|  |  |  |
| --- | --- | --- |
| Texto  Descripción generada automáticamente con confianza media | TECNICATURA UNIVERSITARIA EN DISEÑO INTEGRAL DE VIDEOJUEGOS  FACULTAD DE INGENIERÍA  Universidad Nacional de Jujuy |  |

Trabajo Práctico N°1

*FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A OBJETOS*

*FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A OBJETOS*



Apellido y Nombre

Córdoba Agustina Ayelén

LU / TUV000471

*Profesores:*

*Mg. Ing. Ariel Alejandro Vega*

*Año 2024*

indice

*Ejercicio 1:*

Evaluar (obtener resultado) la siguiente expresión para A = 2 y B = 5

Un dibujo de un perro

Descripción generada automáticamente con confianza media

3\* A - 4 \* B / A ^ 2

Resolución necesaria en Word:

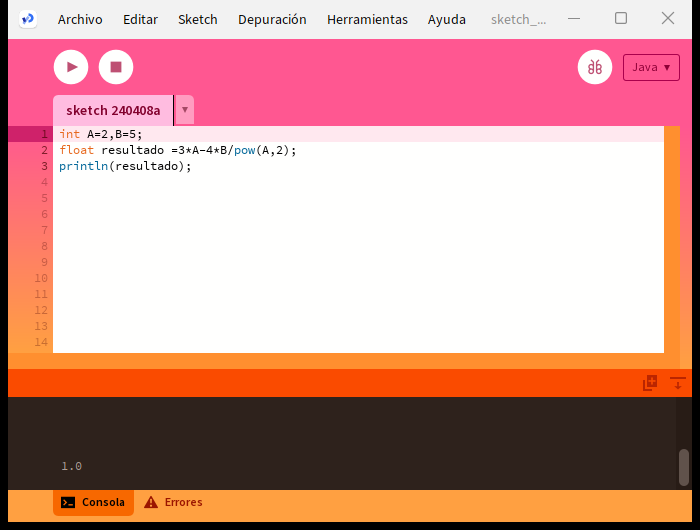
(3\*A) -(4\*B/(A^2))

6-(4\*B/4)

6-5

1

*Captura de Processing*



*Ejercicio 2:*

Evaluar la siguiente expresión



(4 / 2 \* 3 / 6) +( 6 / 2 / 1 / 5 ^ 2 / 4 \* 2)

1 + 3/1/25

1 + 3/25

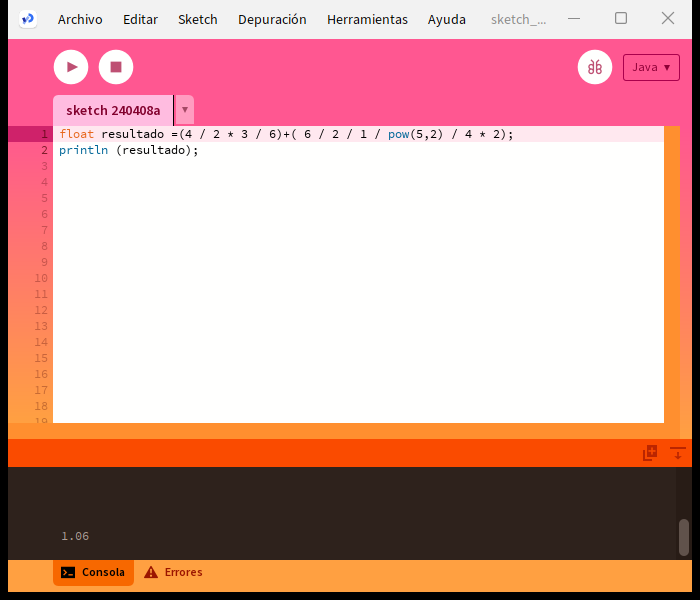
1 + 0,12/4

1 + 0,3\*2

1 + 0,06

1,06

*Captura de Processing*



*Ejercicio 3:*

Escribir las siguientes expresiones algebraicas como expresiones algorítmicas

(en su forma aritmética dentro del algoritmo). En este caso no se pide evaluarlas ni

programarlas.

