

Programa 1: 2. Declara un método llamado `descuento_clientes()`, utilizando los valores de los elementos del objeto (`self`).

Instrucciones:

Si el monto es menor que \$500 ----> no hay descuento.

Si el monto está comprendido entre \$500 y \$1000

inclusive -----> 5% de descuento.

Si el monto esta comprendido entre\$ 1000 y \$ 7000

inclusive -----> 11% de descuento.

Si el monto este comprendido entre \$ 7000 y \$ 15000

inclusive -----> 18% de descuento.

Si el monto es mayor a \$15000

Inclusive -----> 25%

Construya un programa tal, que, dado el monto de la compra de un cliente, determine lo que el mismo debe pagar.

Código del programa:

```
class Tres_programas():
    def __init__(self, cantidad):
        self.cantidad = cantidad

    def descuento_clientes(self):
        descuento = 0
        if self.cantidad < 500:
            descuento = 0
        elif 500 <= self.cantidad <= 1000:
            descuento = self.cantidad * 0.05
        elif 1000 <= self.cantidad <= 7000:
            descuento = self.cantidad * 0.11
        elif 7000 <= self.cantidad <= 15000:
            descuento = self.cantidad * 0.18
```

else:

 descuento = self.cantidad * 0.25

return descuento

compra = float(input("Ingresa la cantidad a pagar: "))

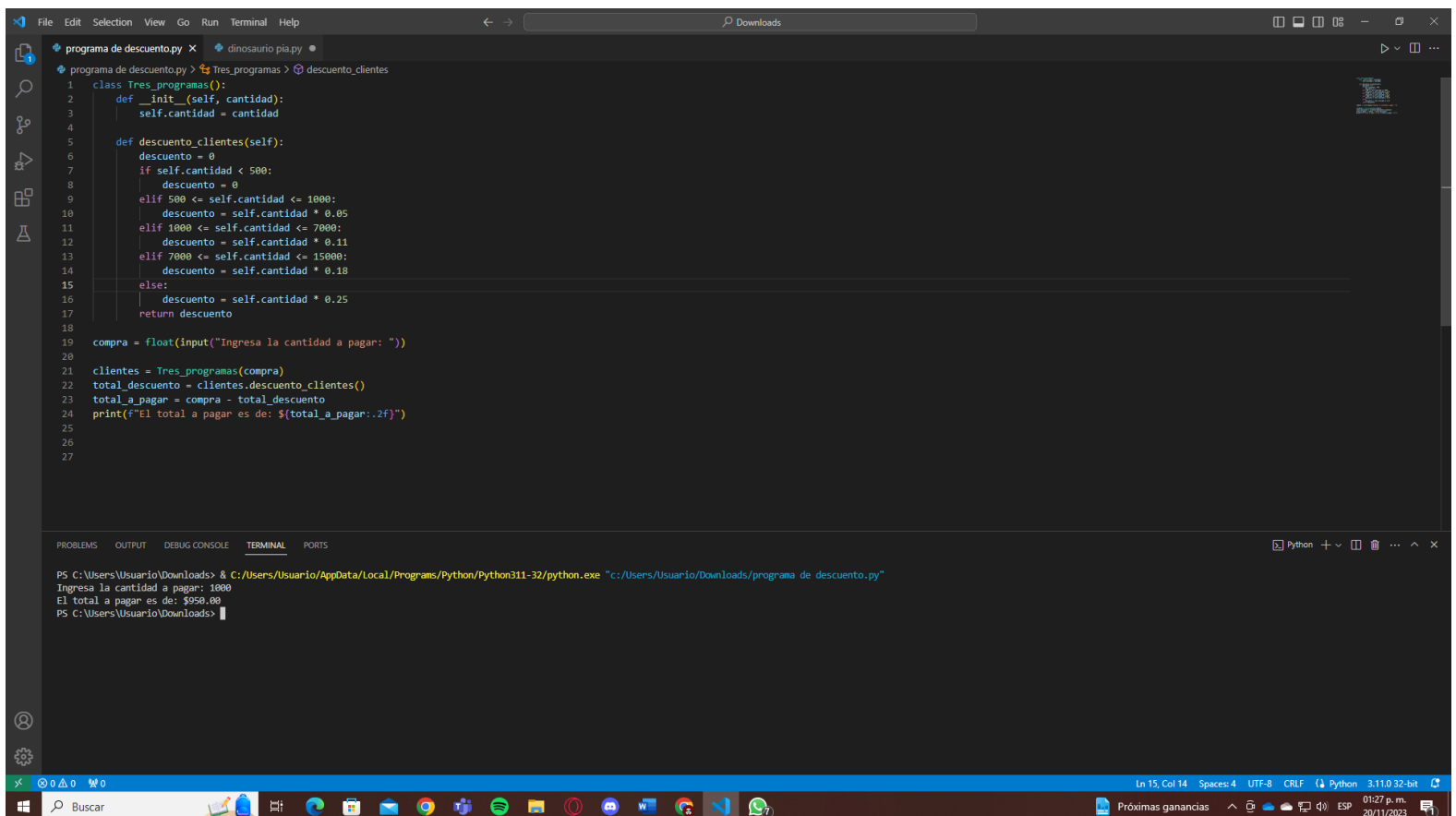
clientes = Tres_programas(compra)

