



Universidade Federal
do Espírito Santo

Departamento de Computação e Eletrônica - CEUNES
PROGRAMAÇÃO I / PROGRAMAÇÃO FUNCIONAL
Prof. Oberlan Romão

Eleições



Como todos sabem, o sistema eleitoral brasileiro utiliza urnas eletrônicas para computar a votação dos candidatos. Apesar de muitos não acreditarem na segurança das urnas (opinião não é argumento), nunca foi provado que algum resultado foi fraudado. Além disso, desde de 2009 o TSE (Tribunal Superior Eleitoral) realiza “o [Teste Público de Segurança](#) do sistema eletrônico de votação, ocasião em que investigadores inscritos apresentam e executam planos de ataque aos componentes externos e internos da urna eletrônica”.

Neste miniEP, sua tarefa é implementar um programa que simule o funcionamento de uma urna eletrônica, ou seja, dadas as informações necessárias, informe a quantidade de votos que cada candidato teve, o número de votos em branco e votos inválidos.

Descrição da entrada

A primeira linha contém um número inteiro N , que representa o número de candidatos (é garantido que $2 \leq N \leq 100$). As próximas N linhas representam o nome de cada candidato, sendo que a posição (começando em 1) que o candidato aparece na lista representa o seu número, por exemplo, se Maria aparece na posição i então seu número (para votação) é i ($1 \leq i \leq N$). A próxima linha contém um número inteiro V indicando o número de votos a ser computado. As próximas V linhas indicam o voto da pessoa p . Seja v_p o voto da pessoa p :

- Se v_p for 0 indica voto em branco;
- Se $1 \leq v_p \leq N$ significa voto no candidato v_p ;
- Se o $v_p > N$ indica voto nulo.

Observe o exemplo abaixo:

```
3
Maria
Enzo
Lara
7
3
1
0
1
1
2
5
```

No exemplo anterior, temos 3 candidatos (Maria, Enzo e Lara). Em seguida, é informado o número de votos/eleitores (7), seguido do voto de cada um dos 7 eleitores. Para essa entrada, temos as seguintes informações:

- Maria: 3 votos
- Enzo: 1 voto
- Lara: 1 voto
- Votos brancos: 1 voto
- Votos nulos: 1 voto

Para evitar problemas com codificação, todas as *strings* foram escritas propositalmente sem acentos.

Descrição da saída

A saída deverá apresentar um breve relatório da votação descrevendo:

- Nome e número de votos de cada candidato;
- O número de votos em branco;
- O número de votos nulos;
- O candidato vencedor (é garantido que não haverá empate e que um dos candidatos vencerá).

Observe o exemplo abaixo (de acordo com a entrada anterior):

```
Maria: 3
Enzo: 1
Lara: 1
Branco: 1
Nulos: 1
Vencedor(a): Maria
```

Dica: Use uma lista para armazenar os votos, onde a posição i representa o número de votos do candidato i . A posição 0 pode ser usada para armazenar a quantidade de votos em brancos e a posição $N + 1$ os votos nulos.

O que entregar

Nesse miniEP você deve enviar, pelo **AVA**, apenas um arquivo, chamado **mEP5.py**, contendo o código do seu programa. O peso desse miniEP é 3.

Data de entrega: até às 6h do dia 09/02/2023.

Observações:

1. Não é permitido usar **variáveis globais**, **estruturas de repetição (loop)**, como **while**, **for**, **funções impuras** e operações que não sejam do Paradigma Funcional (por ex., `lista.index()`, `lista.count()`, `lista.sort()`, `lista.reverse()`, `lista.append()`, `str.lower()`, `str.upper()`, `str.replace()`, etc.). A utilização dessas estruturas/funções implicará em nota 0. Se não tem certeza se pode ou não usar determinada estrutura/função, me pergunte.
2. Funções que modificam uma lista passada como argumento, devem ser chamadas passando uma cópia da lista. Isso evita que a função se torne impura (leia a observação 1). Leia mais [aqui](#);
3. Quando necessário, utilize funções recursivas;
4. Use apenas instruções/comandos visto em sala de aula (teórica ou prática);
5. A submissão de um código que não implementa o algoritmo requisitado, mas que exibe as saídas esperadas dos testes abertos a partir da comparação de trechos da entrada será considerada fraude e acarretará a atribuição de nota 0;
6. Em caso de plágio, será atribuído 0 a todos os envolvidos.