**UNIVERSIDAD DE GUADALAJARA**

**CUCEI**

**DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN**

**DEPARTAMENTO DE CIENCIAS COMPUTACIONALES**

**PRÁCTIA No. 3**

*TEMA***: ESTRUCTURA DE CONTROL SELECTIVA DOBLE O COMPUESTA ANIDADA**

***EQUIPO No. 4***

*INTEGRANTES*:

**ROBLES PULIDO EFRAIN**

**RODRÍGUEZ CANO CHRISTIAN FABIAN**

**RODRÍGUEZ RICO MARIO EDUARDO**

*NOMBRE DE LA MATERIA:* **FUNDAMENTOS FILOSOFICOS DE LA COMPUTACION**

*SECCIÓN:* **D13**  *CALENDARIO:* **2021A**

*NOMBRE DEL PROFESOR:* **MARISCAL LUGO LUIS FELIPE**





**Marco Teórico**

|  |  |
| --- | --- |
| **Tabla 4-3 Operadores Lógicos** | |
| **Operador** | **Significado** |
| and | El operador and conecta dos expresiones booleanas en una expresión compuesta. Ambas subexpresiones deben ser verdaderas para que la expresión compuesta sea verdadera. |
| or | El operador or conecta dos expresiones booleanas en una expresión compuesta. Una o ambas subexpresiones deben ser verdaderas para que la expresión compuesta sea verdadera. Sólo es necesario que una de las subexpresiones sea verdadera, y no importa cuál. |
| not | El operador not es un operador unario, lo que significa que funciona con un solo operando. El operando debe ser una expresión booleana. El operador not invierte la verdad de su operando. Si se aplica a una expresión que es verdadera, el operador devuelve falso. Si se aplica a una expresión que es falsa, el operador devuelve verdadero. |

|  |  |
| --- | --- |
| **Tabla 4-4 Expresiones booleanas compuestas mediante operadores lógicos** | |
| **Expresión** | **Significado** |
| (x > y) and (a < b) | ¿Es "x" mayor que "y" **Y** es "a" menor que "b"? |
| (x == y) or (x == z) | ¿Es "x" igual a "y" **O** es "x" igual a "z"? |
| not (x > y) | ¿La expresión “(x > y)” **NO** es verdadera? |

|  |  |
| --- | --- |
| **Tabla 4-5 Tabla de verdad para el operador AND** | |
| **Expresión** | **Valor de la Expresión** |
| verdadero and falso | falso |
| falso and verdadero | falso |
| falso and falso | falso |
| verdadero and verdadero | verdadero |

|  |  |
| --- | --- |
| **Tabla 4-6 Tabla de verdad para el operador OR** | |
| **Expresión** | **Valor de la Expresión** |
| verdadero or falso | verdadero |
| falso or verdadero | verdadero |
| falso or falso | falso |
| verdadero or verdadero | verdadero |

|  |  |
| --- | --- |
| **Tabla 4-7 Tabla de verdad para el operador NOT** | |
| **Expresión** | **Valor de la Expresión** |
| not verdadero | falso |
| not falso | verdadero |

# Bibliografía

Gaddis, T. (2021). *Starting out with Python* (tercera ed.). Pearson. Recuperado el 13 de Marzo de 2021, de https://fdocuments.in/document/starting-out-with-python-3rd-edition-pdf-firebase-python-programming-2nd-edition.html

**Resolver el problema**

**1.-Definición del problema**

Una importante compañía tiene uno de sus mejores productos a la venta a tan sólo $99, pero para respetar este precio debes comprar mínimo 10 productos y máximo 99 y además se está manejando un descuento especial de acuerdo con la cantidad de productos comprados, revise la siguiente tabla:

|  |  |
| --- | --- |
| **CANTIDAD** | **DESCUENTO** |
| 10 – 19 | 20% |
| 20 – 49 | 30% |
| 50 - 99 | 40% |

**2.- Análisis del problema**

**Entrada:** Solicitar al usuario un dato de tipo entero, se guardara en la variable “p”.

**Proceso:** Calcular cuánto deberá pagar mediante una multiplicación (d p\*99) y después determinar que descuento, utilizando 3 condiciones, el operador lógico “y” y los operadores relacionales para comparar la cantidad de productos a comprar (p>=10 and p<=19), dentro de cada condición se deberá calcular el precio total con su determinado descuento (pd d\*(1-0.2)).

**Salida:** Mostrar el mensaje correspondiente de las condiciones, es decir, mostrar que descuento se utilizó y el costo total con descuento.

**3.-Diseño del algoritmo**

**Diagrama de flujo**

Plataforma: Lucidchart

**Diagrama

Descripción generada automáticamente**

**Pseudocódigo**

# Robles Pulido Efrain

#Rodríguez Cano Christian Fabian.

#Rodríguez Rico Mario Eduardo.

