



## EJERCICIO DE LA MOCHILA

### **Presenta**

Cristian David Mora Sáenz

### **Docente**

Segundo Fidel Puerto Garavito

### **Asignatura**

Diseño de Algoritmos

**NRC:** 7487

Corporación Universitaria Minuto de Dios

Facultad de Ingeniería

Colombia, Bogotá

abril 28 de 2020.

El ejercicio de la mochila consiste en un excursionista que debe preparar su mochila, la cual tiene una capacidad limitada y por tanto no le permite llevar todos los artículos que quisiera tener en la excursión. Cada artículo que el excursionista puede incluir en la mochila le reporta una determinada utilidad.

Para desarrollar este ejercicio definimos dos clases, con unas funciones importantes para realizar lo que describe el texto anterior.

1. Definimos la clase Mochila, esta cuenta con dos funciones:
  - *Init*: Es una función para inicializar el peso y la capacidad que tiene la mochila. Recibe como parámetros el método *self*, *peso* y *capacidad*.
  - *MeterObjeto*: En esta función utilizamos un acumulador para almacenar la cantidad de objetos que entran en la mochila, tiene como parámetro el *objeto*.

```
class Mochila:
    def __init__(self, peso, capacidad):
        self.peso = peso
        self.capacidad = capacidad

    def meterObjeto(self, objeto):
        self.peso += objeto.peso

class Objeto:
    def __init__(self, peso, valor):
        self.peso = peso
        self.valor = valor
        self.valorporunidad = self.valor/self.peso
```

2. Utilizando el algoritmo de ordenamiento por selección se organizan los elementos de adentro de la mochila

```
#algoritmo de ordenación por selección
def selectionsort(lista,tam):
    for i in range(0,tam-1):
        min=i
        for j in range(i+1,tam):
            if lista[min].valorporunidad > lista[j].valorporunidad:
                min=j
        aux=lista[min]
        lista[min]=lista[i]
        lista[i]=aux
```

3. Luego se utiliza las funciones de Mochila y Objeto para colocar la capacidad de la mochila, y los objetos, para que este algoritmo ordene los elementos.

```
mochila = Mochila(0, 100)

objetos = []
objetos.append(Objeto(20, 3))
objetos.append(Objeto(50, 5))
objetos.append(Objeto(20, 8))
objetos.append(Objeto(10, 4))
objetos.append(Objeto(50, 1))
objetos.append(Objeto(60, 3))
objetos.append(Objeto(30, 5))
objetos.append(Objeto(70, 8))
objetos.append(Objeto(80, 1))
objetos.append(Objeto(15, 7))
```

4. Finalmente, utilizando los ciclos WHILE y las comparaciones de IF se realiza el ordenamiento dependiendo de la importancia de los elementos.

```
resultado = []
for i in range(len(objetos)):
    resultado.append(0)

i = len(objetos) - 1 #empezamos desde el final porque están ordenados de menor a mayor
while(mochila.peso < mochila.capacidad):
    objeto = objetos[i]
    if ((mochila.peso + objeto.peso) <= mochila.capacidad):
        resultado[i] = 1
        mochila.meterObjeto(objeto)
    else:
        resultado[i] = (mochila.capacidad - mochila.peso) / objeto.peso
        mochila.peso = mochila.capacidad
    i-=1

print ("Resultado: Fracciones de los objetos en la mochila: ")
for i in range(len(resultado)-1, -1, -1):
    print ("Objeto ", i, ": ", resultado[i])
```

Por último, obtenemos como resultado: donde 1 es el mayor valor de importancia según el objeto y 0 es un objeto sin importancia.

```
Resultado: Fracciones de los objetos en la mochila:
Objeto 9 : 1
Objeto 8 : 1
Objeto 7 : 1
Objeto 6 : 1
Objeto 5 : 1
Objeto 4 : 0.07142857142857142
Objeto 3 : 0
Objeto 2 : 0
Objeto 1 : 0
Objeto 0 : 0
PS C:\Users\CristianM\Desktop\Ejercicios de Diseño>
```

Enlace a GitHub: <https://github.com/CristianMoraS/Mochila-EX>