



Algoritmos y Estructuras de Datos

Participantes:

- Gabriel Antonio Rojas Uriarte

Docente:

Cesar David Marín López

Managua 28 de abril de 2025.

```
pila.py > ...
1 class Pila:
2     def __init__(self):
3         self.lista = [] #creamos la lista
4
5     def esta_vacia(self):
6         return self.lista == [] #verificamos si la lista esta vacia
7
8     def agregar(self, elemento):
9         self.lista.append(elemento) #agregamos el elemento
10
11    def eliminar(self):
12        return self.lista.pop() #eliminamos el ultimo elemento
13
14    def imprimir(self):
15        return self.lista #imprimimos la lista
16
17    def tamaño(self):
18        return len(self.lista) #devolvemos el tamaño de la lista
19
20    p = Pila()#creamos la instancia y mandamos a llamar al constructor
21    print("Estado de la pila: ")
22    print(p.esta_vacia())
23    elemento = int(input("Ingrese un elemento de tipo entero: "))
24    p.agregar(elemento)
25    elemento = float(input("Ingrese un elemento de tipo decimal: "))
26    p.agregar(elemento)
27    elemento = input("Ingrese un elemento de tipo cadena: ")
28    p.agregar(elemento)
29    print("Estado de la pila: ")
30    print(p.esta_vacia())
31    print("Lista de elementos:")
32    print(p.imprimir())
33    print("Tamaño de la pila: ")
34    print(p.tamaño())
35    print("Eliminar elemento")
36    print(p.eliminar())
37    print("Estado de la pila: ")
38    print(p.esta_vacia())
39    print("Lista de elementos: ")
40    print(p.imprimir())
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\gabri\OneDrive\Documentos\Análisis y diseño de sistemas\Pilas> & C
Estado de la pila:
True
Ingrese un elemento de tipo entero: 1
Ingrese un elemento de tipo decimal: 1.12
Ingrese un elemento de tipo cadena: hola
Estado de la pila:
False
Lista de elementos:
[1, 1.12, 'hola']
Tamaño de la pila:
3
Eliminar elemento
hola
Estado de la pila:
False
Lista de elementos:
[1, 1.12]
PS C:\Users\gabri\OneDrive\Documentos\Análisis y diseño de sistemas\Pilas> |
```