

*Definição do trabalho da M2**URNA ELETRÔNICA*

Data de entrega: 26/10/2023. (até 18:00)

Modalidade: em TRIOS.

Visão Geral:



A urna eletrônica começou a ser adotada no processo eleitoral brasileiro em 1996 e já contabiliza várias eleições bem-sucedidas visto que a Justiça Eleitoral utiliza o que há de mais moderno em termos de segurança da informação para garantir a integridade, a confiabilidade, a transparência e a autenticidade do processo eleitoral.

REGRAS PARA O DESENVOLVIMENTO

Desenvolva um código em C++ de controle de funcionamento da urna eletrônica. O código da urna deverá ler e validar o CPF do eleitor. Após ter validado o CPF do eleitor a urna deverá coletar o voto para governador e depois para presidente. Para finalizar a votação de todos os eleitores (encerramento da urna) deverá ser digitado o número 999999999999 como CPF. Encerrada a votação deverá ser apresentado a quantidade de eleitores, a quantidade de votos de cada candidato a governador e a quantidade de votos de cada candidato a presidente, assim como a quantidade de votos em branco e nulo tanto para governador como para presidente. Também deverá ser apresentada a porcentagem de votos válidos (total de votos menos os votos nulos e os votos em branco) para cada candidato a governador e a presidente, e finalmente a informação dizendo se haverá segundo turno ou não. Não havendo segundo turno, informar quem foi eleito (governador e presidente).

Os candidatos e seus respectivos números (partidos) não deverão fazer alusão a qualquer candidato que tenha concorrido a pleitos anteriores. Considerem 3 candidatos a governador e 3 a presidente.

O CPF deverá ser lido como UM número inteiro de 10 ou 11 dígitos.

Obs: inteiros de 11 dígitos são números extremamente grandes e necessitam de 2 ampliadores de faixa (long) para serem corretamente armazenados.

Validação do CPF:

Para exemplificar o processo será usado o CPF: 111.444.777-35, que deverá ser digitado como 11144477735.

Calculando o Primeiro Dígito Verificador

O primeiro dígito verificador do CPF é calculado utilizando-se o seguinte algoritmo.

- 1) **Distribua os 9 primeiros dígitos em um quadro colocando os pesos 10, 9, 8, 7, 6, 5, 4, 3, 2** abaixo da esquerda para a direita, conforme representação abaixo:

1	1	1	4	4	4	7	7	7
10	9	8	7	6	5	4	3	2

- 2) **Multiplique os valores de cada coluna:**

1	1	1	4	4	4	7	7	7
10	9	8	7	6	5	4	3	2
10	9	8	28	24	20	28	21	14

- 3) **Calcule o somatório dos resultados (10+9+...+21+14) = 162.**
- 4) **O resultado obtido (162) será dividido por 11.** Considere como quociente apenas o valor inteiro, o **resto da divisão** será responsável pelo cálculo do primeiro dígito verificador.

Obs: 162 dividido por 11 obtêm-se 14 como quociente e 8 como resto da divisão. Caso o resto da divisão seja menor que 2, o primeiro dígito verificador se torna 0 (zero), caso contrário subtrai-se o valor obtido de 11, que é nosso caso. Sendo assim o dígito verificador é 11-8, ou seja, 3 (três). Já se tem parte do CPF: 111.444.777-3X.

Calculando o Segundo Dígito Verificador

- 1) **Para o cálculo do segundo dígito será usado o primeiro dígito verificador já calculado. Será montada uma tabela semelhante a anterior só que desta vez usando na segunda linha os valores 11,10,9,8,7,6,5,4,3,2**, já que está sendo incorporado mais um algarismo para esse cálculo. Veja:

1	1	1	4	4	4	7	7	7	3
11	10	9	8	7	6	5	4	3	2

- 2) **Na próxima etapa será feito como na situação do cálculo do primeiro dígito verificador**, multiplicar os valores de cada coluna e efetuar o somatório dos resultados obtidos: $(11+10+\dots+21+6) = 204$.