*Ejercicio 4:*

Evaluar las siguientes expresiones aritméticas, para lo cual indicar en el caso

de las variables, el valor indicado. Luego escribirlas como expresiones algebraicas.

a) b ^ 2 – 4 \* a \* c

b) 3 \* X ^ 4 – 5 \* X ^ 3 + X 12 – 17

c) (b + d) / (c + 4)

d) (x ^ 2 + y ^ 2) ^ (1 / 2)

Para aclarar que indicamos con” Luego escribirlas como expresiones algebraicas” lo

aplicamos con el punto a)

𝑏2 − 4. 𝑎. c

*Ejercicio 5:*

Si el valor de A es 4, el valor de B es 5 y el valor de C es 1, evaluar las siguientes

expresiones:

1. (B \* A) – (B ^ 2 / 4 \* C)

20 - 25/4 \*C

20 - 6,25 \*C

20 - 6,25

13,75

b) (A \* B) / 3 ^ 2

20 / 9

2,22



c) (((B + C) / 2 \* A + 10) \* 3 \* B) – 6

((6/2\*A+10) \*3\*B) -6

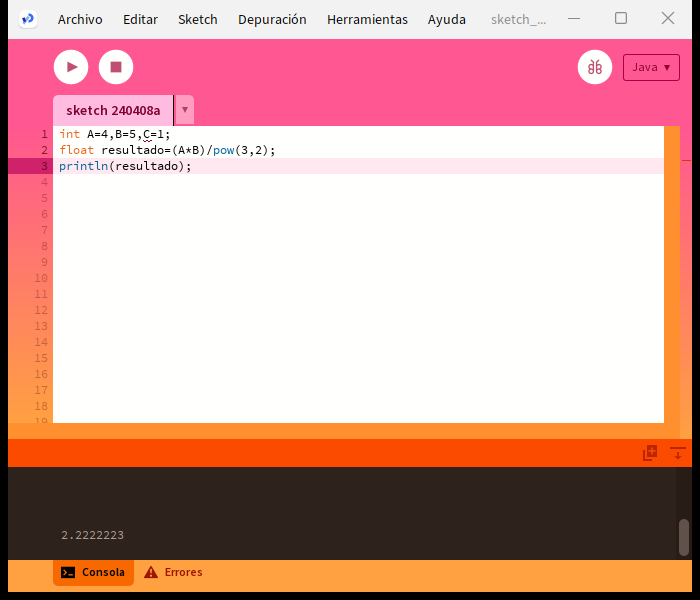
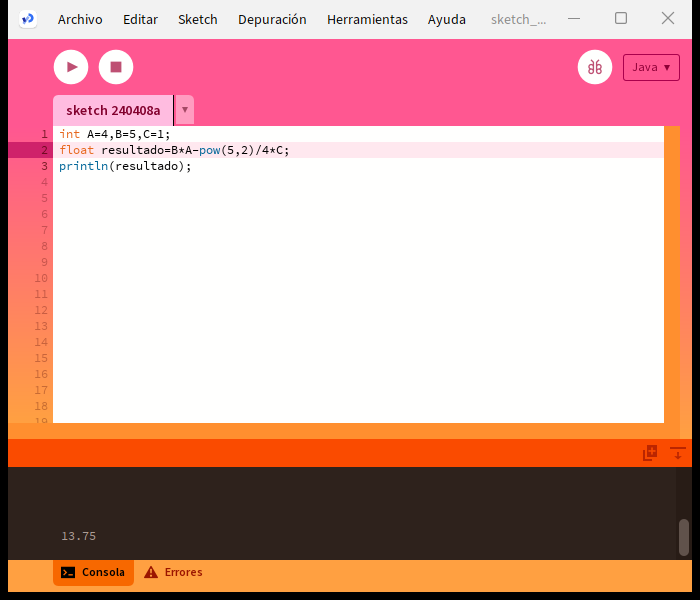
((3\*A+10) \*3\*B)-6

