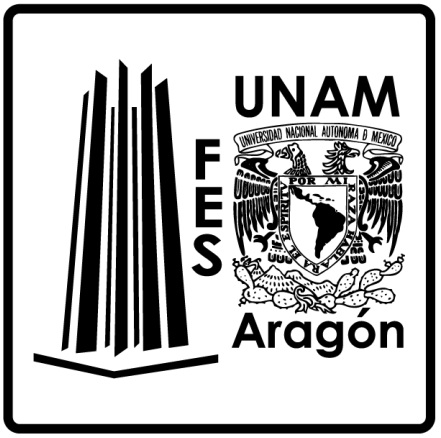


UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

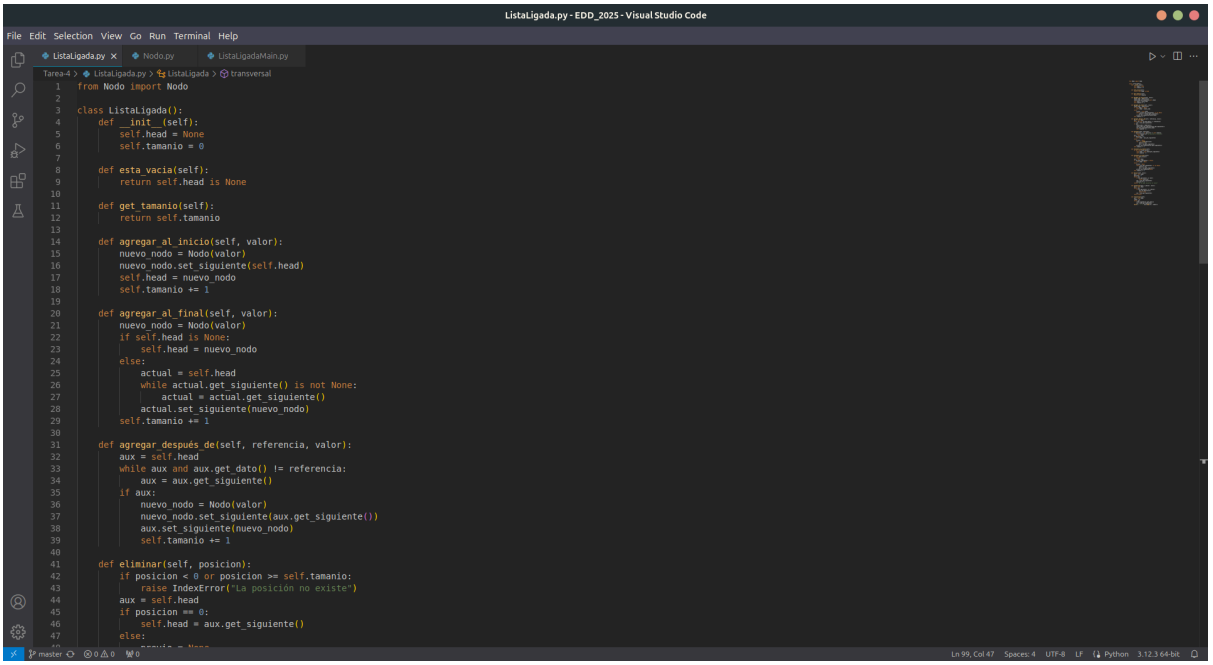
INGENIERÍA EN COMPUTACIÓN

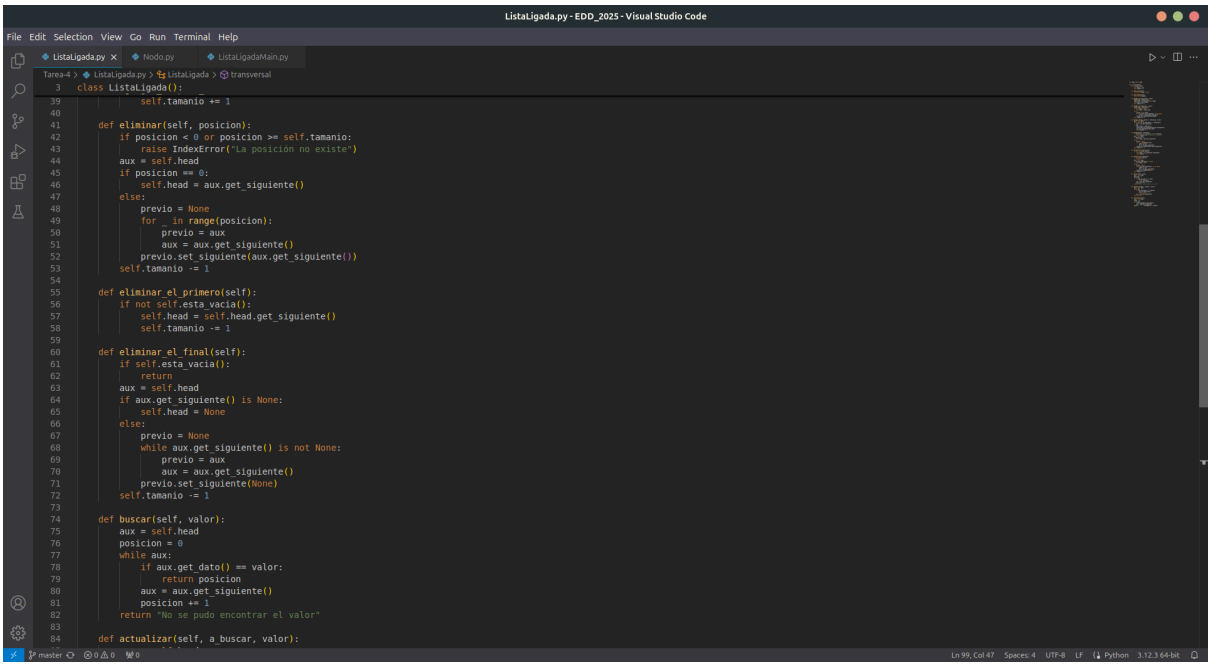
Estructura de Datos

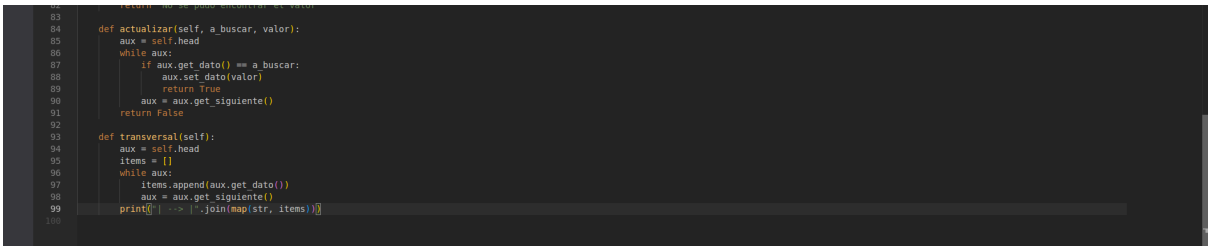
Profesor. Jesús Hernández Cabrera

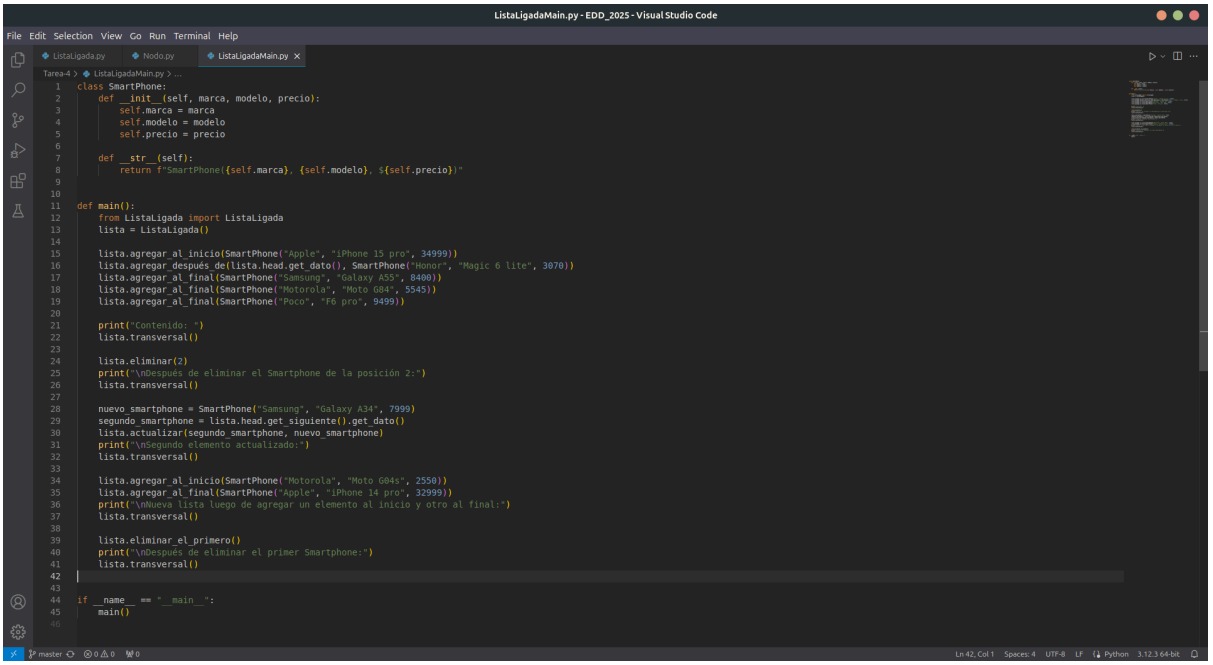
Grupo: 1360

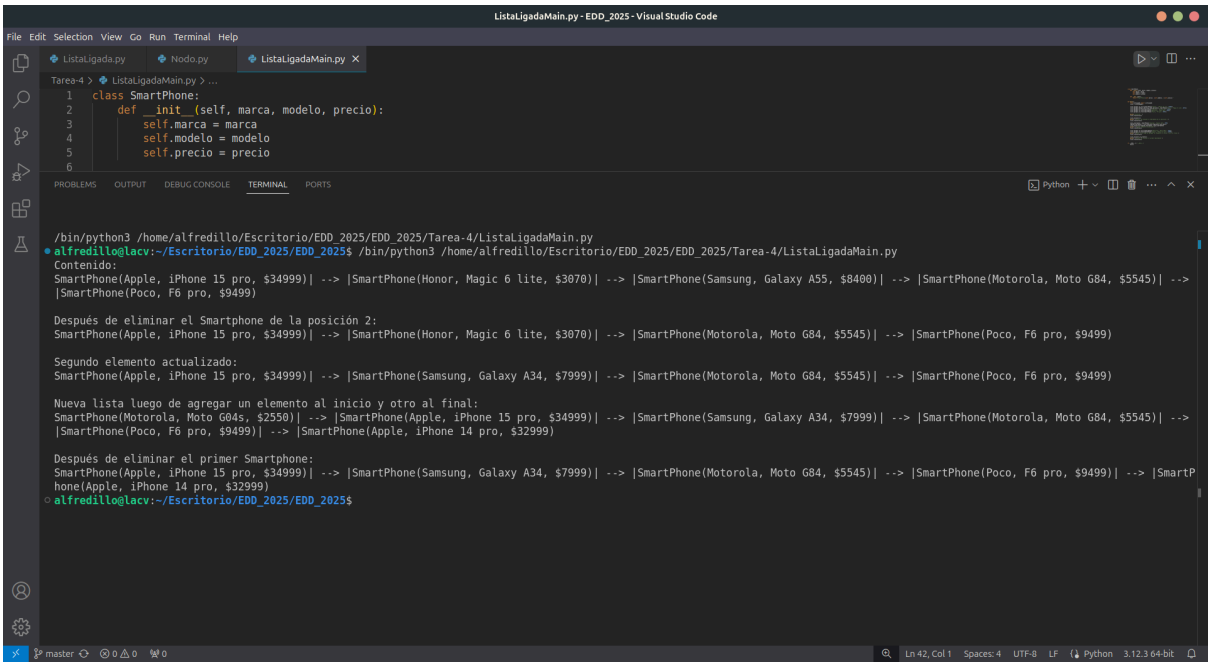
Castro Vázquez Luis Alfredo


ListaLigada.py - EDD_2025 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
ListaLigada.py
Nodo.py
ListaLigadaMain.py
Tarea-4 > ListaLigada.py > ListaLigada > transversal
from Nodo import Nodo
class ListaLigada():
def __init__(self):
self.head = None
self.tamanio = 0
def esta_vacia(self):
return self.head is None
def get_tamanio(self):
return self.tamanio
