

Avaliação: ED1

1. **Descreva em Python um algoritmo para inserir um novo nó em uma lista simplesmente ligada de forma que a lista permaneça ordenada em ordem crescente.**

```
class Node:
```

```
    def __init__(self, valor):
```

```
        self.valor = valor
```

```
        self.prox = None
```

```
def inserir_ordenado(head, novo_valor):
```

```
    novo = Node(novo_valor)
```

```
    # Caso 1: Inserção no início
```

```
    if head is None or novo_valor < head.valor:
```

```
        novo.prox = head
```

```
        return novo
```

```
    # Caso 2: Inserção no meio/fim
```

```
    atual = head
```

```
    while atual.prox and atual.prox.valor < novo_valor:
```

```
        atual = atual.prox
```

```
    novo.prox = atual.prox
```

```
    atual.prox = novo
```

```
    return head
```

- 2. Descreva em Python como mesclar duas listas simplesmente ligadas ordenadas em uma única lista também ordenada, ajustando os ponteiros.**

```
def mesclar_listas(l1, l2):  
    dummy = Node(0)  
    atual = dummy  
  
    while l1 and l2:  
        if l1.valor < l2.valor:  
            atual.prox = l1  
            l1 = l1.prox  
        else:  
            atual.prox = l2  
            l2 = l2.prox  
        atual = atual.prox  
  
    # Liga o restante de uma das listas  
    atual.prox = l1 if l1 else l2  
    return dummy.prox
```

- 3. Escreva um algoritmo em Python para remover nós duplicados de uma lista simplesmente ligada ordenada.**

```
def remover_duplicatas(head):  
    atual = head  
    while atual and atual.prox:  
        if atual.valor == atual.prox.valor:  
            atual.prox = atual.prox.prox # pula duplicata  
        else:  
            atual = atual.prox  
    return head
```