UNIVERSIDAD DE GUADALAJARA

CUCEI

DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN

DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

TAREA No. 3

TEMA: EXPRESIÓN ARITMÉTICA, OPERADORES ARITMÉTICOS, PRIORIDAD DE LOS OPERADORES, TIPOS DE DATOS Y VARIABLE

ARELLANO GRANADOS ANGEL MARIANO

FUNDAMENTOS FILOSOFICOS DE LA COMPUTACION

D13 2021-A

LUIS FELIPE MARISCAL LUGO

**MARCO TEÓRICO:**

Expresión Aritmética:

Las expresiones aritméticas le permiten realizar operaciones matemáticas en atributos de recurso que sean parte de la norma. Si selecciona esta opción, podrá seleccionar varios atributos de recurso, valores y operadores (suma, resta, multiplicación o división) para utilizarlos entre sí.

# Tabla De Los Operadores Aritméticos:

|  |  |  |
| --- | --- | --- |
| Symbol | Operation | Description |
| + | Addition | Adds two numbers |
| - | Subtraction | Subtracts one number from another |
| \* | Multiplication | Multiplies one number by another |
| / | Division | Divides one number by another and gives the result as a floating-point number |
| // | Integer division | Divides one number by another and gives the result as an integer |
| % | Remainder | Divides one number by another and gives the remainder |
| \*\* | Exponent | Raises a number to a power |

# Prioridad De Los Operadores:

First, operations that are enclosed in parentheses are performed first. Then, when two operators share an operand, the operator with the higher precedence is applied first. The precedence of the math operators, from highest to lowest, are:

1. Exponentiation: \*\*

2. Multiplication, division, and remainder: \* / // %

3. Addition and subtraction: + -

# Tipos De Datos:

A number that is written into a program’s code is called a ***numeric literal.***

* A numeric literal that is written as a whole number with no decimal point is considered an ***int***. Examples are 7, 124, and -9.
* A numeric literal that is written with a decimal point is considered a ***float***. Examples are 1.5, 3.14159, and 5.0.

Python also has a data type named ***str***, which is used for storing strings in memory.

# Variable:

A variable is a name that represents a value stored in the computer’s memory.

Programs use variables to access and manipulate data that is stored in memory. A variable is a name that represents a value in the computer’s memory.

When a variable represents a value in the computer’s memory, we say that the variable references the value.

# REFERENCIAS:

* Gaddis, T. (2012). starting out with python. Google. <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwi_rITUy6bvAhUDWqwKHXywCo0QFjAAegQIARAD&url=http%3A%2F%2Findex-of.es%2FPython%2FStarting%2520Out%2520With%2520%2520Python%2520Second%2520Edition.pdf&usg=AOvVaw3s1kmo1BGL5EkeL2ELSN9N>
* IBM. (2003, August 27). Expresiones aritméticas. <https://publib.boulder.ibm.com/wcmid/mp/v50/helpsystem/es/ct000102.html>

**PROBLEMAS:**

# PROBLEMA 1:

## Definición Del Programa:

Solicitar al usuario dos números enteros, aplicar todos los operadores aritméticos (+,-,\*,/,//,%,\*\*), guardar el resultado en una tercera variable (se utilizará la misma variable para guardar cada uno de los resultados), por último imprimir el resultado de cada operación.

## Análisis Del Programa:

ENTRADA:

Solicitar al usuario dos datos y guardar cada uno en variables diferentes *n1* y *n2*.

PORCESO:

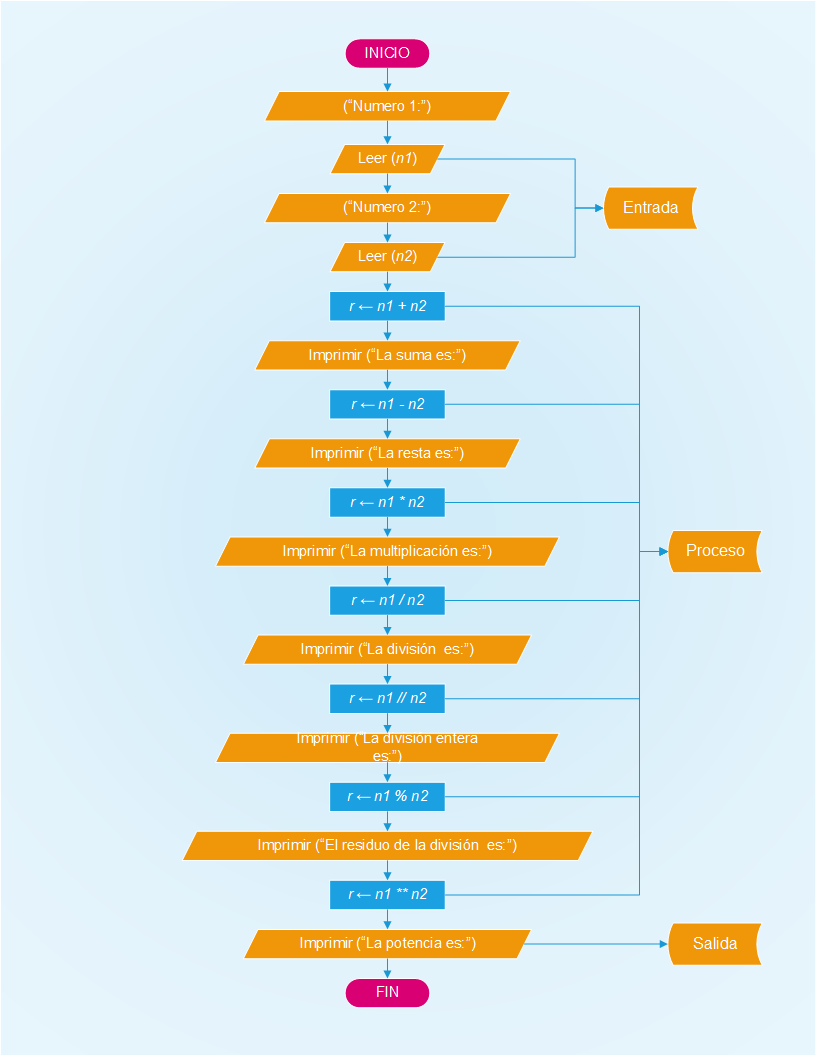
Someter ambas variables a todos los operadores aritméticos, dentro de una misma variable *r*, imprimiendo el resultado de cada operador antes de iniciar el proceso del siguiente operador.

SALIDA:

Mostrar en la pantalla todos los resultados de ambas variables en cada operador especificando de cual es cada uno.

## Diseño Del Algoritmo:

Diagrama De Flujo:



Pseudo Código:

#Arellano Granados Angel Mariano

#Algoritmo para usar todos los operadores aritméticos con dos variables e imprimir todos los resultados

#Entrada:

Entrada (“Numero 1”)

Leer n1

Entrada (“Numero 2”)

Leer n2

#Proceso

r ← n1 + n2

#Salida

Imprimir ( “La suma es:”, r)

#Proceso

r ← n1 - n2

#Salida

Imprimir ( “La resta es:”, r)

#Proceso

r ← n1 \* n2

#Salida

Imprimir ( “La multiplicación es:”, r)

#Proceso

r ← n1 / n2

#Salida

Imprimir ( “La división es:”, r)

#Proceso

r ← n1 // n2

#Salida

Imprimir ( “La división entera es:”, r)

#Proceso

r ← n1 % n2

#Salida

Imprimir ( “El residuo de la división es:”, r)

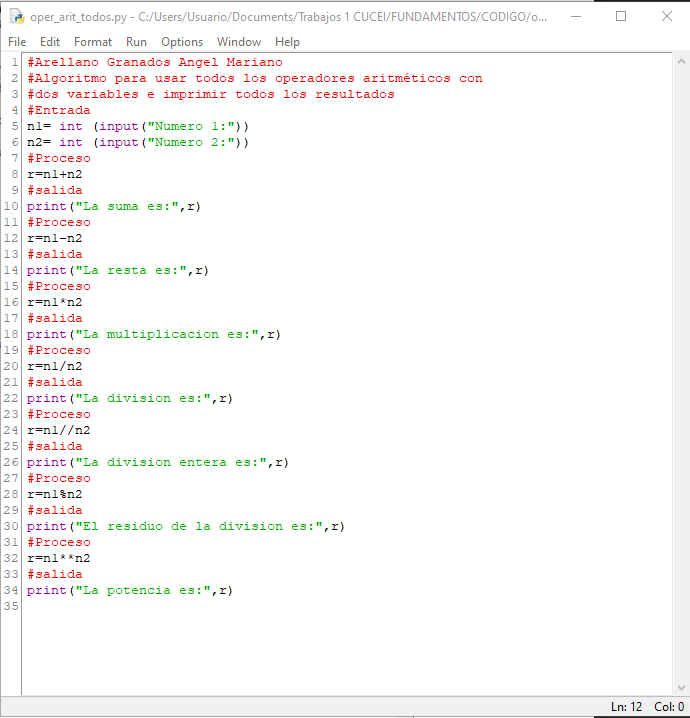
#Proceso

r ← n1 \*\* n2

#Salida

Imprimir ( “La potencia es:”, r)

## Capturas:





# PROBLEMA 2:

## Definición Del Programa:

Solicitar al usuario tres números enteros, calcular el promedio, guardar el resultado en una cuarta variable, por último imprimir el resultado.

