UNIVERSIDAD DE GUADALAJARA

CUCEI

DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN

DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

PRACTICA No. 6

TEMA: CICLOS ANIDADOS

ARELLANO GRANADOS ANGEL MARIANO

FUNDAMENTOS FILOSOFICOS DE LA COMPUTACION

D13 2021-A

LUIS FELIPE MARISCAL LUGO

**MARCO TEÓRICO:**

**Nested Loops**

A loop that is inside another loop is called a nested loop.

A nested loop is a loop that is inside another loop. A clock is a good example of something that works like a nested loop. The second hand, minute hand, and hour hand all spin around the face of the clock. The hour hand, however, only makes 1 revolution for every 12 of the minute hand’s revolutions. And it takes 60 revolutions of the second hand for the minute hand to make 1 revolution. This means that for every complete revolution of the hour hand, the second hand has revolved 720 times. Here is a loop that partially simulates a digital clock. It displays the seconds from 0 to 59:

for seconds in range(60):

print(seconds)

We can add a minutes variable and nest the loop above inside another loop that cycles through 60 minutes:

for minutes in range(60):

for seconds in range(60):

print(minutes, ':', seconds)

To make the simulated clock complete, another variable and loop can be added to count the hours:

for hours in range(24):

for minutes in range(60):

for seconds in range(60):

print(hours, ':', minutes, ':', seconds)

This code’s output would be:

0:0:0

0:0:1

0:0:2

(The program will count through each second of 24 hours.)

23:59:59

The innermost loop will iterate 60 times for each iteration of the middle loop. The middle loop will iterate 60 times for each iteration of the outermost loop. When the outermost loop has iterated 24 times, the middle loop will have iterated 1,440 times and the innermost loop will have iterated 86,400 times! Figure 5-9 shows a flowchart for the complete clock simulation program previously shown.

# 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>

**PROBLEMAS:**

## Definición Del Programa:

## Desarrolla un programa que muestre en pantalla el siguiente patrón.

## Análisis Del Programa:

**ENTRADA:**

Solicitar al usuario un número entero y guardarlo en la variable *lim.*

**ESTRUCTURA DE CONTROL REPETITIVA FOR:**

Usaremos la estructura de control repetitiva for para crear dos ciclos

**SALIDA:**

Imprimir cada numero del ciclo separado por un signo “+” y al final la sumatoria total de todos los números.

## Diseño Del Algoritmo:

**Diagrama De Flujo:**

**Microsoft Visio**

**Pseudo Código:**

#Arellano Granados Angel Mariano

#Algoritmo que imprime un patrón de triangulo

## BASE\_SIZE = 8

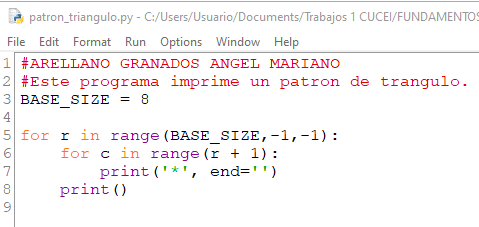
## desde r en rango (BASE\_SIZE,-1,-1):

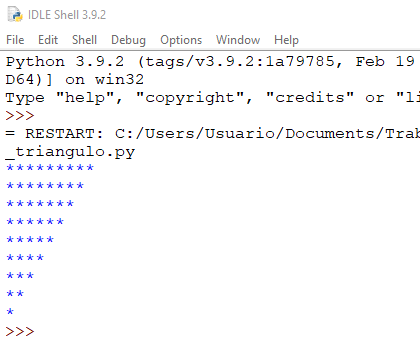
## desde c en rango (r + 1):

## imprimir ('\*')

## imprimir ()

## Capturas:





**CONCLUCIÓN:**