**MAPA – Material de Avaliação Prática da Aprendizagem**

|  |  |  |
| --- | --- | --- |
| **Acadêmico: Lucas Antonio Rocha Santos** | **R.A. 22134899-5** | |
| **Curso: Ciências de Dados e Análise de Comportamento** | | |
| **Disciplina: Estrutura De Dados** | | |
| **Valor da atividade: 3,00** | | **Prazo: 07/07/2023** |

class Pilha:

    def \_\_init\_\_(self):

        self.itens = []

    def vazia(self):

        return len(self.itens) == 0

    def push(self, item):

        self.itens.append(item)

    def pop(self):

        if not self.vazia():

            return self.itens.pop()

        else:

            return None

    def topo(self):

        if not self.vazia():

            return self.itens[-1]

        else:

            return None

    def limpar(self):

        self.itens = []

p = Pilha()

while True:

    print("\nDigite 1 para adicionar uma palavra à pilha")

    print("Digite 2 para desempilhar a palavra")

    print("Digite 3 para limpar a pilha")

    print("Digite 4 para exibir o topo da pilha")

    print("Digite 0 para sair")

    opcao = int(input("\nEscolha uma opção: "))

    if opcao == 1:

        palavra = input("\nDigite a palavra a ser adicionada: ")

        for letra in palavra:

            p.push(letra)

        print(f"\nPalavra '{palavra}' adicionada à pilha.")

    elif opcao == 2:

        palavra\_invertida = ""

        while not p.vazia():

            palavra\_invertida += p.pop()

        print(f"\nPalavra desempilhada: '{palavra\_invertida}'")

    elif opcao == 3:

        p.limpar()

        print("\nPilha limpa.")

    elif opcao == 4:

        topo = p.topo()

        if topo is None:

            print("\nPilha vazia. Nenhum elemento para mostrar.")

        else:

            print(f"\nElemento no topo da pilha: {topo}")

    elif opcao == 0:

        break

    else:

        print("\nOpção inválida. Digite 1, 2, 3, 4 ou 0.")