palabra reservada, no.

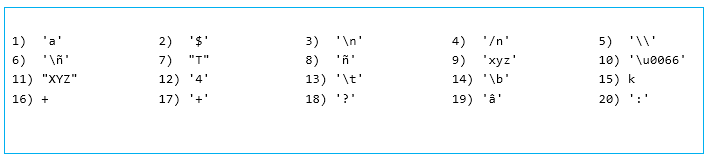
1. **Ejercicio sobre literales: indica qué literales Java son válidos**.

Indica cuáles de los siguientes literales son válidos en Java. Si el literal es válido indica además de qué tipo es (int, double, long, etc) y el sistema de numeración en el que está escrito (decimal, binario, octal, hexadecimal). Si el literal no es válido explica porqué no lo es.



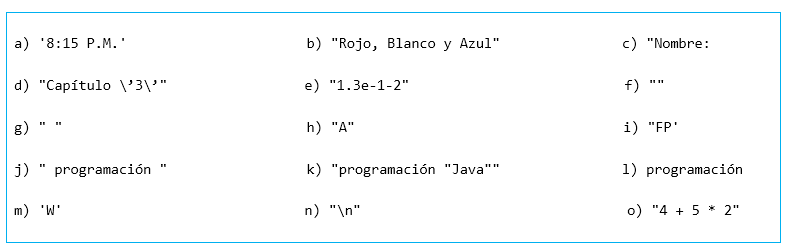
1. Valido, double, decimal
2. Valido, double, decimal
3. Valido, double, decimal
4. Valido, double, decimal
5. Valido, long, decimal
6. Invalido, carácter \_
7. Invalido, operador +
8. Inavlido, espacio en blanco
9. Valido, int, octal
10. Invalido
11. Valido, int, hexadecimal
12. Valido, int, hexadecimal
13. Valido, long, decimal
14. Invalido, el carácter \_ separa el 0 de la B
15. Valido, int, octal
16. Valido, int, octal
17. Invalido, solo puede tener un punto
18. Valido, float, decimal
19. Valido, int, binario
20. Valido, long, decimal
21. Valido, int, hexadecimal
22. Valido, long, hexadecimal
23. Invalido, carácter \_
24. Invalido,
25. Valido, int, hexadecimal
26. Valido
27. Invalido, no se eleva a la f
28. Valido, float,decimal
29. Invalido, separación
30. Invalido, ob
31. Si,int,hexadecimal
32. Si, long, hexadecimal
33. Invalido
34. Si,int, octal
35. Si,doublé,decimal
36. Si,doublé,decimal
37. Invalido, barra
38. Valido,doublé,decimal
39. Invalido,no tiene exponente
40. Invalido,solo puede tener 0 y 1
41. Si,int,decimal
42. Valido,int,decimal
43. Valido,int hexadecimal
44. Valido,doublé,decimal
45. Invalido,tiene que empezar con 0b
46. Invalido, solo puede tener un .
47. Valido,float,decimal
48. Invalido,G no es un valor valido
49. **Ejercicio sobre literales de tipo char.**

Indica cuáles de los siguientes literales de tipo char son válidos en Java. Si el literal no es válido explica el motivo.



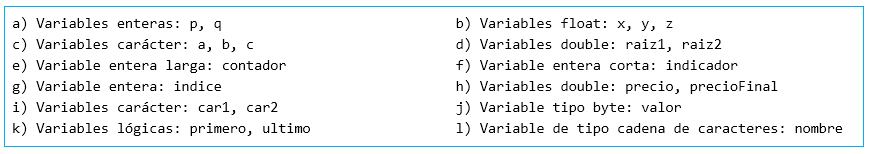
1. Valido
2. Valido
3. Valido
4. Invalido, símbolo al revés
5. Valido
6. Invalido, símbolo al revés
7. Invalido, tiene que ir comillas simple
8. Valido
9. Invalido, string
10. Valido
11. Invalido,string
12. Valido
13. Valido
14. Valido
15. Invalido,comillas
16. Invalido,operador
17. Valido
18. Valido
19. Valido
20. Valido
21. **Ejercicio sobre literales de tipo String.**

Indica cuáles de los siguientes literales de tipo String son válidos en Java. Si el literal no es válido explica el motivo.



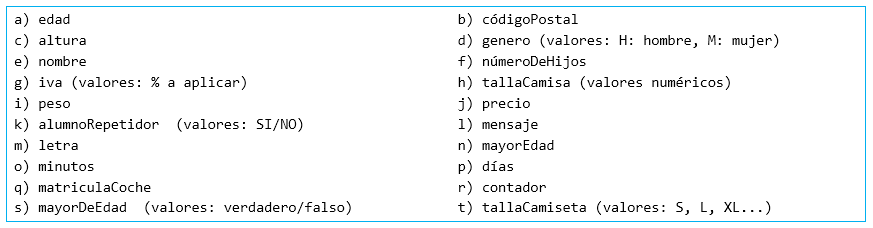
1. Invalido,doble comilla
2. Valido
3. Invalido,falta comillas
4. Valido
5. Valido
6. Valido
7. Valido
8. Valido
9. Invalido, comillas
10. Valido
11. Invalido, comillas
12. Invalido,falta comillas
13. Invalido, doble comillas
14. Valido
15. valido
16. Ejercicio sobre declaraciones de variables

**Ejercicio 1**: Escribe la declaración para cada una de las variables y asígnales un valor inicial en la propia declaración de variable.



