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

Trabajo Práctico/Actividad

N° 1

Apellido y Nombre:

Sanchez, Santiago Carlos Ezequiel

LU: TUV000774

*Profesores:*

*Mg. Ing. Ariel Alejandro Vega*

*Primer Año*

**Desarrollo:**

Ejercicio 1: Evaluar(obtener resultado) la siguiente expresión para A = 2 y B = 5

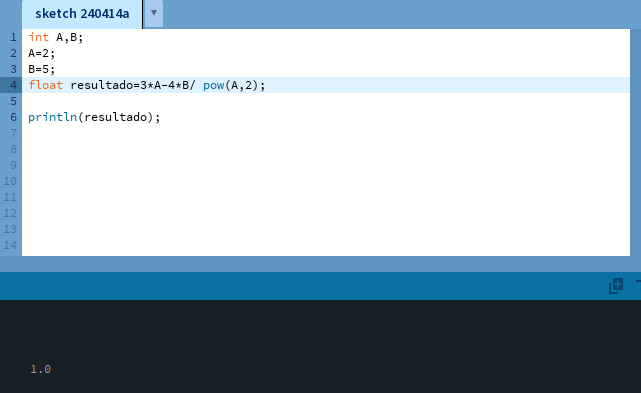
3\* A - 4 \* B / A ^ 2

Resolución necesaria en Word:

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

6-(4\*B/4)

6-5

1

Ejercicio 2: Evaluar la siguiente expresión 4 / 2 \* 3 / 6 + 6 / 2 / 1 / 5 ^ 2 / 4 \* 2

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

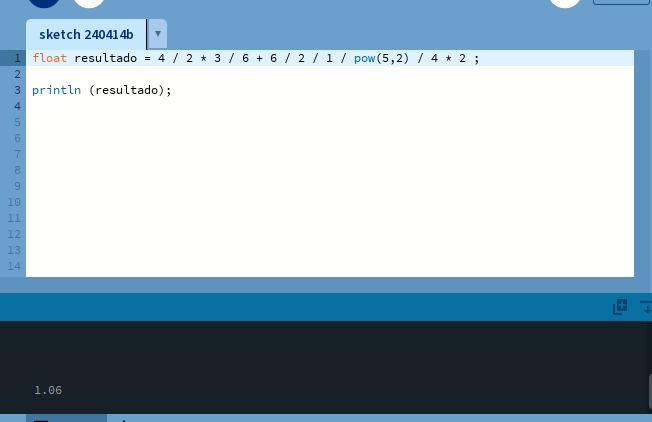
(4 / 2 \* 3 / 6) + (6 / 2 / 1 / (5 ^ 2) / 4 \* 2)

(4 / 2 \* 3 / 6) + (6 / 2 / 1 / 25 / 4 \* 2)

1 + (6 / 2 / 1 / 25 / 4 \* 2)

1 + 0,06

1,06

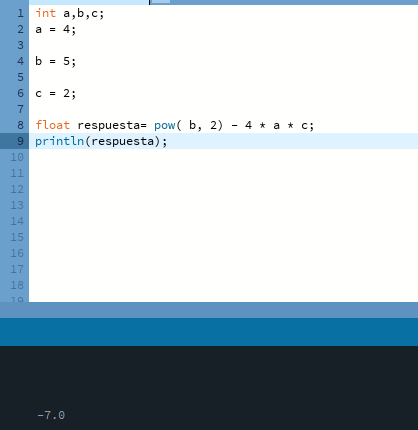


Ejercicio 4: Evaluar las siguientes expresiones aritméticas, para lo cual indicar en el caso

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

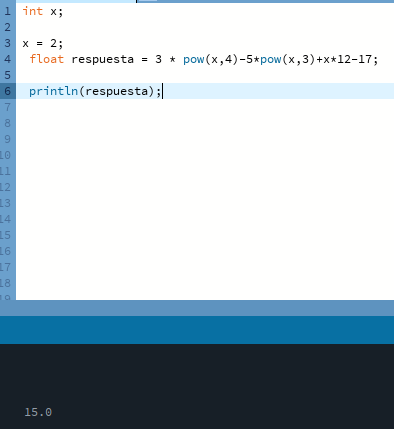
a) b ^ 2 – 4 \* a \* c a = 4 b = 5 c = 2

|  |  |
| --- | --- |
| Expresión aritmetica | Expresión algebraica |
| (b ^ 2) – (4 \* a \* c)  25 – (4 \* a \* c)  25 – 32  -7 |  |