total_descuento = clientes.descuento_clientes()

total_a_pagar = compra - total_descuento

print(f"El total a pagar es de: \${total_a_pagar:.2f}")

Ejecución:



The screenshot shows a Visual Studio Code editor window with a Python file named 'programa de descuento.py'. The code defines a class 'Tres_programas' with an '__init__' method and a 'descuento_clientes' method. The 'descuento_clientes' method calculates a discount based on the quantity: 0% for quantities up to 500, 5% for 500-1000, 11% for 1000-7000, 18% for 7000-15000, and 25% for quantities above 15000. The main code prompts the user for a purchase amount, creates an instance of the class, calculates the discount, and prints the total amount to be paid.

```
1 class Tres_programas():
2     def __init__(self, cantidad):
3         self.cantidad = cantidad
4
5     def descuento_clientes(self):
6         descuento = 0
7         if self.cantidad < 500:
8             descuento = 0
9         elif 500 <= self.cantidad <= 1000:
10            descuento = self.cantidad * 0.05
11        elif 1000 <= self.cantidad <= 7000:
12            descuento = self.cantidad * 0.11
13        elif 7000 <= self.cantidad <= 15000:
14            descuento = self.cantidad * 0.18
15        else:
16            descuento = self.cantidad * 0.25
17        return descuento
18
19 compra = float(input("Ingresa la cantidad a pagar: "))
20
21 clientes = Tres_programas(compra)
22 total_descuento = clientes.descuento_clientes()
23 total_a_pagar = compra - total_descuento
24 print(f"El total a pagar es de: ${total_a_pagar:.2f}")
25
26
27
```

The terminal output shows the execution of the script. The user enters '1000' as the purchase amount, and the program outputs 'El total a pagar es de: \$950.00'.

```
PS C:\Users\Usuario\Downloads> & c:\Users\Usuario\AppData\Local\Programs\Python\Python311-32\python.exe "c:\Users\Usuario\Downloads\programa de descuento.py"
Ingresa la cantidad a pagar: 1000
El total a pagar es de: $950.00
PS C:\Users\Usuario\Downloads>
```

Programa 2: Declara un método llamado dinosaurio (), utilizando los valores de los elementos del objeto (self).

Instrucciones: Dado como datos el nombre de un dinosaurio, su peso y su longitud, expresados estos dos últimos en toneladas y pies respectivamente; escriba el nombre del dinosaurio, su peso expresado en kilogramos y su longitud expresada en metros.