1. Int p=2 int q=10
2. Float x=2,34 F float y=3.54353f float z=345345f
3. Char a=’a’ char b=’$’ char c=’?’
4. Doublé raiz1=2,4 Doublé raiz2=2,8
5. long contador=2435345346456546
6. short indicador=3
7. int índice=34
8. doublé precio=2.3 double precioFinal=4.5
9. char car1=’p’ char car2=’8’
10. byte valor=3
11. boolean primero=false boolean ultimo=true
12. String nombre=”paulo”

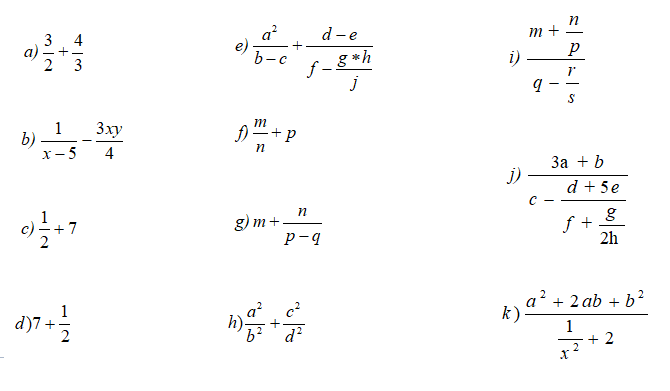
**Ejercicio 2**: Escribe la declaración más apropiada para cada una de las siguientes variables. El nombre de cada una indica el tipo de dato que contendrá y servirá para determinar el tipo de dato más adecuado en la declaración de la variable.



1. Int edad;
2. String codigoPostal;
3. Doublé altura;
4. Char genero;
5. String nombre;
6. Int numerodeHijos;
7. Doublé iva;
8. Short tallaCamisa;
9. Doublé peso;
10. Doublé precio;
11. String alumnoRepetidor;
12. String mensaje;
13. Char letra;
14. Boolean mayorEdad;
15. Int minutos
16. Int días;
17. String matricual
18. Int contador;
19. Boolean mayordeEdad;
20. String tallaCamisa;

6. Convertir expresiones algebráicas en expresiones algorítmicas

**Ejercicio:** Convierte en expresiones algorítmicas las siguientes expresiones algebráicas. Coloca paréntesis solamente donde sean necesarios.



1. (3/2)+(4/3)
2. (1/(x-5))-((3xy)/4)
3. (1/2)+7
4. 7+(1/2)
5. (a^2/(b-c))+((d-e)/[f-((g\*h)/j)]
6. (m/n)+p
7. M+(n/(p-q))
8. (a^2/b^2)+( c^2/d^2)
9. (M+(n/p))/(q-(r/s))
10. (3a+b)/c-(d+5e)/(f+(g/2h))
11. (A^2+2ab+b^2)/(1/x^2)+2

7. Convertir expresiones algorítmicas en expresiones algebráicas

**Ejercicio:** Convierte en expresiones algebráicas las siguientes expresiones algorítmicas.

**Ejercicio 1:** Si a, b y c son variables enteras con valores a=8, b=3, c=-5, determina el valor de las siguientes expresiones aritméticas:



1. 3\*x\*y-5\*x+12\*x-17 x=1,3 y=2
2. d=-2

**Ejercicio 2:**Si x, y, z son variables de tipo double con valores x= 88, y = 3.5, z = -5.2, determina el valor de las siguientes expresiones aritméticas. Obtén el resultado de cada expresión con un máximo de cuatro decimales.

|  |
| --- |
| a) x + y + z b) 2 \* y + 3 \* (x – z)  c) x / y d) x % y  e) x / (y + z) f) (x / y) + z  g) 2 \* x / 3 \* y h) 2 \* x / (3 \* y)  i) x \* y % z j) x \* (y % z)  k) 3 \* x – z – 2 \* x l) 2 \* x / 5 % y  m) x - 100 % y % z n) x - y - z \* 2 |

1. **86.3**
2. **286.6**
3. **25.14**
4. **0.5**
5. **-51.76**
6. **19.94**
7. **16.76**
8. **16.76**
9. **-2.4**
10. **308**
11. **93.2**
12. **0.2**
13. **84.68**
14. **94.9**

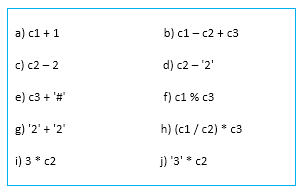
**Ejercicio 3.**Si c1, c2 y c3 son variables de tipo char con valores c1=’E’, c2=’5’, c3=’?’, determina el valor numérico de las siguientes expresiones aritméticas. Para resolverlo necesitas saber el valor numérico correspondiente a esos caracteres según la tabla ASCII:

’E’      69

’5’      53

’?’      63

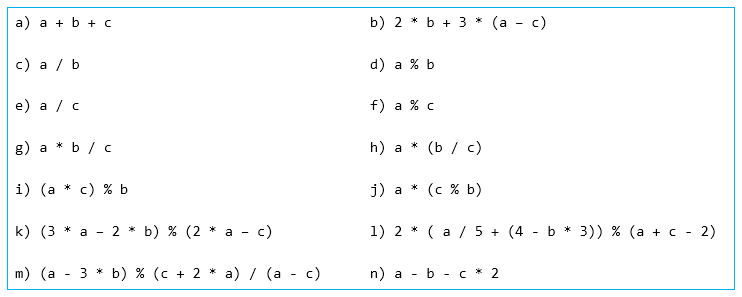
Consulta en la tabla ASCII los valores numéricos del resto de caracteres que aparecen en las operaciones.



1. 70
2. 79
3. 51
4. 3
5. 98
6. 9
7. 100
8. 81,17
9. 156
10. 2703

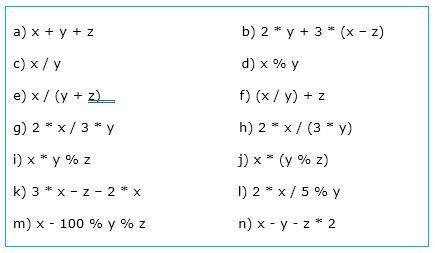
8. Ejercicio sobre operadores aritméticos

**Ejercicio 1:** Si a, b y c son variables enteras con valores a=8, b=3, c=-5, determina el valor de las siguientes expresiones aritméticas:



1. 16
2. 15
3. 2.66
4. 2
5. 1.6
6. 3
7. 4.8
8. 4.8
9. 1
10. 16
11. 0
12. -7.8
13. 6
14. -5

**Ejercicio 2:**Si x, y, z son variables de tipo double con valores x= 88, y = 3.5, z = -5.2, determina el valor de las siguientes expresiones aritméticas. Obtén el resultado de cada expresión con un máximo de cuatro decimales.