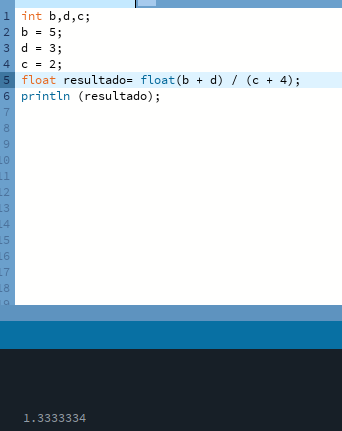
b) 3 \* X ^ 4 – 5 \* X ^ 3 + X 12 – 17 x = 2

|  |  |
| --- | --- |
| Expresión aritmética | Expresión algebraica |
| (3 \* (x ^ 4)) – (5 \* (x ^ 3)) +( x \* 12) - 17  (3 \* 16) - (5 \* 8) + (x \* 12) – 17  48 - 40 + 24 – 17  15 |  |



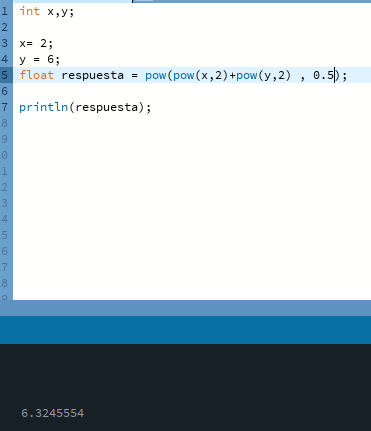
c) (b + d) / (c + 4) b = 5 d = 3 c = 2

|  |  |
| --- | --- |
| Expresión aritmetica | Expresión algebraica |
| (b + d) / (c + 4)  8 / 6  1,333333 |  |



d) (x ^ 2 + y ^ 2) ^ (1 / 2) x = 2 y = 6

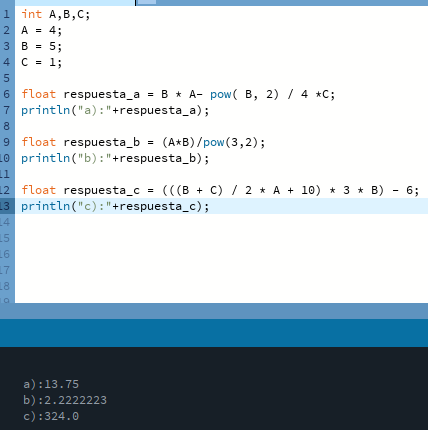
|  |  |
| --- | --- |
| Expresión aritmetica | Expresión algebraica |
| ((x ^ 2) + (y ^ 2)) ^ (1 / 2)  (4 + 36) ^ (1 / 2)  40 ^ (1 / 2)  40 ^ 0,5  6,324555 |  |



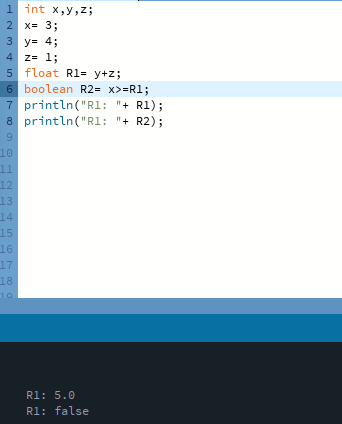
Ejercicio 5: Si el valor de A es 4, el valor de B es 5 y el valor de C es 1, evaluar las siguientes

expresiones:

|  |  |  |
| --- | --- | --- |
| a) B \* A – B ^ 2 / 4 \* C | b) (A \* B) / 3 ^ 2 | c) (((B + C) / 2 \* A + 10) \* 3 \* B) – 6 |
| (B \* A) – (((B ^ 2) / 4) \* C)  20 – ((25 / 4) \* C)  20 - (6,25 \* C)  20 - 6,25  13,75 | ((A\*B)) / (3 ^ 2)  20 / (3 ^ 2)  20 / 9  2,22222222 | (((B + C) / 2 \* A + 10) \* 3 \* B) – 6  ((6 / 2 \* A + 10) \* 3 \* B) – 6  (22 \* 3 \* B) – 6  330 – 6  324 |