def agregar_al_inicio(self, valor):
nuevo_nodo = Nodo(valor)
nuevo_nodo.set_siguiente(self.head)
self.head = nuevo_nodo
self.tamanio += 1
def agregar_al_final(self, valor):
nuevo_nodo = Nodo(valor)
if self.head is None:
self.head = nuevo_nodo
else:
actual = self.head
while actual.get_siguiente() is not None:
actual = actual.get_siguiente()
actual.set_siguiente(nuevo_nodo)
self.tamanio += 1
def agregar_después_de(self, referencia, valor):
aux = self.head
while aux and aux.get_dato() != referencia:
aux = aux.get_siguiente()
if aux:
nuevo_nodo = Nodo(valor)
nuevo_nodo.set_siguiente(aux.get_siguiente())
aux.set_siguiente(nuevo_nodo)
self.tamanio += 1
def eliminar(self, posicion):
if posicion < 0 or posicion >= self.tamanio:
raise IndexError("La posición no existe")
aux = self.head
if posicion == 0:
self.head = aux.get_siguiente()
else:
master
Ln 99, Col 47
Spaces: 4
UTF-8
LF
Python
3.12.3 64-bit


ListaLigada.py - EDD_2025 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
ListaLigada.py
Nodo.py
ListaLigadaMain.py
Tarea-4 > ListaLigada.py > ListaLigada > transversal
class ListaLigada():
self.tamanio += 1
def eliminar(self, posicion):
if posicion < 0 or posicion >= self.tamanio:
raise IndexError("La posición no existe")
aux = self.head
if posicion == 0:
self.head = aux.get_siguiente()
else:
previo = None
for _ in range(posicion):
previo = aux
aux = aux.get_siguiente()
previo.set_siguiente(aux.get_siguiente())
self.tamanio -= 1
def eliminar_el_primero(self):
if not self.esta_vacia():
self.head = self.head.get_siguiente()
self.tamanio -= 1
def eliminar_el_final(self):
if self.esta_vacia():
return
aux = self.head
if aux.get_siguiente() is None:
self.head = None
else:
previo = None
while aux.get_siguiente() is not None:
previo = aux
aux = aux.get_siguiente()
previo.set_siguiente(None)
self.tamanio -= 1
def buscar(self, valor):
aux = self.head
posicion = 0
while aux:
if aux.get_dato() == valor:
return posicion
aux = aux.get_siguiente()
posicion += 1
return "No se pudo encontrar el valor"
def actualizar(self, a_buscar, valor):
master
Ln 99, Col 47
Spaces: 4
UTF-8
LF
Python
3.12.3 64-bit


def actualizar(self, a_buscar, valor):
aux = self.head
while aux:
if aux.get_dato() == a_buscar:
aux.set_dato(valor)
return True
aux = aux.get_siguiente()
return False
def transversal(self):
aux = self.head
items = []
while aux:
items.append(aux.get_dato())
aux = aux.get_siguiente()
print(" | --> |".join(map(str, items)))