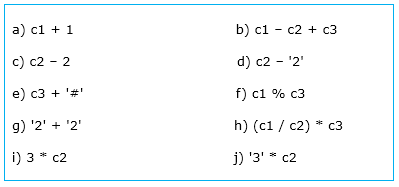
**Ejercicio 3.**Si c1, c2 y c3 son variables de tipo char con valores c1=’E’, c2=’5’, c3=’?’, determina el valor numérico de las siguientes expresiones aritméticas. Para resolverlo necesitas saber el valor numérico correspondiente a esos caracteres según la tabla ASCII:

’E’      69

’5’      53

’?’      63

Consulta en la tabla ASCII los valores numéricos del resto de caracteres que aparecen en las operaciones.



9. Determinar el tipo del resultado en operaciones aritméticas

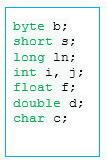
En aquellas expresiones aritméticas en las que intervienen operandos de distinto tipo (int, double, char, etc.) java convierte el tipo de los operandos que intervienen al tipo del operando de mayor precisión y este será el tipo del resultado obtenido en la operación.

Esta conversión de tipos se realiza de forma temporal, solamente para el cálculo de la operación. Los tipos originales de los operandos que intervienen siguen siendo los mismos después de la operación.

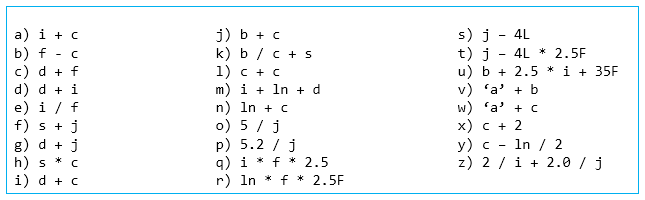
Debemos tener en cuenta que cuando se realiza la conversión temporal de tipos, los operandos de tipo short, byte y char se convierten a int.

Teniendo en cuenta todo esto, realiza el siguiente ejercicio.

**Ejercicio**: A partir de las siguientes declaraciones de variables:



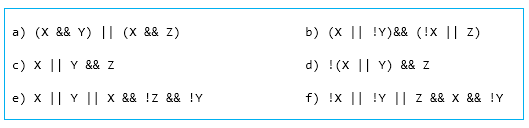
Determina cuál es el tipo de dato del resultado de las siguientes expresiones aritméticas:



1. Int
2. Float
3. Double
4. Double
5. Float
6. Int
7. Double
8. Int
9. Double
10. Int
11. Int
12. Int
13. Double
14. Long
15. Int
16. Double
17. Double
18. Float
19. long
20. float
21. double
22. int
23. int
24. int
25. int
26. double

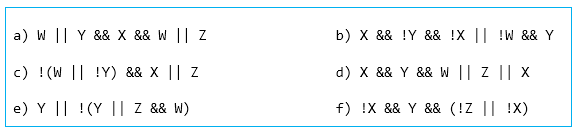
10. Ejercicio sobre operadores lógicos

**Ejercicio 1:** Si X, Y y Z son variables de tipo boolean con valores X = true, Y = false, Z = true, determina el valor de las siguientes expresiones lógicas:



1. true
2. true
3. true
4. false
5. true
6. true

**Ejercicio 2:**Si W, X, Y y Z son variables de tipo boolean con valores W = false, X = true, Y = true, Z = false, determina el valor de las siguientes expresiones lógicas:



1. false
2. true
3. true
4. true
5. true
6. false

Para resolver ejercicios con operadores lógicos tienes que conocer las tablas de verdad de estos operadores:

|  |  |  |
| --- | --- | --- |
| Operador AND | Operador OR | Operador NOT |
|  |  |  |

Como puedes observar:

* El operador lógico && (AND) da como resultado **true** cuando ambos operandos son true, en cualquier otro caso el resultado es false.
* El operador || (OR) da como resultado **false** cuando ambn cualquier otro caso el resultado es true.
* El operador ! (NOT) cambia el valor lógico del operando.

También debemos tener en cuenta para resolver estos ejercicios cuál es la precedencia de los operadores lógicos en Java:

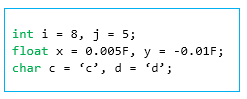
|  |
| --- |
| Precedencia de los operadores lógicos |
|  |

11. Ejercicios sobre operadores relacionales

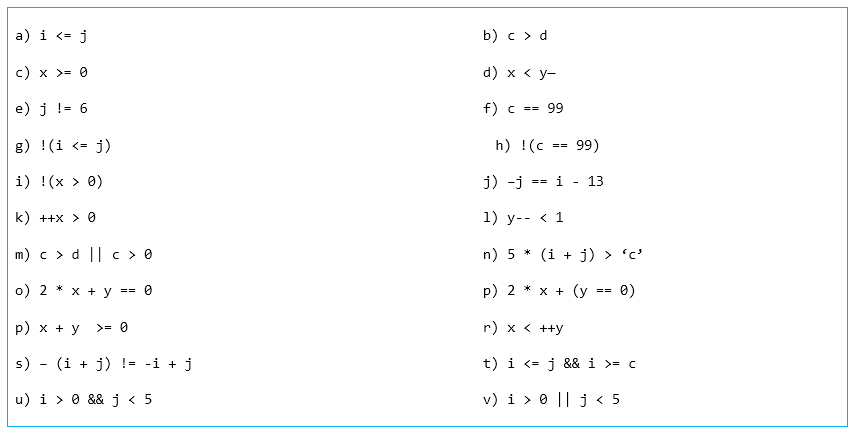
En estos ejercicios practicaremos con los operadores Java relacionales. En algunos de los ejercicios, los operadores relacionales aparecerán en expresiones en los que también intervienen el resto de operadores Java: operadores aritméticos, lógicos y los operadores incremento y decremento.