Ejercicio 6: Para x=3, y=4; z=1, evaluar el resultado de

R1 = y + z

R1= 4+1

R1= 5

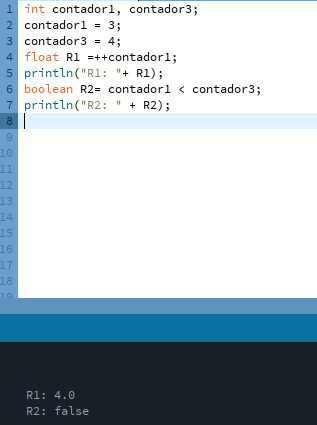
R2 = x >= R1

R2= 3 >= 5

Falso

Ejercicio 7: Para contador1=3, contador3=4, evaluar el resultado de

R1 = ++contador1

R1 = (1 + contador1) //contador1 se le suma 1 y ahora vale 4 y R1 vale contador1 que es 4

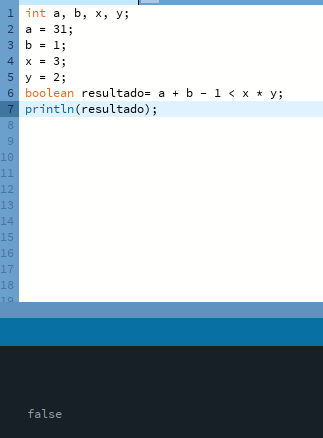
R1 = 4

R2 = contador1 < contador3

R2 = 4 < 4

R2 = Falso

Ejercicio 8: Para a=31, b=-1; x=3, y=2, evaluar el resultado de

a+b-1 < x\*y

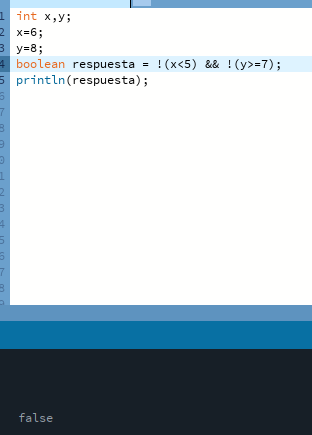
31 + 1 – 1 < 3 \* 2

31 + 1 – 1 < 6

31 < 6

Falso

Ejercicio 9: Para x=6, y=8, evaluar el resultado de

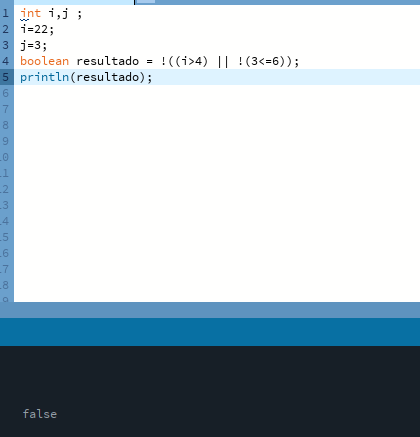
!(x<5)CC !(y>=7)

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

!(falso) CC !(verdadero)

Verdadero CC falso

Falso

Ejercicio 10: Para i=22,j=3, evaluar el resultado de

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

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

!(verdadero || !verdadero)

!(verdadero || falso)

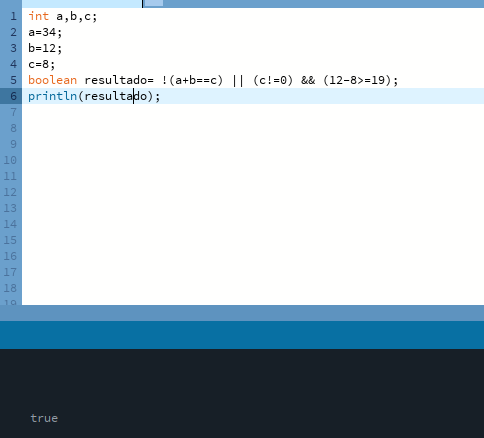
!verdadero

falso

Ejercicio 11: Para a=34, b=12,c=8, evaluar el resultado de

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

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

!falso || verdadero CC falso

Verdadero || verdadero CC falso

Verdadero || falso

verdadero

Sección Análisis – Diseño y Codificación de algoritmos – Aplicación de estructuras de

control

Para cada ejercicio, en el archivo Word agregar las secciones de análisis y diseño, mientras

que, para la codificación, crear el archivo de Processing.