1	1	1	4	4	4	7	7	7	3
11	10	9	8	7	6	5	4	3	2
11	10	9	32	28	24	35	28	21	6

- 3) **Novamente será realizado o cálculo do módulo 11**. Será dividido o total do somatório por 11 e considerado o resto da divisão. Acompanhando: 204 dividido por 11 obtêm-se 18 como quociente e 6 como resto da divisão.
- 4) Caso o valor do resto da divisão seja menor que 2, esse valor passa automaticamente a ser zero, caso contrário (como no nosso caso) é necessário subtrair o valor obtido de 11 para se obter o dígito verificador.
- Logo, $11 - 6 = 5$, que é o segundo dígito verificador.

Neste caso chega-se ao final dos cálculos e descobre-se que os dígitos verificadores do CPF hipotético são os números **3** e **5**, portanto pode-se afirmar que o CPF é válido, pois o dígito verificador encontrado é igual ao digitado pelo usuário:

$$35 == 111444777\mathbf{35}.$$

- 5) Caso os dígitos verificadores calculados sejam diferentes dos informados pelo eleitor, deverá ser apresentado uma mensagem informando que o CPF é inválido que o usuário deverá digitar novamente o seu CPF.
- A votação só deverá ser iniciada após ter sido informado um CPF válido.

Votação:

Inicie a votação apresentando os nomes dos candidatos a governador e seus respectivos números;

Leia o voto para governador;

Limpe a tela e apresente os nomes dos candidatos a presidente e seus respectivos números;

Leia o voto para presidente;

Volte para a tela inicial onde se pede o CPF do eleitor e repita o processo até ser digitado 999999999999 como CPF.

Finalização da votação:

Apresentar o total de eleitores;

Quantidade de votos para cada candidato a governador;

Quantidade de votos nulos e em branco para governador;
Quantidade de votos para cada candidato a presidente;
Quantidade de votos nulos e em branco para presidente;
Porcentagem de votos válidos para cada candidato a governador;
Porcentagem de votos válidos para cada candidato a presidente;
Informar se haverá segundo turno ou não para governador e para presidente.
Informar os candidatos que passaram para o segundo turno ou o nome do candidato eleito.

Obs.: Para o desenvolvimento do código não poderão ser utilizadas variáveis compostas (*arrays*), funções e *structs*.

Outros dois comandos bastante úteis no desenvolvimento de programas no console, são os comandos `system("cls")` e `system("pause")`. Para o uso destes comandos é necessário acrescentar a biblioteca `<stdlib.h>`.

- `system("cls")` é um comando que limpa a tela do console (**clear screen**). Esse comando é bastante útil, pois em uma tela limpa é mais fácil dar destaque aquilo que se está mostrando no momento.
- `system("pause")` é um comando útil, principalmente quando usado em conjunto com o `system("cls")`, pois ele pausa a execução da aplicação até que o usuário aperte qualquer tecla, bastante útil quando se quer exibir algo antes de limpar a tela para iniciar uma nova execução.

Defesa (Obrigatória)

Durante a defesa serão realizados questionamento sobre o trabalho realizado pelo grupo. A defesa é obrigatória e deverá ser feita pelos integrantes do grupo na aula. Se algum integrante não estiver presente durante a aula de defesa, deverá justificar a falta, o mesmo defenderá posteriormente em data a ser agendada com o professor.

Entregas:

- *Postar no repositório criado especialmente para o trabalho no material didático: **Trabalho T2***
- *Código fonte desenvolvido obrigatoriamente em C++: é de responsabilidade do grupo verificar se o arquivo postado é o correto.*

Critérios de Avaliação:

1. *Organização e clareza do código = 10% da nota.*
2. *Identificação dos autores e Comentários pertinentes e oportunos no código = 5% da nota.*
3. *Funcionamento correto conforme a especificação = 30% da nota.*
4. *Recursos da linguagem utilizados = 15% da nota.*
5. *Apresentação/defesa do código = 40% da nota.*

Obs.: Todas as notas relativas ao código dependem do desempenho na apresentação. Sem apresentação o trabalho terá nota ZERO; todos os componentes do grupo devem ter conhecimento de todo o código e a nota da apresentação do grupo será única.