**Ejercicio:**

A partir de las siguientes declaraciones de variables:



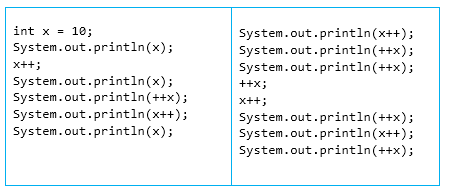
Determina el valor de las siguientes expresiones:



1. True
2. False
3. True
4. False
5. True
6. True
7. True
8. False
9. False
10. True
11. True
12. True
13. True
14. True
15. True
16. False
17. False
18. False
19. True
20. False
21. False
22. False
23. True
24. False
25. False
26. true

12. Ejercicio con los operadores incremento y decremento: ++ y --.

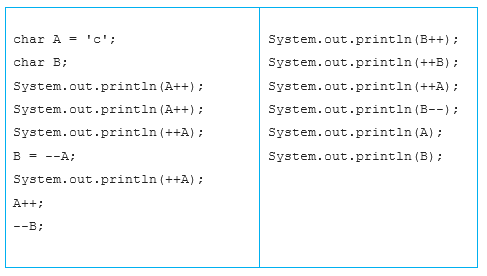
**Ejercicio 1:** Si x es una variable de tipo int con valor 10, determina qué se muestra por pantalla cuando se ejecutan las siguientes instrucciones:



Respuestas(De cada print)

1. 10
2. 11
3. 12
4. 12
5. 13
6. 13
7. 15
8. 16
9. 19
10. 19
11. 21

**Ejercicio 2:**Dadas las variables A y B de tipo char,calcula qué se muestra por pantalla cuando se ejecutan las siguientes instrucciones**:**



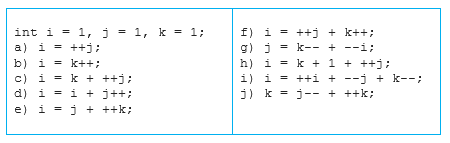
Respuestas(De cada print)

1. ‘c’
2. ‘d’
3. ‘f’
4. ‘f’
5. ‘g’
6. ‘d’
7. ‘f’
8. ‘h’
9. ‘f’
10. ‘h’
11. ‘e’

**Ejercicio 3:**Un programa Java contiene las siguientes declaraciones y asignaciones iniciales:

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

Determina el valor final de las tres variables en cada una de las siguientes instrucciones. Las instrucciones son independientes unas de otras, es decir, el valor inicial de las variables en cada instrucción es i = 1, j = 1, k = 1.



Repuestas

1. I=2,j=2,k=1
2. I=1,j=1,k=2
3. I=3,j=2,k=1
4. I=2,j=2,k=1
5. I=3,j=1,k=2
6. I=3,j=2,k=2
7. I=0,j=1,k=0
8. I=4,j=2,k=1
9. I=3,j=0,k=0
10. I=1,j=0,k=3

13. Ejercicios para practicar la salida de datos por pantalla: System.out.print y System.out.println – I

Relación Nº 1: Ejercicios 1 y 2  
  
Empezamos con una serie de ejercicios para aquellos que están comenzando desde cero con el lenguaje Java y tampoco tienen experiencia con ningún otro lenguaje de programación.

En estos ejercicios básicos inicales se realizan las siguientes instrucciones: declarar variables, asignarles un valor, operar con ellas y mostrar resultados por pantalla.

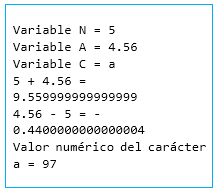
Estos ejercicios son todos de estructura secuencial, es decir, no hay condiciones ni bucles. Tampoco se lee nada por teclado. El objetivo es familiarizarse con la declaración de variables y practicar la salida por consola utilizando los métodos print y println.

**Ejercicio básico inicial 1**:

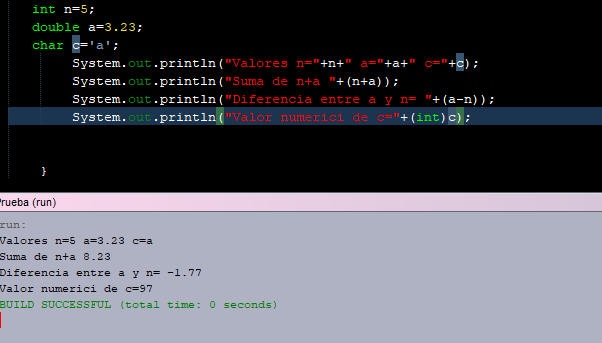
Escribe un programa Java que realice lo siguiente: declarar una variable N de tipo int, una variable A de tipo double y una variable C de tipo char y asigna a cada una un valor. A continuación muestra por pantalla:

* El valor de cada variable.
* La suma de N + A
* La diferencia de A - N
* El valor numérico correspondiente al carácter que contiene la variable C.

Si por ejemplo le hemos dado a N el valor 5, a A el valor 4.56 y a C el valor ‘a’, se debe mostrar por pantalla:

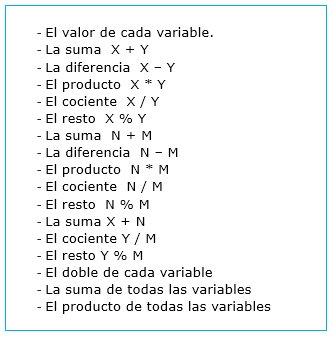
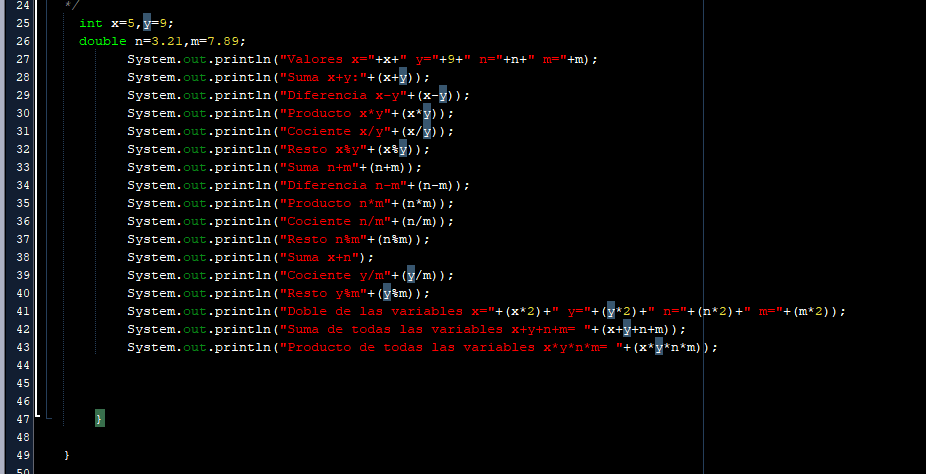


