

Universidade Federal de Ouro Preto - UFOP
Departamento de Ciência da Computação - DECOM

Relatório atividade 1

BCC402 - ALGORITMOS E PROGRAMACAO AVANCADA

Kayo Xavier Nascimento Cavalcante Leite - 21.2.4095

Professor: Rafael Alves Bonfim

Ouro Preto
31 de março de 2025

Sumário

1	Código e enunciado.	1
2	Pseudocódigo e descrição do problema	1
3	Casos teste - Input e output esperado.	2

Lista de Códigos Fonte

1	Pseudocódigo do problema.	2
---	-----------------------------------	---

1 Código e enunciado.

Na Atividade 1 - o problema selecionado foi o de Australian Voting. O Problema simula um sistema eleitoral de votação preferencial, onde eleitores classificam os candidatos por ordem de preferência. O objetivo é determinar o vencedor seguindo regras específicas de contagem e eliminação. O código comentado e documentado, casos de teste e executável pré compilado se encontram no .zip da atividade. O código foi feito com base na referência encontrada no site:

<https://github.com/Diusrex/UVA-Solutions/blob/master/10142%20Australian%20Voting.cpp>

Caso queira, para rodar e compilar o código, é necessário ter o compiler g++ e utilizar o seguinte comando no terminal dentro do diretório da pasta da atividade específica:

Compilando e rodando o exercício

```
para compilar:
g++ AustralianVoting.cpp -o executavel

e para rodar basta utilizar .\executavel no cmd.

para utilizar os cenários de teste:
.\executavel < sampleinput.txt
.\executavel < testinput.txt
```

2 Pseudocódigo e descrição do problema

Funcionamento Básico: **Descrição:** Sistema eleitoral que determina o vencedor através de múltiplas rodadas de eliminação e redistribuição de votos.

Entrada:

- Número de candidatos C e seus nomes
- Número de cédulas N com preferências ordinais (ex: 1 3 2)

Saída:

- Nome(s) do(s) candidato(s) vencedor(es)

Passos:

1. Contagem Inicial:

- Para cada cédula, conte o primeiro candidato não eliminado
- Se algum candidato tem $> 50\%$ dos votos válidos: declare vencedor

2. Eliminação Iterativa:

- Encontre candidato(s) com menor número de votos
- Em caso de empate: elimine todos os candidatos empatados
- Redistribua votos dos eliminados para a próxima preferência válida

3. Condições de Término:

- Um candidato alcança maioria absoluta
- Todos candidatos restantes têm igual número de votos (empate)

Casos Especiais:

- **Cédula Esgotada:** Ignorar se não houver preferências válidas restantes

- **Eliminação em Massa:** Todos candidatos com votos mínimos são removidos simultaneamente

```
1 // Para cada caso de teste:
2 // 1. Ler Nomes dos Candidatos (N).
3 // 2. Ler Cédulas de Votacao (preferencias de cada eleitor).
4 // 3. Marcar todos os candidatos como NAO eliminados.
5
6 // 4. Loop (ate encontrar vencedor ou empate):
7 // 5.   Zerar contagem de votos atual para todos.
8 // 6.   Para cada cedula de eleitor:
9 // 7.     Encontrar o primeiro candidato na ordem de preferencia que NAO esta
        eliminado.
10 // 8.     Incrementar o voto para esse candidato encontrado.
11
12 // 9.   Encontrar o MAIOR e o MENOR numero de votos entre os candidatos NAO
        eliminados.
13 // 10.  Se MAIOR > 50% do total de votos OU MAIOR == MENOR (empate entre
        restantes):
14 // 11.   Marcar a contagem MAIOR como a contagem vencedora.
15 // 12.   Sair do Loop.
16 // 13.   Senao: // Ninguem tem maioria e nao ha empate total
17 // 14.   Eliminar todos os candidatos NAO eliminados que tiveram o MENOR
        numero de votos.
18
19 // 15. Imprimir os nomes de todos os candidatos NAO eliminados que tem a
        contagem vencedora de votos.
```

Código 1: Pseudocódigo do problema.

3 Casos teste - Input e output esperado.

Para os casos de teste do problema, foi disponibilizado junto a pasta do mesmo os seguintes arquivos :sampleinput.txt e testinput.txt, sendo o primeiro o próprio caso de teste disponibilizado pelo exercício e o segundo um caso de teste encontrado na plataforma <https://www.udebug.com/>. Além disso, encontra-se também o arquivo com os outputs esperados para cada input. Ambos os resultados foram validados e tiveram o output esperado.

sampleinput.txt

```
1
3
John Doe
Jane Smith
Sirhan Sirhan
1 2 3
2 1 3
2 3 1
1 2 3
3 1 2
```

output esperado

John Doe

O testinput.txt é muito grande para caber no relatório