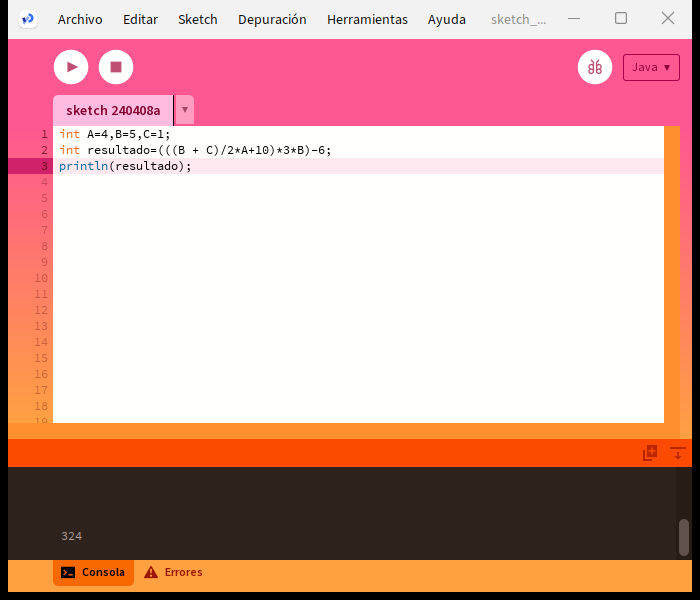
((12+10) \*3\*B) -6

(22\*3\*B)-6

(66\*B)-6

330-6=324

***Captura de Processing*

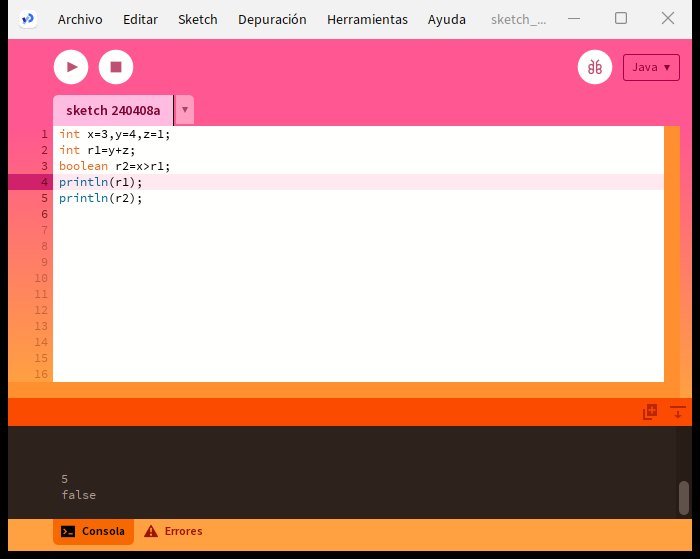


*Ejercicio 6:*

Para x=3, y=4; z=1, evaluar el resultado de R1 = y+z R2 = x >= R

R1=4+1=5

R2=3>5 =Falso

*Captura de Processing*

*Ejercicio 7*