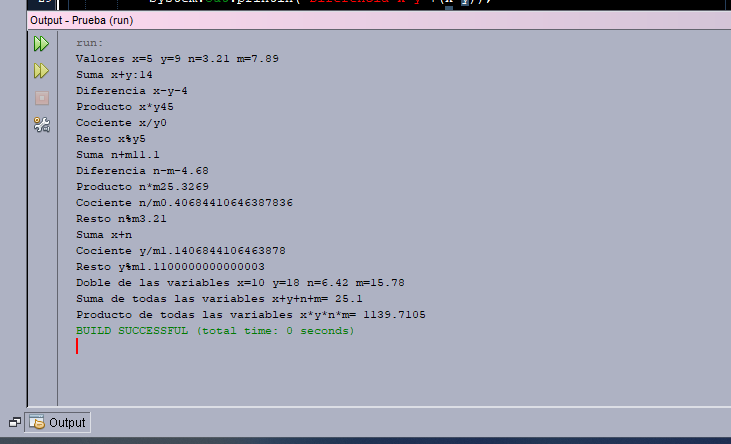
Repuesta:



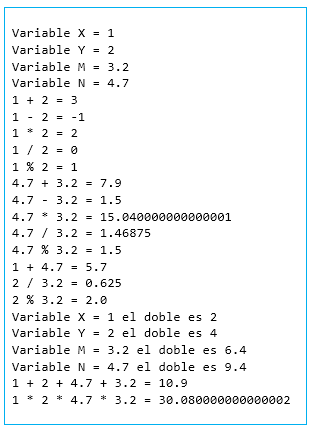
**Ejercicio básico inicial 2**:

Escribe un programa Java que realice lo siguiente: declarar dos variables X e Y de tipo int, dos variables N y M de tipo double y asigna a cada una un valor. A continuación muestra por pantalla:

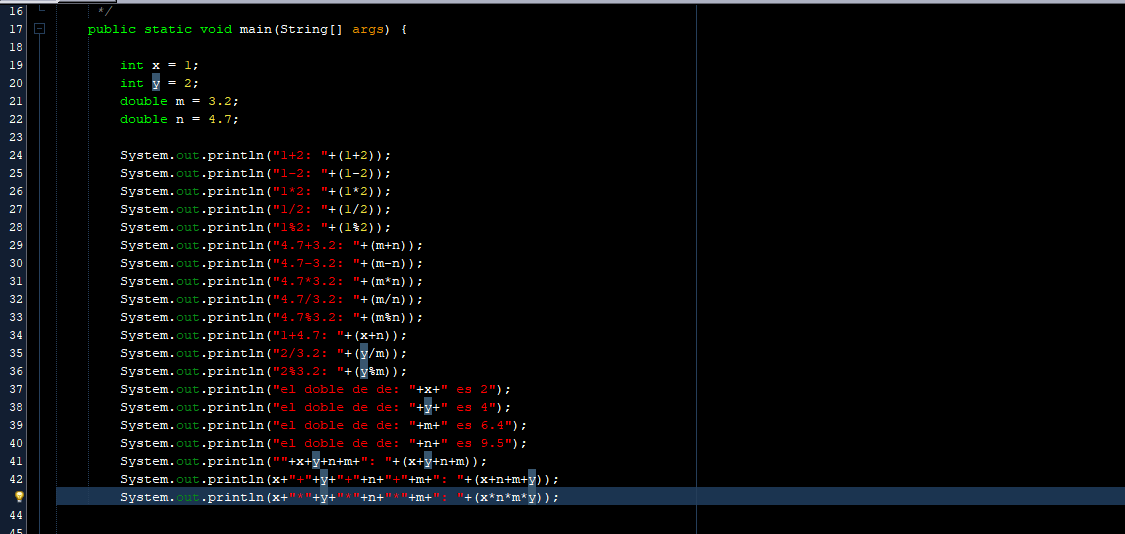
 

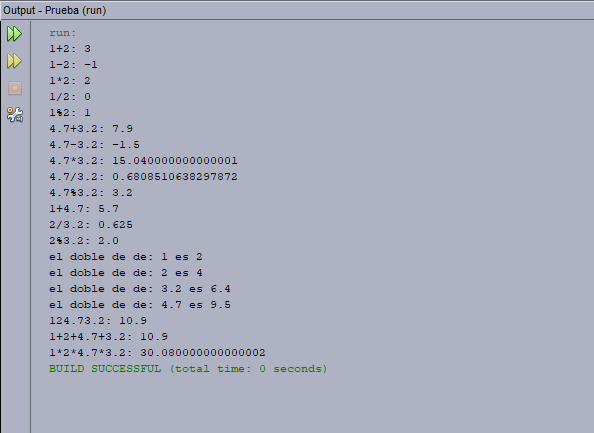


Si por ejemplo le hemos dado a X el valor 1, a Y el valor 2, a M el valor 3.2 y a N el valor 4.7 se debe mostrar por pantalla:



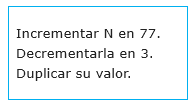
14. Ejercicios para practicar la salida de datos por pantalla: System.out.print y System.out.println – II