ListaLigadaMain.py - EDD_2025 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
ListaLigada.py
Nodo.py
ListaLigadaMain.py
Tarea-4 > ListaLigadaMain.py > ...
class SmartPhone:
def __init__(self, marca, modelo, precio):
self.marca = marca
self.modelo = modelo
self.precio = precio
def __str__(self):
return f"SmartPhone({self.marca}, {self.modelo}, ${self.precio})"
def main():
from ListaLigada import ListaLigada
lista = ListaLigada()
lista.agregar_al_inicio(SmartPhone("Apple", "iPhone 15 pro", 34999))
lista.agregar_después_de(lista.head.get_dato(), SmartPhone("Honor", "Magic 6 lite", 3070))
lista.agregar_al_final(SmartPhone("Samsung", "Galaxy A55", 8400))
lista.agregar_al_final(SmartPhone("Motorola", "Moto G84", 5545))
lista.agregar_al_final(SmartPhone("Poco", "F6 pro", 9499))
print("Contenido: ")
lista.transversal()
lista.eliminar(2)
print("\nDespués de eliminar el Smartphone de la posición 2:")
lista.transversal()
nuevo_smartphone = SmartPhone("Samsung", "Galaxy A34", 7999)
segundo_smartphone = lista.head.get_siguiente().get_dato()
lista.actualizar(segundo_smartphone, nuevo_smartphone)
print("\nSegundo elemento actualizado:")
lista.transversal()
lista.agregar_al_inicio(SmartPhone("Motorola", "Moto G04s", 2550))
lista.agregar_al_final(SmartPhone("Apple", "iPhone 14 pro", 32999))
print("\nNueva lista luego de agregar un elemento al inicio y otro al final:")
lista.transversal()
lista.eliminar_el_primero()
print("\nDespués de eliminar el primer Smartphone:")
lista.transversal()
if __name__ == "__main__":
main()
master
Ln 42, Col 1
Spaces: 4
UTF-8
LF
Python
3.12.3 64-bit


ListaLigadaMain.py - EDD_2025 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
ListaLigada.py
Nodo.py
ListaLigadaMain.py
Tarea-4 > ListaLigadaMain.py > ...
class SmartPhone:
def __init__(self, marca, modelo, precio):
self.marca = marca
self.modelo = modelo
self.precio = precio
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Python
/bin/python3 /home/alfredillo/Escritorio/EDD_2025/EDD_2025/Tarea-4/ListaLigadaMain.py
alfredillo@lacv:~/Escritorio/EDD_2025/EDD_2025$ /bin/python3 /home/alfredillo/Escritorio/EDD_2025/EDD_2025/Tarea-4/ListaLigadaMain.py
Contenido:
SmartPhone(Apple, iPhone 15 pro, $34999)| --> |SmartPhone(Honor, Magic 6 lite, $3070)| --> |SmartPhone(Samsung, Galaxy A55, $8400)| --> |SmartPhone(Motorola, Moto G84, $5545)| --> |SmartPhone(Poco, F6 pro, $9499)
Después de eliminar el Smartphone de la posición 2:
SmartPhone(Apple, iPhone 15 pro, $34999)| --> |SmartPhone(Honor, Magic 6 lite, $3070)| --> |SmartPhone(Motorola, Moto G84, $5545)| --> |SmartPhone(Poco, F6 pro, $9499)
Segundo elemento actualizado:
SmartPhone(Apple, iPhone 15 pro, $34999)| --> |SmartPhone(Samsung, Galaxy A34, $7999)| --> |SmartPhone(Motorola, Moto G84, $5545)| --> |SmartPhone(Poco, F6 pro, $9499)
Nueva lista luego de agregar un elemento al inicio y otro al final:
SmartPhone(Motorola, Moto G04s, $2550)| --> |SmartPhone(Apple, iPhone 15 pro, $34999)| --> |SmartPhone(Samsung, Galaxy A34, $7999)| --> |SmartPhone(Motorola, Moto G84, $5545)| --> |SmartPhone(Poco, F6 pro, $9499)| --> |SmartPhone(Apple, iPhone 14 pro, $32999)
Después de eliminar el primer Smartphone:
SmartPhone(Apple, iPhone 15 pro, $34999)| --> |SmartPhone(Samsung, Galaxy A34, $7999)| --> |SmartPhone(Motorola, Moto G84, $5545)| --> |SmartPhone(Poco, F6 pro, $9499)| --> |SmartPhone(Apple, iPhone 14 pro, $32999)
alfredillo@lacv:~/Escritorio/EDD_2025/EDD_2025$
master
Ln 42, Col 1
Spaces: 4
UTF-8
LF
Python
3.12.3 64-bit