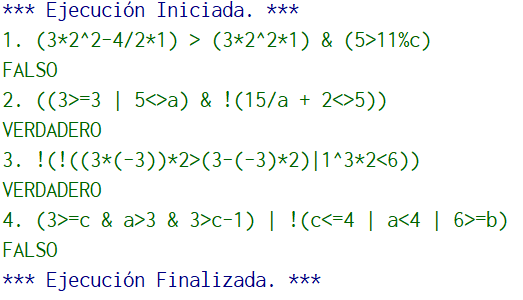
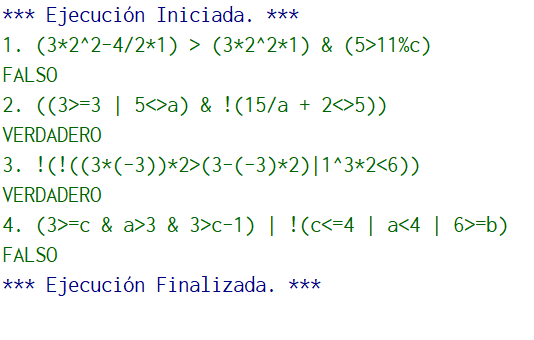
**Relación 1**

**Expresiones aritmetico-logicas**

**1. Evalúa las siguientes expresiones aritmético-lógicas. a= 5, b=7, c=4. (NOTA. el operador ^ es el operador de exponenciación, es decir, por ejemplo a ^ 3 sería “a elevado a 3”. En el orden de precedencia, el operador ^ se evalúa después de los paréntesis pero antes de la multiplicación, división y módulo).**

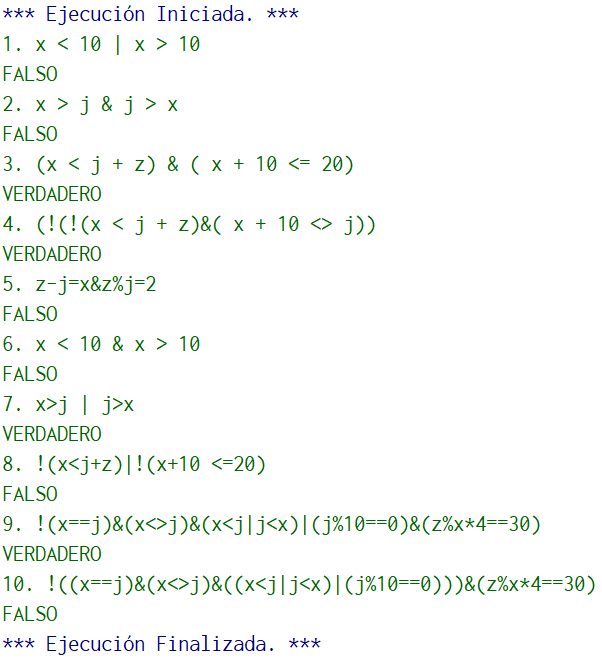
1. ( 3 \* 2 ^ 2 – 4 / 2 \* 1 ) > ( 3 \* 2 ^ 2 \* 1 ) AND ( 5 > 11 MOD c )
2. ( 3 >= 3 OR 5 <> a ) AND NOT ( 15 / a + 2 <> 5 )
3. NOT ( NOT ( ( 3 ( - 3 ) ) \* 2 > ( 3 – ( - 3 ) \* 2 ) OR 1 ^ 3 \* 2 > 6 ) )
4. ( 3 >= c AND a > 3 AND 3 > c-1 ) OR NOT ( c <= 4 OR a > 4 OR 6 >= b )





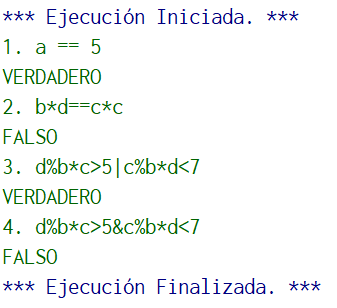
**2. Evalúa las siguientes expresiones aritmético-lógicas, siendo x=10, y=20, and z=30. Ten en cuenta que algunas están escritas en notación de pseudocódigo pero otras lo están en lenguaje de programación C (o Java). Indica si alguna de las expresiones es siempre verdadera o siempre falsa independientemente de los valores de x, y, z. (NOTA. El operador % es – en C y Java- el operador “módulo” o “resto de dividir de forma entera el primer operando entre el segundo”)**

1. x < 10 OR x > 10
2. x > y && y > x
3. (x < y + z) AND ( x + 10 <= 20)
4. NOT (NOT (x < y + z) AND ( x + 10 <> y))
5. z – y = x AND z MOD y = 2
6. x < 10 && x > 10
7. x > y || y > x
8. !(x < y+ z) || !(x+10 <=20)
9. !(x == y) && (x!= y) && (x<y || y<x) || (y % 10 ==0) && ( z % x\*4 == 30)
10. !((x == y) && (x!= y) && ((x<y || y<x) || (y % 10 ==0))) && ( z % x\*4 == 30)



**3. Dado que a = 5, b = 2, c = 4, y d = 5, cuál es el resultado de las siguientes expresiones en Java?**

1. a == 5
2. b \* d == c \* c
3. d % b \* c > 5 || c % b \* d < 7
4. d % b \* c > 5 && c % b \* d < 7



**4. Dado que: a = 5, b = 2, c = 4, d = 6, e = 3. ¿Cuál es el resultado de las siguientes expresiones?**

1. a > b
2. a != b
3. d % b == c % b
4. a \* c != d \* b
5. d \* b == c \* e
6. a \* b < a % b \* c
7. c % b \* a == b % c \* a
8. b % c \* a != a \* b
9. d % b \* c > 5 || c % b \* d < 7