Código del programa:

```
class Dinosaurio:

    def __init__(self, nombre, peso_toneladas, long_pies):

        self.nombre = nombre

        self.peso_toneladas = peso_toneladas

        self.longitud_pies = long_pies

    def peso_kg(self):

        return self.peso_toneladas * 1000

    def longitud_metros(self):

        return self.longitud_pies * 0.3048

nombre_dinosaurio = input("Ingresa el nombre del dinosaurio: ")
peso_dinosaurio_ton = float(input("Ingresa el peso del dinosaurio en toneladas: "))
longitud_dinosaurio_pies = float(input("Ingresa la longitud del dinosaurio en pies: "))

dino = Dinosaurio(nombre_dinosaurio, peso_dinosaurio_ton, longitud_dinosaurio_pies)

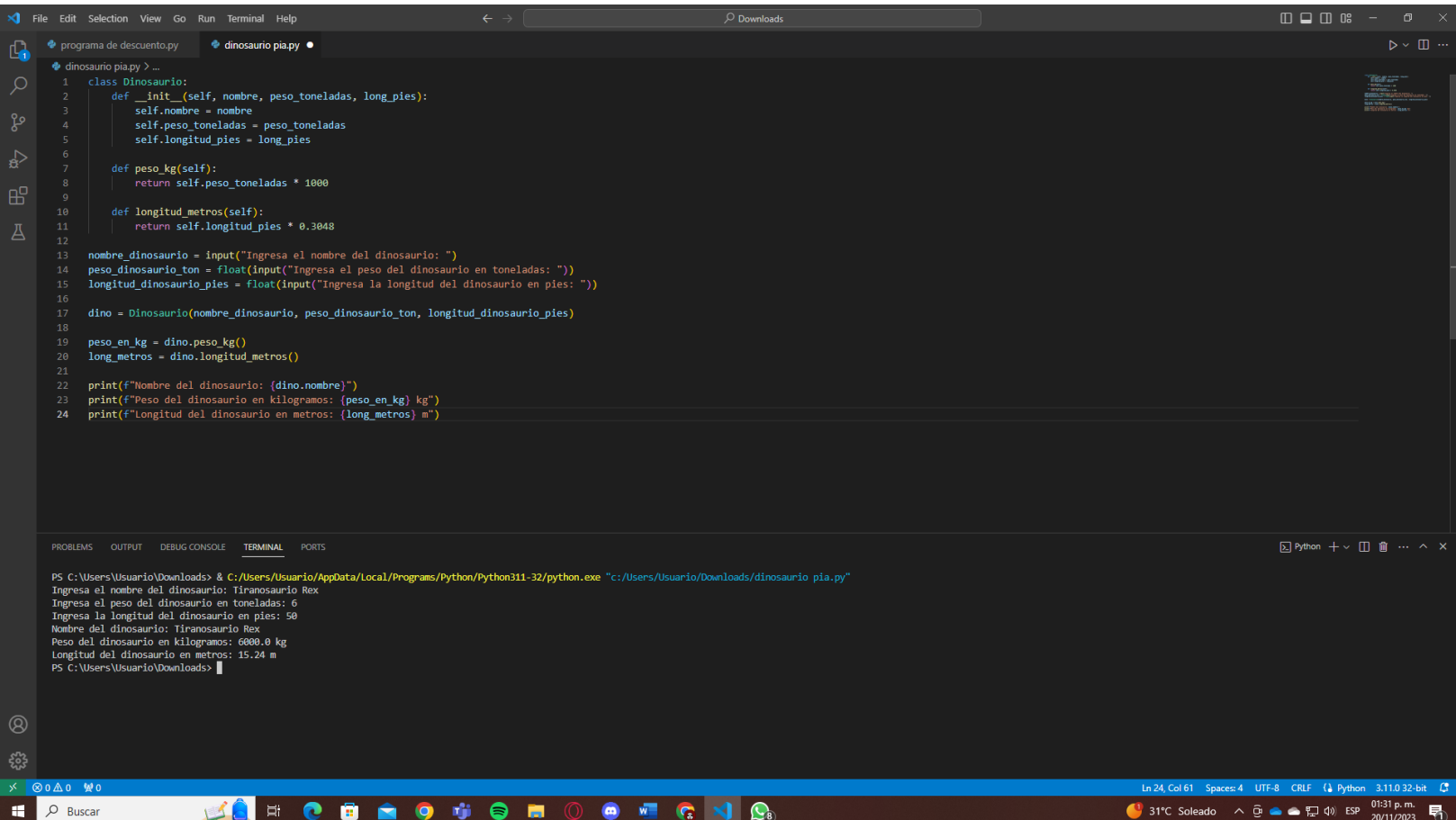
peso_en_kg = dino.peso_kg()
long_metros = dino.longitud_metros()
```

```
print(f"Nombre del dinosaurio: {dino.nombre}")
```

```
print(f"Peso del dinosaurio en kilogramos: {peso_en_kg} kg")
```

```
print(f"Longitud del dinosaurio en metros: {long_metros} m")
```

Ejecución:



The screenshot shows a Visual Studio Code editor with a Python file named `dinosaurio pia.py`. The code defines a `Dinosaurio` class with methods to calculate weight in kg and length in meters based on user input. The terminal output shows the program running with inputs: `Tiranosaurio Rex`, `6`, and `50`, resulting in `6000.0 kg` and `15.24 m`.

