

Lista 2 – Complexidade Ciclomática

INFORMAÇÕES DOCENTE						
CURSO:	DISCIPLINA:	TURNO	MANHÃ	TARDE	NOITE	PERÍODO/SALA:
ENGENHARIA DE SOFTWARE	FUNDAMENTOS DE PROJETO E ANÁLISE DE ALGORITMOS				x	
PROFESSOR (A): João Paulo Carneiro Aramuni						

Lista 2

Complexidade Ciclomática

1) O algoritmo abaixo implementa uma função que encontra o maior valor dentro de uma lista.

```

1 def max(lista):
2     temp = lista[0] # Inicializa temp com o primeiro elemento da lista
3     for i in range(1, len(lista)): # Percorre os elementos do índice 1 até o final
4         if temp < lista[i]: # Se encontrar um elemento maior que temp
5             temp = lista[i] # Atualiza temp com esse valor
6     return temp # Retorna o maior valor encontrado

```

1. Monte o grafo de fluxo de controle da função:

- Identifique os nós (representando os pontos de decisão e instruções da função).
- Identifique as arestas (representando as transições entre os nós).

2. Calcule a complexidade ciclomática da função usando a fórmula:

$$M = E - N + 2P$$

- Onde: E é o número de arestas no grafo.
- N é o número de nós no grafo.
- P é o número de componentes conexos (neste caso, $P = 1$, pois a função é uma unidade única).

3. Interprete o valor da complexidade ciclomática:

- Explique o que significa o valor obtido para o número de caminhos independentes no código.

4. Descreva os caminhos independentes possíveis no grafo de fluxo de controle para essa função.

2) O algoritmo abaixo retorna o maior e o menor valor dentro de uma lista de números.

```
1 def max_min(lista):
2     temp = [lista[0], lista[0]] # Inicializa temp com o primeiro elemento como maior e menor
3     for i in range(1, len(lista)): # Percorre a lista do índice 1 até o final
4         if temp[0] < lista[i]: # Se encontrar um número maior que temp[0]
5             temp[0] = lista[i] # Atualiza temp[0] com esse valor (maior número)
6         elif temp[1] > lista[i]: # Se encontrar um número menor que temp[1]
7             temp[1] = lista[i] # Atualiza temp[1] com esse valor (menor número)
8     return temp # Retorna uma lista [maior_valor, menor_valor]
```

1. Monte o grafo de fluxo de controle da função:

- Identifique os nós (representando os pontos de decisão e instruções da função).
- Identifique as arestas (representando as transições entre os nós).

2. Calcule a complexidade ciclomática da função usando a fórmula:

$$M = E - N + 2P$$

- Onde: E é o número de arestas no grafo.
- N é o número de nós no grafo.
- P é o número de componentes conexos (neste caso, $P = 1$, pois a função é uma unidade única).

3. Interprete o valor da complexidade ciclomática:

- Explique o que significa o valor obtido para o número de caminhos independentes no código.

4. Descreva os caminhos independentes possíveis no grafo de fluxo de controle para essa função.