#seccion D13

#calendario 2021A

#Algoritmo para determinar el descuento utilizado y el precio a pagar

#Entrada

Entrada (“Teclea la cantidad de productos comprados:”)

Leer (p)

#Proceso

d p\*99 #Operacion para calcular el precio total a pagar

#Estructura de control selectiva doble anidada y el operador lógico “y”

Si p>=10 y p<=19:

pd d\*(1-0.2) #Operacion para calcular el precio total a pagar con su descuento

Imprimir (“Tu descuento es de 20% y pagaras $”, pd) #Salida

Sino:

Si p>=20 y p<=49:

pd d\*(1-0.3) #Operacion para calcular el precio total a pagar con su descuento

Imprimir (“Tu descuento es de 30% y pagaras $”, pd) #Salida

Sino:

Si p>=50 y p<=99:

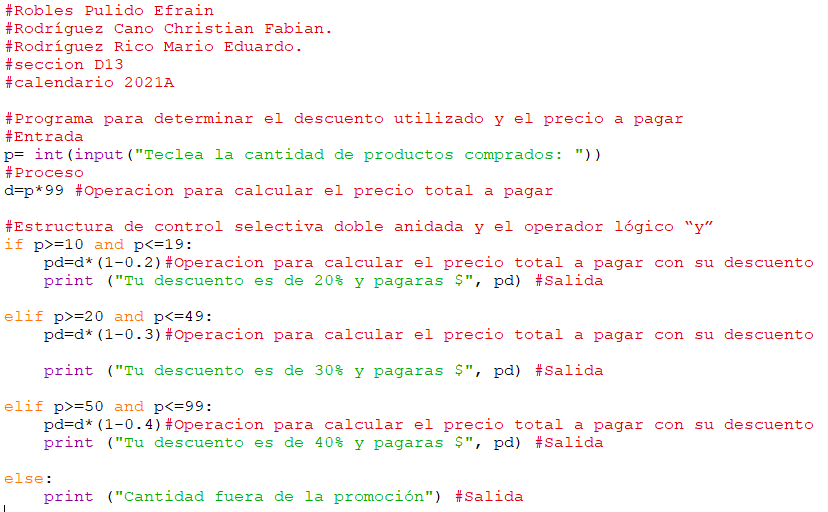
pd d\*(1-0.4) #Operacion para calcular el precio total a pagar con su descuento

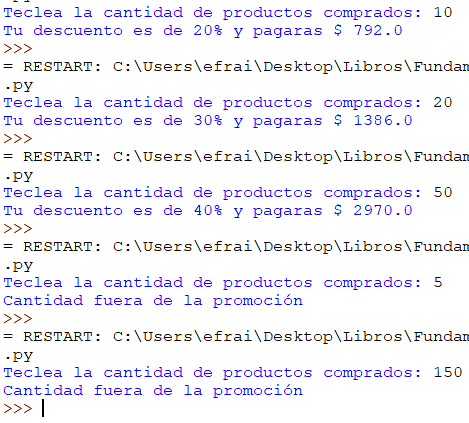
Imprimir (“Tu descuento es de 40% y pagaras $”, pd) #Salida

Sino:

Imprimir (“Cantidad fuera de la promoción”) #Salida

**4.-Codificación**





**5.- Conclusiones personales**

**Robles Pulido Efraín*:***

En esta práctica aprendí como se debe de aplicar la estructura de control selectiva doble anidada para comparar un valor es necesario utilizar los operadores lógicos y los operadores relacionales en las condiciones en donde tomara una determinada salida deseada que programemos. También le tuvimos que agregar operaciones para determinar el valor total de los productos y otra para tener el precio final con el descuento determinado para después mostrarlo como mensaje. La práctica me permitió aplicar los conocimientos que vi en clase en una situación sencilla de la vida.

**Rodríguez Cano Christian Fabian:**

Para concluir con esta práctica como tuvimos presente está vez se trató de un problema para el descuento que se le hará a una compra dependiendo del número de productos y la oferta y pues utilizamos la estructura de control selectiva doble anidada y también el operador lógico y, la verdad fue algo confuso para mí tratar de plasmar en código el desarrollo del problema pero al final encontré una solución igual no pareció ser tan difícil pero me dejó con experiencia para la próxima vez que nos topemos con un problema similar, también utilizamos en parte conocimientos de clases pasadas para resolver el problema lo cual tenemos presente tomar nota porque estamos usando cosas ya aprendidas previamente.

**Rodríguez Rico Mario Eduardo:**

En la práctica se utilizó la estructura de control selectiva doble anidada para poder resolver el problema a tratar, que sería un descuento a un precio dependiendo del número de productos que se compraban. Se añadieron distintas operaciones para poder encontrar y determinar el valor total de los productos y sí entraban dentro de la promoción. Se utilizaron condicionales enseñadas en clase como el “elif” y “if” para poder resolver el problema.

En general, la práctica me gustó ya que nos puso a prueba de los conocimientos adquiridos con la estructura de control selectiva que se enseñó que clase.