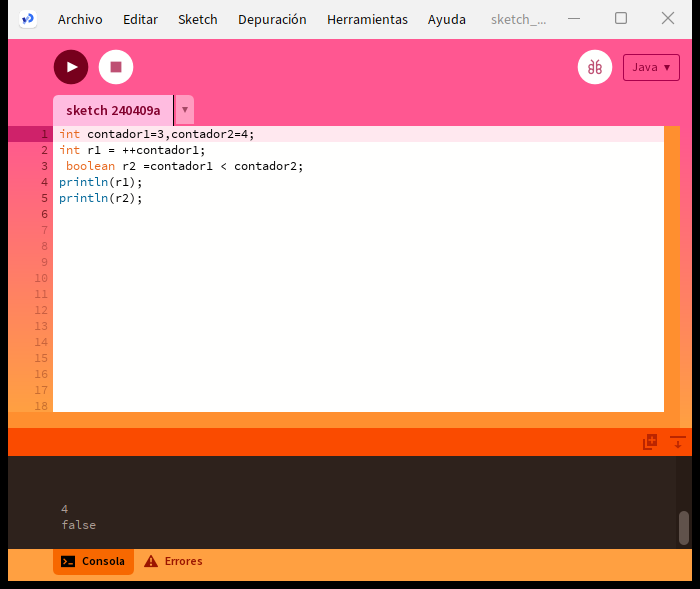
 Para contador1=3, contador2=4, evaluar el resultado de

R1 = ++contador1

R2 = contador1 < contador2

R1 = 1+3 = 4

R2 = 4 < 4 = False

*Captura de Processing*

Ejercicio 8

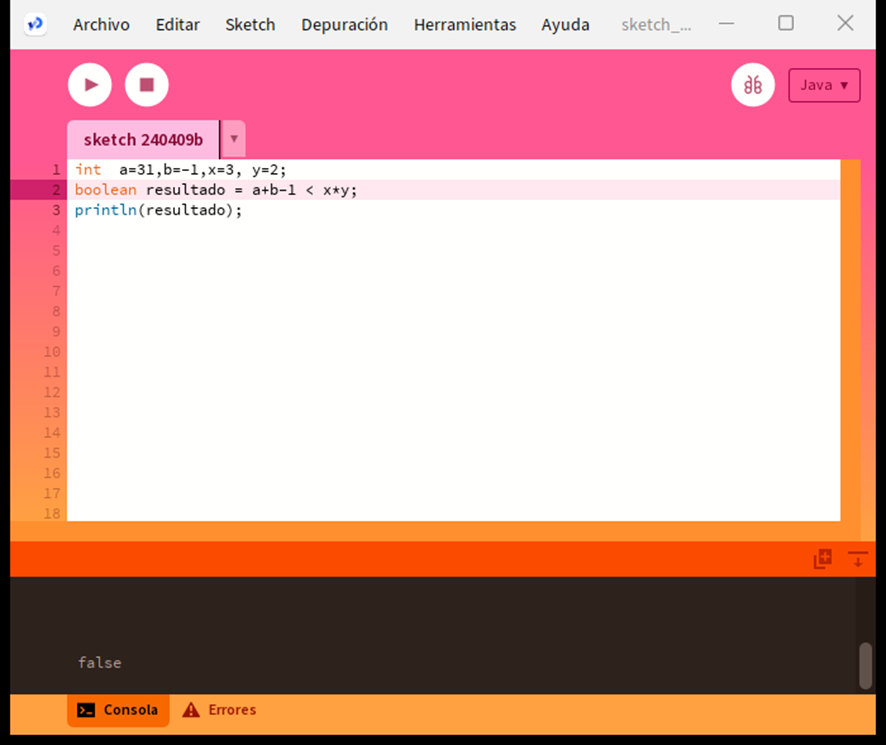
Para a=31, b=-1; x=3, y=2, evaluar el resultado de