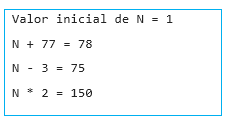


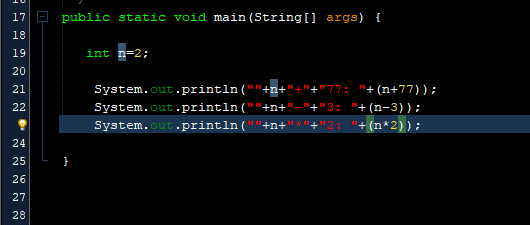
**Ejercicio básico inicial 3**

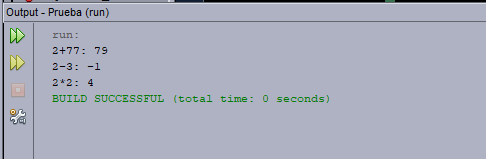
Escribe un programa Java que declare una variable entera N y asígnale un valor. A continuación escribe las instrucciones que realicen los siguientes:



Si por ejemplo N = 1 la salida del programa será:

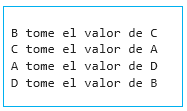


****

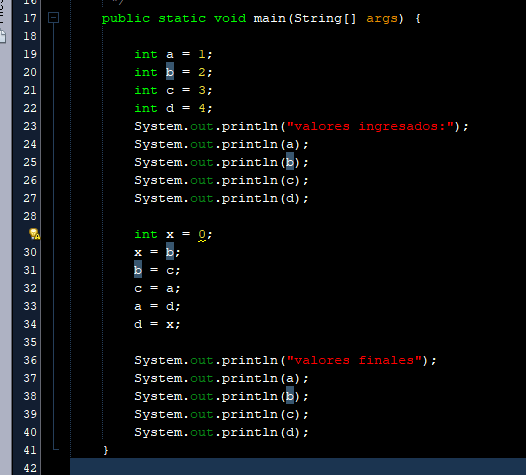
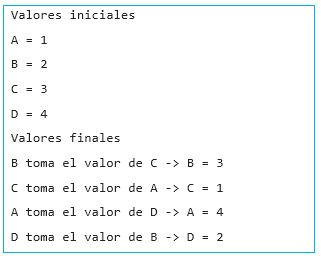
****

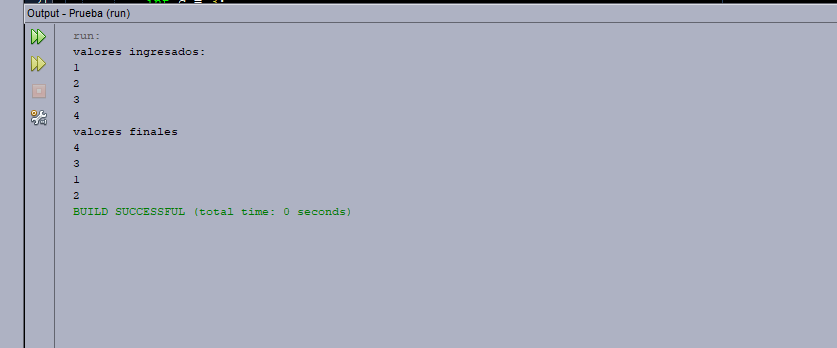
**Ejercicio básico inicial 4**

Programa java que declare cuatro variables enteras A, B, C y D y asígnale un valor a cada una. A continuación realiza las instrucciones necesarias para que:



Si por ejemplo A = 1, B = 2, C = 3 y D = 4 el programa debe mostrar:





15. Ejercicios con el operador condicional ( ? : ) - I

En esta entrada vamos a ver tres ejemplos de utilización del operador condicional ? :  
Se trata de usar el operador condicional en lugar de la instrucción condicional if para mostrar por pantalla un mensaje u otro dependiendo de una condición.  
  
**Ejercicio básico inicial 5**

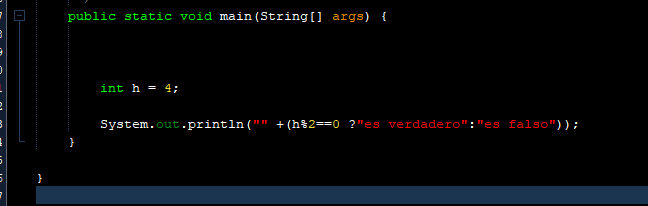
Escribe un programa java que declare una variable A de tipo entero y asígnale un valor. A continuación muestra un mensaje indicando si A es par o impar. Utiliza el operador condicional ( ? : ) dentro del println para resolverlo.

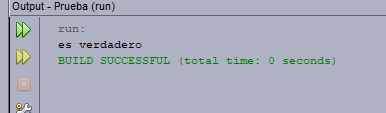
Si por ejemplo A = 14 la salida será



Si fuese por ejemplo A = 15 la salida será:







**Ejercicio básico inicial 6**

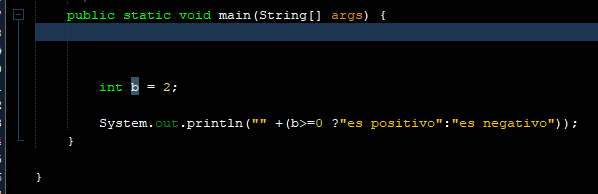
Escribe un programa java que declare una variable B de tipo entero y asígnale un valor. A continuación muestra un mensaje indicando si el valor de B es positivo o negativo. Consideraremos el 0 como positivo. Utiliza el operador condicional ( ? : ) dentro del println para resolverlo.

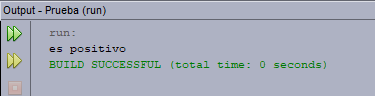
Si por ejemplo B = 1 la salida será



Si fuese por ejemplo B = -1 la salida será:



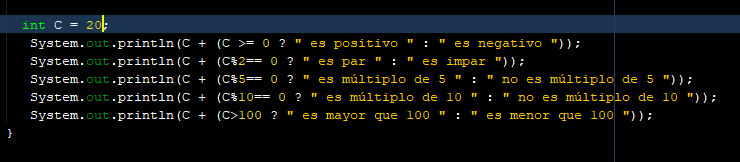


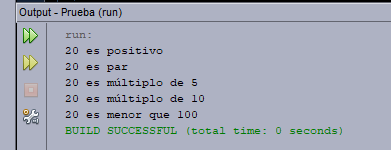


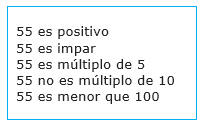
**Ejercicio básico inicial 7**

Escribe un programa java que declare una variable C de tipo entero y asígnale un valor. A continuación muestra un mensaje indicando si el valor de C es positivo o negativo, si es par o impar, si es múltiplo de 5, si es múltiplo de 10 y si es mayor o menor que 100. Consideraremos el 0 como positivo. Utiliza el operador condicional ( ? : ) dentro del println para resolverlo.