Ejercicio 12: Un problema sencillo. Deberá pedir por teclado al usuario un nombre y

posteriormente realizará la presentación en pantalla de un saludo con el nombre indicado.

**Análisis:**

**Datos de Entrada:**

Nombre del usuario

**Datos de Salida:**

saludo por pantalla

**Proceso: ¿Quién lo realiza?**

El algoritmo

**¿Qué proceso realiza?**

 El usuario ingresa su nombre, se lee el nombre y se devuelve el nombre junto con un saludo que se muestra en la pantalla

|  |
| --- |
| Entidad que resuelve el problrma: algoritmo |
| Variables:  Nusuario: String //el nombre del usuario  Saludo1: string //la primera parte del saludo que se realizara  Saludo2: string// segunda parte del saludo  Saludo3: string //donde se guardaran los saludos y el nombre del usuario |
| Nombre del algoritmo: saludo\_al\_usuario |
| 1 inicio  2 saludo1🡨 Hola  3 saludo2🡨 siéntete bienvenido  4 *leer* Nusuario  5 saludo3🡨 saludo1 + Nusuario + saludo2  6 *mostrar* saludo3 |

Ejercicio 13: Será común resolver problemas utilizando variables. Calcule el perímetro y

área de un rectángulo dada su base y su altura.

**Análisis:**

**Datos de Entrada:**

Base y altura

**Datos de Salida:**

Perímetro y area

**Proceso: ¿Quién lo realiza?**

El usuario

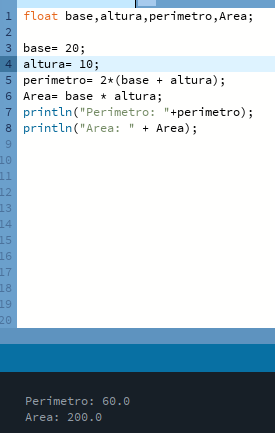
**¿Qué proceso realiza?**

Genera el perímetro multiplicando por 2 la suma de l base y la altura

Y luego calcula el area multiplicando base por altura

**Diseño:**

|  |
| --- |
| **Entidad que resuelve el problema: persona** |
| **Variables:**   * **base: float //** almacena un valor decimal * **altura: float** // almacena un valor decimal * **perimetro: float //** * **area: float //** |
| **Nombre del Algoritmo:** **Area\_y\_perimetro\_de\_un\_rectangulo** |
| **Proceso del algoritmo:**   1. *inicio* 2. *Leer base* 3. *Leer area* 4. *perimetro ← 2\*(base + altura)* 5. *area ← base \* altura* 6. *Mostrar perímetro* 7. *Mostrar area* 8. *fin* |



Ejercicio 14: Una ayuda importante al momento de resolver problemas con algoritmos es

asumir que su gran amigo son las matemáticas. Obtenga la hipotenusa de un triángulo

rectángulo conociendo sus catetos

**Análisis:**

**Datos de Entrada:**

cateto1 y cateto2

**Datos de Salida:**

hipotenusa

**Proceso: ¿Quién lo realiza?**

El usuario

**¿Qué proceso realiza?**

Aplica el teorma de pitagoras elevando los dos catetos al cuadrado luego los suma y el al resultado se le aplica la raíz cuadrada

**Diseño:**

|  |
| --- |
| **Entidad que resuelve el problema: algoritmo** |
| **Variables:**   * **cateto1: float** * **cateto2: float** * **hipotenusa: float //el resultado final** * **H2: float //donde se guardara el resultado de la operacion** |
| **Nombre del Algoritmo:** **Area\_y\_perimetro\_de\_un\_rectangulo** |
| **Proceso del algoritmo:**   1. *inicio* 2. *Leer cateto1* 3. *Leer cateto2* 4. *H2🡨 cateto1 ^ 2 + cateto2 ^ 2* 5. *Hipotenusa🡨 h2^(1/2)* 6. *Mostrar hipotenusa* 7. *fin* |

