

Simulación de votación en C++

Introducción:

Lo que se presentará a continuación es un simulador de votación en C++, se explicará y mostrará el código hecho, además del problema.

Sistema de Votación

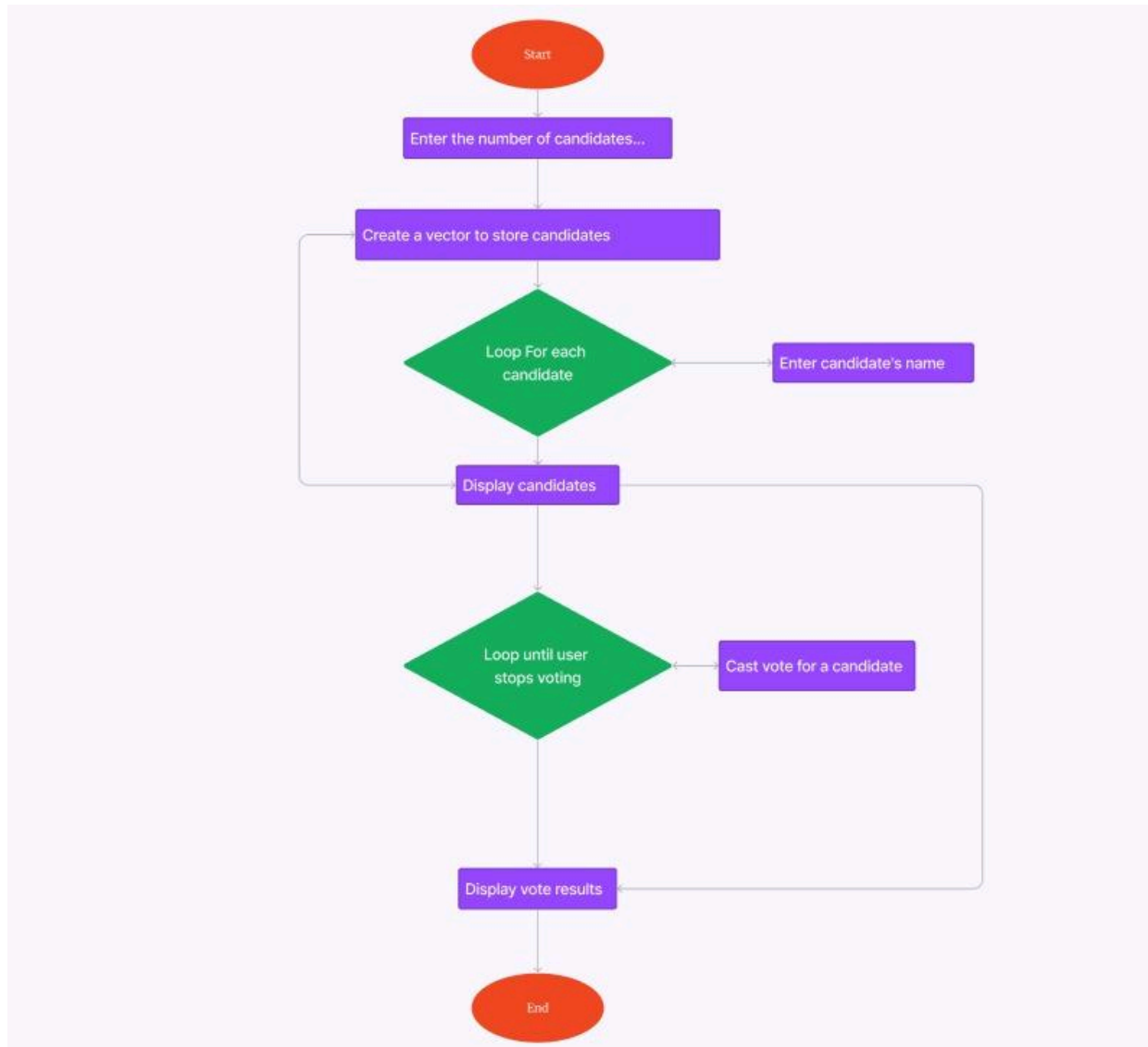
El problema consiste en desarrollar un simulador de votación en C++. Este programa debe permitir la creación de una elección donde los votantes pueden seleccionar entre múltiples candidatos. El simulador debe gestionar el registro de votos, contar los votos para cada candidato y finalmente mostrar los votos.