a+b-1 < x\*y

30-1<6

29<6 = false

*Captura de Processing*

*Ejercicio 09*

Para , evaluar el resultado de



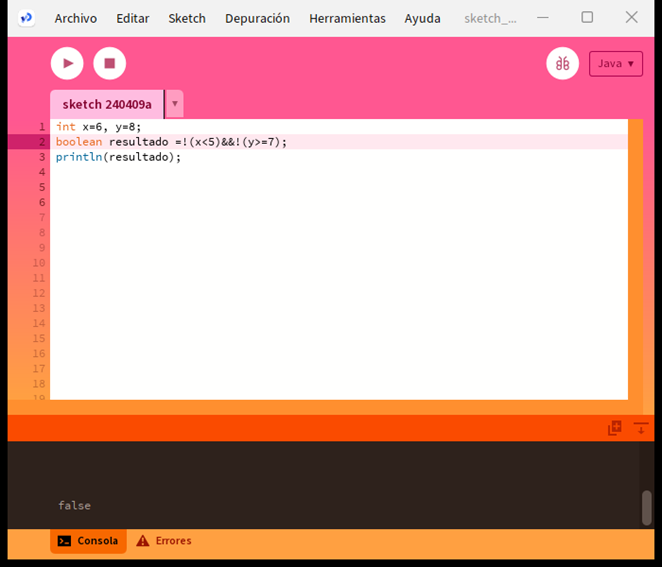
!(x<5)&&!(y>=7)

!(6<5)&&!(8>=7)

!(Falso)&&!(verdadero)

Verdadero && Falso=Falso

*Captura de Processing*



*Ejercicio 10*

Para i=22, j=3, evaluar el resultado de

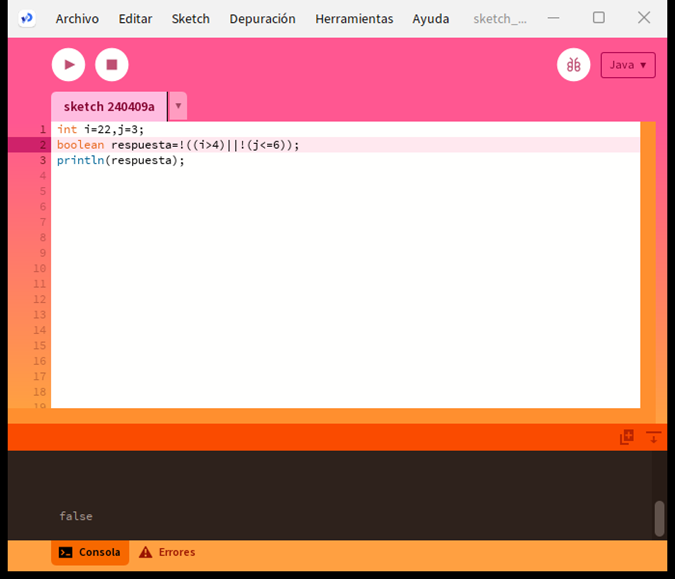
! ((i>4) ||! (j<=6))

! ((22>4)||!(3<=6))

! (Verdadero)||!(Verdadero)

Falso

*Captura de Processing*



*Ejercicio 11*

Para a=34, b=12, c=8, evaluar el resultado de

! (a+b==c) || (c! = 0) &&(b-c>=19)

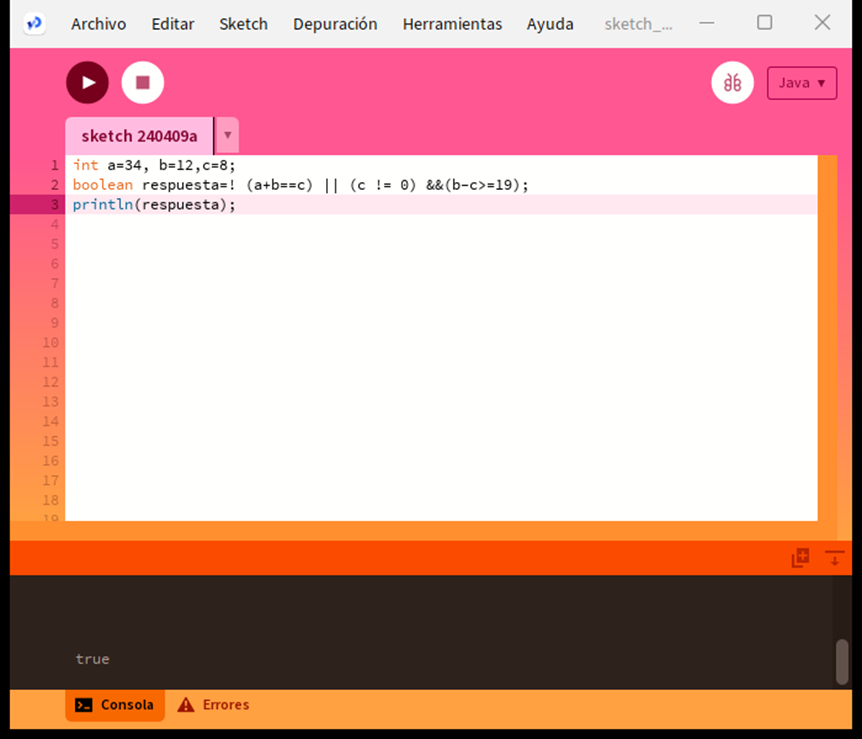
! (34+12==8)||(8!=0)&&(12-8>=19)