Si por ejemplo C = 55 la salida será

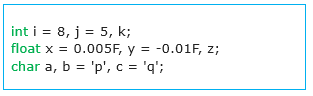




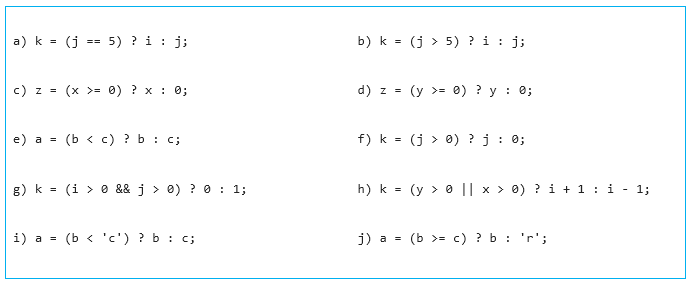


16. Ejercicios con el operador condicional ( ? : ) – II

**Ejercicio 1:**Un programa contiene las siguientes declaraciones y asignaciones iniciales de variables:



Determina el valor de cada una de las siguientes expresiones:



a) k = 8

b) k = 5

c) z = 0.005

d) z = 0

e) a = 'p'

f) k = 5

g) k = 0

h) k = 9

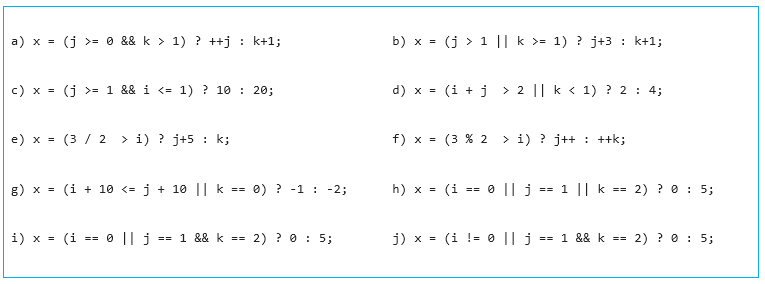
i) a = 'q'

j) a = 'r'

**Ejercicio 2:**Un programa contiene las siguientes declaraciones y asignaciones iniciales de variables:



Determina el valor que toma la variable x al ejecutarse cada una de las siguientes expresiones. Las instrucciones son independientes unas de otras, es decir, el valor inicial para las variables i, j, k en cada instrucción es i = 1, j = 1, k = 1.



a) x = 2

b) x = 4

c) x = 10

d) x = 4

e) x = 1

f) x = 2

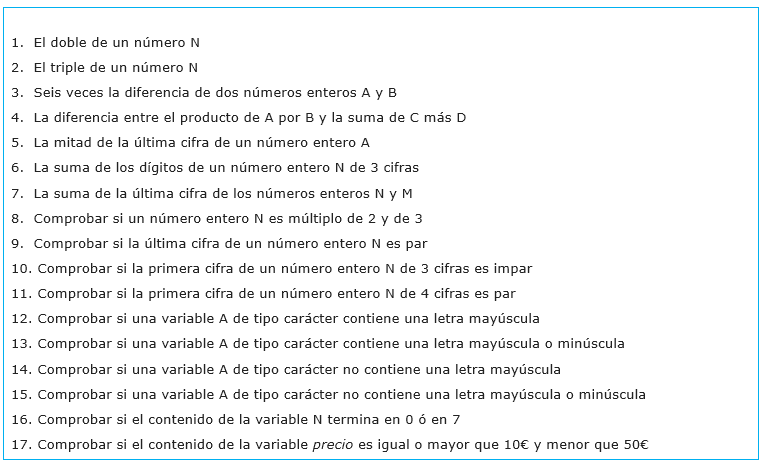
g) x = -1

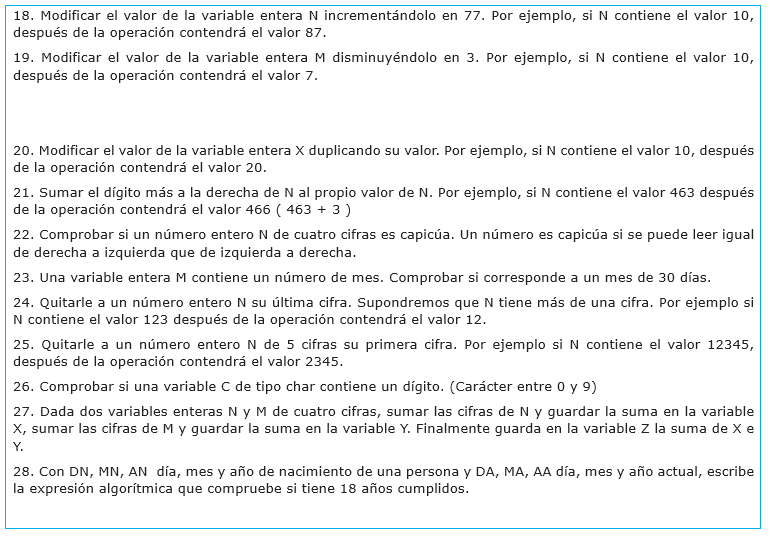
h) x = 0

i) x = 5

j) x = 0

17. Ejercicios sobre expresiones algorítmicas





1.  El doble de un número N

2 \* N

2.  El triple de un número N

3 \* N

3.  Seis veces la diferencia de dos números enteros A y B

6 \* (A – B)