## Análisis Del Programa:

ENTRADA:

Solicitar al usuario tres datos que se almacenaran en tres variables diferentes *n1, n2 y n3.*

PORCESO:

Para sacar el promedio de las variables usaremos el operador aritmético SUMA y DIVISIÓN y el operador de asignación (←) en dos procesos, de tal manera que:

*r* ← *n1 + n2 + n3*

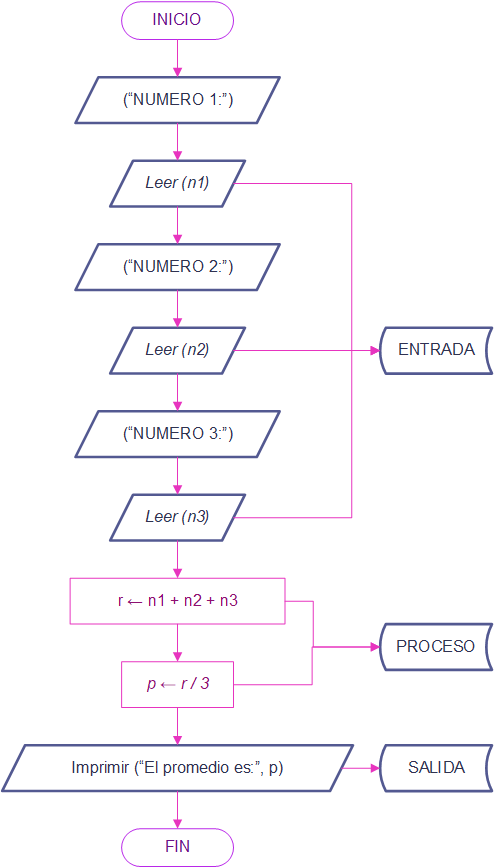
*p* ← *r / 3*

SALIDA:

Mostrar el resultado en pantalla, es decir, mostrar el promedio.

## Diseño Del Algoritmo:

Diagrama De Flujo:



Pseudo Código:

#Arellano Granados Angel Mariano

#Algoritmo para calcular el promedio de tres números

#Entrada:

Entrada (“Numero 1”)

Leer n1

Entrada (“Numero 2”)

Leer n2

Entrada (“Numero 3”)

Leer n3

#Proceso

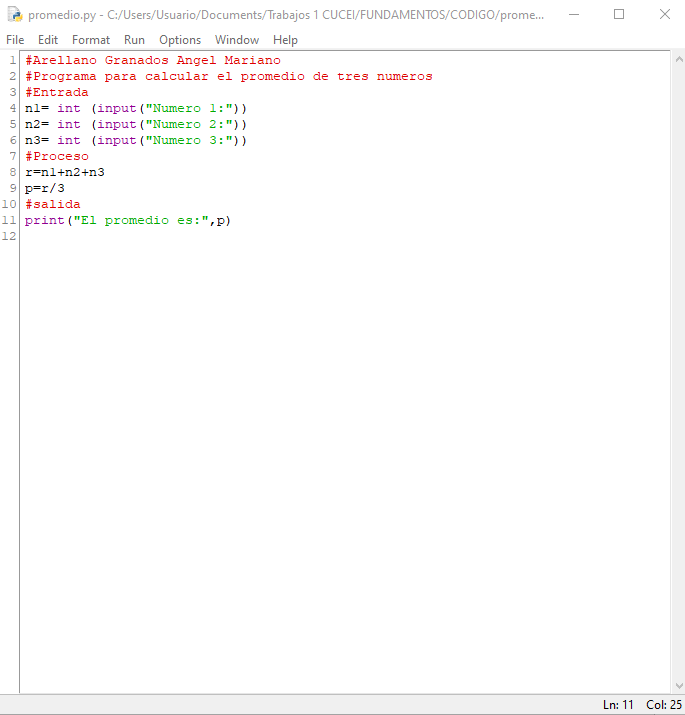
r ← n1 + n2 + n3

p ← r / 3

#Salida

Imprimir ( “El promedio es:”, p)

## Capturas:





**CONCLUCIÓN:**

Tras elaborar esta actividad me di cuenta que el proceso de elaboración de un programa en PYTHON, puede llegar a ser bastante preocupante, pero siguiendo los pasos metódicos se hace más ameno en trabajo y es muy reconfortarle ver como los programas funcionan a la primera.