```
=== Student System ===
1. Add Student
2. Show All Students
3. Search Student (Binary Search)
4. Save to File
5. Load from File
6. Show Data Types
7. Show Pointers
8. Show Arrays
9. Show Strings
0. Exit
```

```
Choose: 1
Enter ID: 30064758
Enter Name: Aayush Bhattarai
Enter Grade: 4
Student added!
```

```
Choose: 2

--- All Students ---
ID: 30064758, Name: Aayush Bhattarai, Grade: 4
Total: 1 students
```

```
Choose: 3
Enter ID to search: 30064758
Searching for ID: 30064758
Checking position 0 (ID: 30064758)
Found student!
Student found: ID: 30064758, Name: Aayush Bhattarai, Grade: 4
```

```
Choose: 4
Data saved to students.txt!
```

```
Choose: 5

--- Array Example ---
Array elements using bracket notation: 85 90 78 92 88
Array elements using a pointer: 85 90 78 92 88
```

```
--- String Example ---
First name: Ayush
Last name: Bhatt
Full name: Ayush Bhatt
Length of full name: 11
```

```
Choose: 6

--- Data Types and Sizes ---
int size: 4 bytes
double size: 8 bytes
string size: 24 bytes
Student size: 40 bytes
```

Program Explanation:

This program allows users to add student records (ID, name, grade), display all stored data, and search for specific students.RetryClaude can make mistakes. Please double-check responses.

Concepts:
Data types and data sizes
File types (e.g., .txt for data I/O)
Use of pointers
arrays
A binary search algorithm (on any other search)
Use of strings
File I/O (read from or write to a file)
At least two classes (with meaningful interaction)

Challenges:

Programs looped infinitely when names contained spaces
Input glitches and premature program exit after each input

Solution:

Used getline() and cin.ignore() to clear input buffer and premature operation.