4.  La diferencia entre el producto de A por B y la suma de C más D

A \* B – ( C + D )

5.  La mitad de la última cifra de un número entero A

A % 10 / 2

6.  La suma de los dígitos de un número entero N de 3 cifras

N/100 + N%10 + N/10%10

7.  La suma de la última cifra de los números enteros N y M

N%10 + M%10

8.  Comprobar si un número entero N es múltiplo de 2 y de 3

N%2 == 0 && N%3 == 0

9.  Comprobar si la última cifra de un número entero N es par

N%10%2 == 0

10. Comprobar si la primera cifra de un número entero N de 3 cifras es impar

N/100%2 != 0   También es válida esta solución ->  N/100%2 == 1

11. Comprobar si la primera cifra de un número entero N de 4 cifras es par

N/1000%2 == 0

12. Comprobar si una variable A de tipo carácter contiene una letra mayúscula

A >= ’A’ && A <= ’Z’

13. Comprobar si una variable A de tipo carácter contiene una letra mayúscula o minúscula

A >= ’A’ && A <= ’Z’ || A >= ’a’ && A <= ’z’

14. Comprobar si una variable A de tipo carácter no contiene una letra mayúscula

A < ’A’ || A > ’Z’

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

(A < ’A’ || A > ’Z’) && (A < ’a’ || A > ’z’)

También es válida esta solución:

!(A >= ’A’ && A <= ’Z’ || A >= ’a’ && A <= ’z’)

16. Comprobar si el contenido de la variable N termina en 0 ó en 7

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

17. Comprobar si el contenido de la variable *precio* es igual o mayor que 10€ y menor que 50€

precio >= 10 && precio < 50

18. Modificar el valor de la variable entera N incrementándolo en 77.

Por ejemplo, si N contiene el valor 10, después de la operación contendrá el valor 87.

N = N +77   ó   N+=77

19. Modificar el valor de la variable entera M disminuyéndolo en 3.

Por ejemplo, si N contiene el valor 10, después de la operación contendrá el valor 7.

N = N – 3   ó   N-=3

20. Modificar el valor de la variable entera X duplicando su valor.

Por ejemplo, si N contiene el valor 10, después de la operación contendrá el valor 20.

N = N \* 2   ó   N\*=2

21. Sumar el dígito más a la derecha de N al propio valor de N.

Por ejemplo, si N contiene el valor 463 después de la operación contendrá el valor 466 ( 463 + 3 )

N = N + N%10   ó   N+=N%10

22. Comprobar si un número entero N de cuatro cifras es capicúa.

Un número es capicúa si se puede leer igual de derecha a izquierda que de izquierda a derecha.

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

23. 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

24. 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

25. 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

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

C >= ‘0’ && C <= ‘9’

27. Dada dos variables enteras N y M de cuatro cifras, sumar las cifras de N y guardar la suma en la variable X, sumar las cifras de M y guardar la suma en la variable Y. Finalmente guarda en la variable Z la suma de X e Y.

X = N/1000 + N/100%10 + N/10%10 + N%10

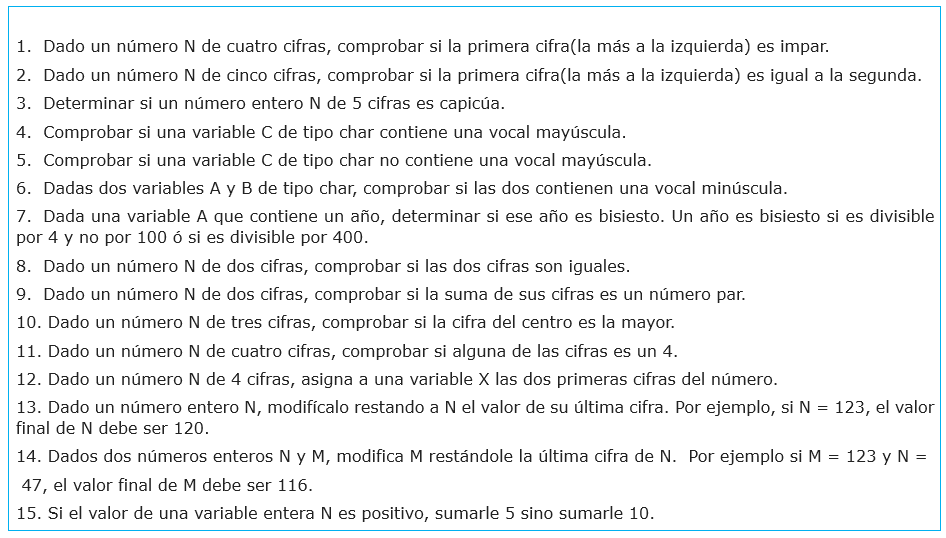
Y = M/1000 + M/100%10 + M/10%10 + M%10

Z = X + Y

28. A partir de las variables enteras DN, MN y AN que contienen el día, mes y año de nacimiento de una persona y las variables DA, MA y AA que contienen el día, mes y año actual, escribe la expresión algorítmica que compruebe si esa persona ya ha cumplido los 18 años.

AA – AN > 18 || AA – AN == 18 && MA > MN || AA – AN == 18 && MA == MN && DA >= DN

18. Ejercicios sobre expresiones algorítmicas – II

Escribe las expresiones algorítmicas equivalentes para cada uno de los siguientes enunciados:

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

N / 1000 % 2 == 0

2.  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

3.  Determinar si un número entero N de 5 cifras es capicúa

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

4.  Comprobar si una variable C de tipo char contiene una vocal mayúscula

C == ‘A’ || C == ‘E’ || C == ‘I’ || C == ‘O’ || C == ‘U’

5.  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’

6.  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’)

7.  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

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

N / 10 == N % 10

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

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

10. 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

11. 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

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

X = N / 100

13. 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

14. 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

15. Si el valor de una variable entera N es positivo, sumarle 5 sino sumarle 10

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

16. Si el valor de una variable entera N es negativo, asignarle el valor 5 sino asignarle el valor 100

N = N < 0 ? 5 : 100;