Relevancia e Interés

1. **Educación Cívica:** Un simulador de votación es relevante porque puede ser utilizado como una herramienta educativa para enseñar a los estudiantes y ciudadanos sobre el proceso electoral. Comprender cómo funciona una elección es crucial para fomentar una participación informada y activa en la democracia.
2. **Desarrollo de Habilidades:** Desde el punto de vista de la programación, desarrollar un simulador de votación permite a los estudiantes aplicar conceptos fundamentales como el uso de variables, bucles, condicionales y funciones. Además, les da experiencia en la planificación y estructuración de un programa, desde la concepción inicial hasta la implementación y prueba final.
3. **Transparencia y Confianza:** En un contexto más amplio, la simulación de votaciones puede ayudar a desarrollar sistemas reales que mejoren la transparencia y la confianza en los procesos electorales. Simulaciones precisas pueden ser utilizadas para probar y validar sistemas de votación antes de su implementación en elecciones reales.

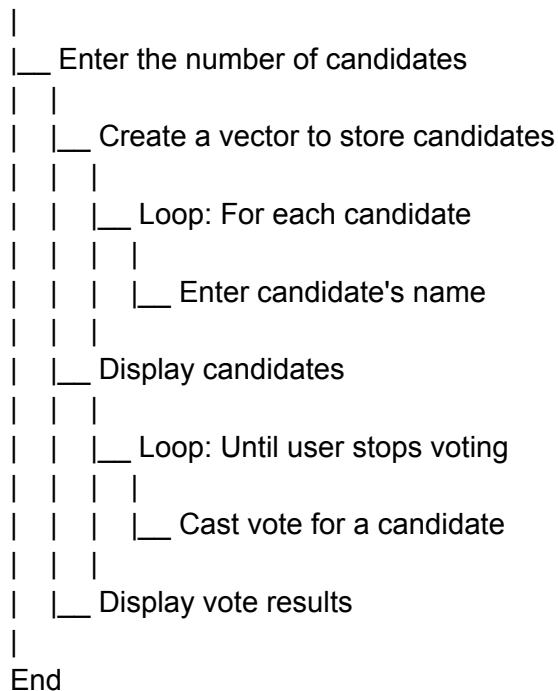
En resumen, un simulador de votación es relevante e interesante tanto desde una perspectiva educativa como práctica, proporcionando beneficios significativos para la sociedad y para el desarrollo de habilidades técnicas y analíticas en los estudiantes.

Planificación y diseño



Explicación del diagrama de flujo

Start



Start: Representado por un óvalo.

Enter the number of candidates: Representado por un rectángulo (proceso).

Create a vector to store candidates: Otro rectángulo (proceso).

Loop: For each candidate: Representado por un rombo (decisión), con una flecha de retorno al inicio del bucle.

Enter candidate's name: Rectángulo (proceso) dentro del bucle.

Display candidates: Rectángulo (proceso) después del bucle.

Loop: Until user stops voting: Otro rombo (decisión), con una flecha de retorno al inicio del bucle.

Cast vote for a candidate: Rectángulo (proceso) dentro del bucle.

Display vote results: Rectángulo (proceso) después del bucle.

End: Representado por un óvalo.

Español:

Inicio: El programa comienza.

Ingrese el número de candidatos: se le solicita al usuario que ingrese el número de candidatos.

Cree un vector para almacenar candidatos: se crea un vector de objetos candidatos para almacenar los candidatos.

Bucle: Para cada candidato: Para cada candidato, se realizan los siguientes pasos:

Ingrese el nombre del candidato: se le solicita al usuario que ingrese el nombre del candidato.

Mostrar candidatos: Se muestran los nombres de todos los candidatos.

Bucle: Hasta que el usuario deje de votar: Los siguientes pasos se realizan hasta que el usuario decida dejar de votar:

Voto emitido por un candidato: Se solicita al usuario que vote por un candidato ingresando el número correspondiente.

Mostrar resultados de votación: se muestran los resultados de la votación, mostrando el número de votos que recibió cada candidato.

Fin: El programa finaliza.

Inglés:

Start: The program begins.

Enter the number of candidates: The user is prompted to input the number of candidates.

Create a vector to store candidates: A vector of Candidate objects is created to store the candidates.

Loop: For each candidate: For each candidate, the following steps are performed:

Enter candidate's name: The user is prompted to input the name of the candidate.

Display candidates: The names of all the candidates are displayed.

Loop: Until user stops voting: The following steps are performed until the user decides to stop voting:

Cast vote for a candidate: The user is prompted to vote for a candidate by entering the corresponding number.

Display vote results: The voting results are displayed, showing the number of votes each candidate received.

End: The program ends.