Entrance form

This is a simple console-based entrance form implemented in C++. The system allows users to display input data add symbol no by program itself to multiple inputs and store info in text form by opening files and display info and search info by symbol no provided .

Structures and Functions.

Entrants details

The **program** represents a data's with the following attributes:

- **Name**: a char representing the name of the entrant

- **email**: a string representing the email address of the entrant

- **Age**: an integer representing the age of the entrant

- **address**: an integer representing the address number of the entrant

- **others** : strings representing the other functions and structure of the entrant

void display Function

- The **void display** function takes a **const Entrant &** object as an argument and displays the details of the entrant to the console. **Fill form Function**

- The **fill form** function takes a **Entrant &** object as an argument and allows the user to fill the form.

Search symbol/savesymbol/loadsymbol Function

- This function search for the symbol no which is not used by previous entrants started from 20200 and uses for loop to increase numbers for symbol no to filled in system and add another symbol number to another entrants . **Main Program**

- The main program creates entrants and create loop for multiple entrants , after completing task program ask to user press enter key to go main menu and processing their input.

Menu Options

The main menu has three options:

1. **fill out the form**: displays the structure to fill form .
2. **Display Entrant**: Displays the details of available entrant by using symbol no provided by program .

   3 .Search your information: displays your information only one information which is attached to one specific symbol no .

   4. Exit: Exits the program.

Sample Output:

Notes

- The program uses the **system("clear")** and **system("cls")** functions to clear the console screen, which may not work on all platforms.

- The program uses **cin.ignore()** and **cin.get()** to pause the program and wait for the user to press Enter before continuing.

Conclusion:

The Entrance form effectively demonstrates the fundamental concepts of C++ programming, including structures, functions, and basic user interaction. It provides a clear and functional interface for displaying available entrants, handling entrant symbol no, and exiting the application. The use of the `Entrant` structure to manage entrant details and form fill simplifies code organization and readability. While the system meets its primary objectives, improvements such as enhanced error handling and cross-platform compatibility could further enhance its functionality and user experience. Overall, the system serves as a practical and educational example of managing simple entrance form in C++.