!(46==8)||Verdadero&&(4>=19)

!(Falso)||Verdadero&& Falso

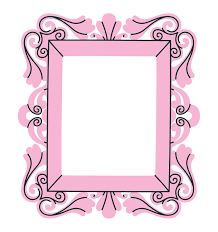
Verdadero||Falso=Verdadero

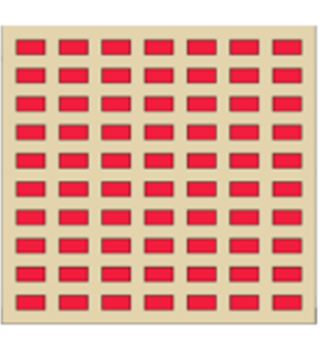
*Captura de Processing*



EJERCICIO 20

Dibuje en toda la extensión del lienzo de (440, 420) rectángulos de idénticas medidas (40 ancho y 20 de alto) y que mantengan una distancia de 20 pixeles entre ellos tanto horizontal como verticalmente. Utilice la estructura de control repetitiva for. El lienzo debería verse así:





**PLANTILLA EJERCICIOS**

**Definición del Problema: Dibujar en toda la extensión del lienzo rectángulos de idénticas medidas**

**Análisis:**

* Datos de Entrada:
* Entero: anchoRectangulo
* Entero: altoRectangulo
* Entero: distanciaEntreRectangulo;
* PVector posición
* Datos de Salida:
* dibujarRectangulo
* Proceso:

¿Quién debe realizar el proceso?: Programador

¿Cuál es el proceso que realiza …?

Dibujar rectángulos de idénticas medidas usando la ESTRUCTURA ITERATIVA PARA (For)

|  |
| --- |
| ENTIDAD QUE RESUELVE EL PROBLEMA :**Programador** |
| **VARIABLES**  anchoRectangulo:Entero//almacena el ancho del rectangulo  altoRectangulo:Entero// almacena el alto del rectangulo  distanciaEntreRectangulo: Entero//almacena la distancia entre los rectangolos  posición : PVector // almacena la posición x e y del rectángulo |
| NOMBRE ALGORITMO: dibujar\_Rectangulo  PROCESO DEL ALGORITMO   1. Inicio 2. anchoRectangulo 🡨 40 3. altoRectangulo🡨 20 4. distanciaEntreRectangulo 🡨 20 5. Posición 🡨distanciaEntreRectangulo, distanciaEntreRectangulo 6. Para x 🡨 posicion.x hasta x<width incremento x+🡨(anchoRectangulo+distanciaEntreRectangulo)) 7. Para y 🡨posicion.y hasta y<heigth incremento y+🡨(altoRectangulo+distanciaEntreRectangulo)) 8. rect(x,y,anchoRectangulo,altoRectangulo) 9. fin\_para 10. fin para 11. fin |

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente*Captura de Processing*