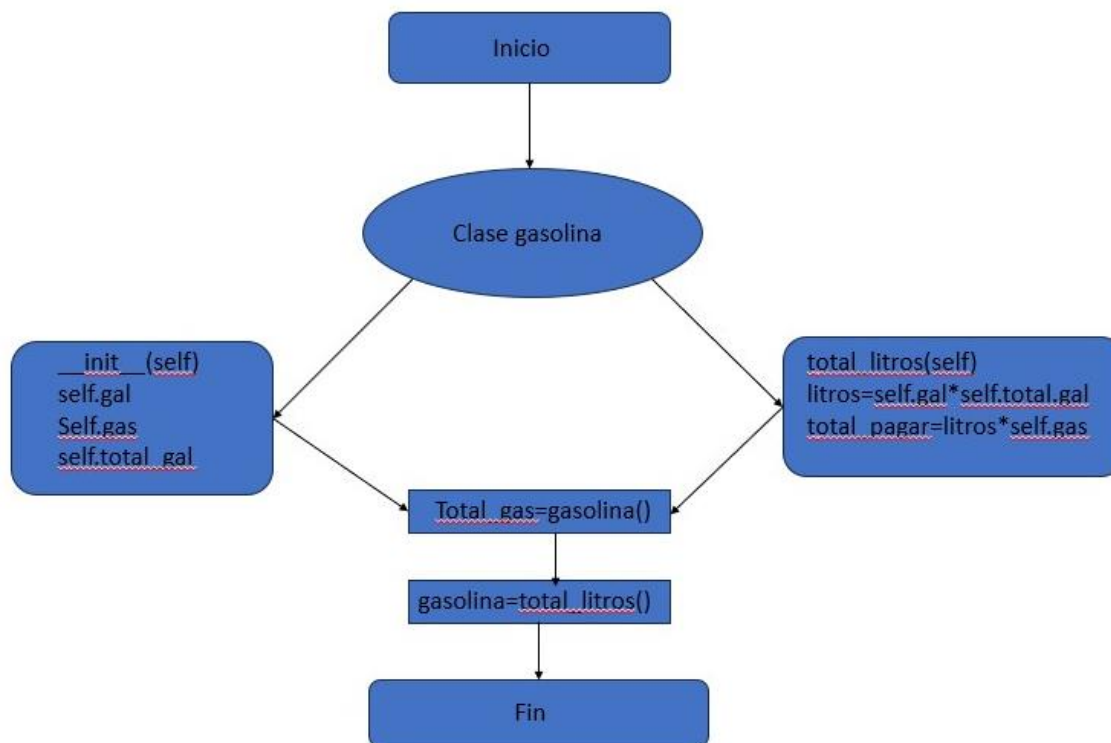
```
1 class Dinosaurio:
2     def __init__(self, nombre, peso_toneladas, long_pies):
3         self.nombre = nombre
4         self.peso_toneladas = peso_toneladas
5         self.longitud_pies = long_pies
6
7     def peso_kg(self):
8         return self.peso_toneladas * 1000
9
10    def longitud_metros(self):
11        return self.longitud_pies * 0.3048
12
13    nombre_dinosaurio = input("Ingresa el nombre del dinosaurio: ")
14    peso_dinosaurio_ton = float(input("Ingresa el peso del dinosaurio en toneladas: "))
15    longitud_dinosaurio_pies = float(input("Ingresa la longitud del dinosaurio en pies: "))
16
17    dino = Dinosaurio(nombre_dinosaurio, peso_dinosaurio_ton, longitud_dinosaurio_pies)
18
19    peso_en_kg = dino.peso_kg()
20    long_metros = dino.longitud_metros()
21
22    print(f"Nombre del dinosaurio: {dino.nombre}")
23    print(f"Peso del dinosaurio en kilogramos: {peso_en_kg} kg")
24    print(f"Longitud del dinosaurio en metros: {long_metros} m")
```

Terminal Output:

```
PS C:\Users\Usuario\Downloads> & C:\Users\Usuario\AppData\Local\Programs\Python\Python311-32\python.exe "c:/Users/Usuario/Downloads/dinosaurio pia.py"
Ingresa el nombre del dinosaurio: Tiranosaurio Rex
Ingresa el peso del dinosaurio en toneladas: 6
Ingresa la longitud del dinosaurio en pies: 50
Nombre del dinosaurio: Tiranosaurio Rex
Peso del dinosaurio en kilogramos: 6000.0 kg
Longitud del dinosaurio en metros: 15.24 m
PS C:\Users\Usuario\Downloads>
```

Programa 3: Declara un método llamado gasolina(), utilizando los valores de los elementos del objeto (self).

Instrucciones: Construya un diagrama de flujo que resuelva el problema que tienen en una gasolinera. Los surtidores de esta registran lo que "surten" en galones, pero el precio de la gasolina está fijado en litros. El diagrama de flujo debe calcular e imprimir lo que hay que cobrarle al cliente.



Programa4: Declara un método llamado archivos_texto(): que me permita la creación, escritura y lectura de los métodos anteriores utilizando los valores de los elementos del objeto (self).

Código

```
class archivos_texto():

    def __init__(self):
        self.gal=float(input("ingrese galones: "))
        self.gas=float(8.20)
        self.total_gal=float(3.785)

    def total_litros(self):
        print("litros",self.gal*self.total_gal)
        print("total a pagar",(self.gal*self.total_gal)*self.gas)

gasolina=archivos_texto()
gasolina.total_litros()
```

Ejecución

