SUBMITTED BY : SPOGMAI JAN

ROLL NO : 2430 – 0081

SUBJECT : PROGRAMMING FUNDAMENTAL

FINAL PROJECT

Student management system:

Project | Classes | Debug

1.student managment.cpp

```cpp
#include <iostream>
#include <string>
using namespace std;

int studentCount = 3; // Keeps track of the number of records
struct Student {
    int ID;
    string name, address, phone, email, course, semester;
    char grade;
    float marks;
    float cgpa;
} students[50] = {
    {1, "Spogmai Jan", "F-17", "03327742213", "spogmaijan011@gmail.com", "Computer Science", "1", 'A', 95},
    {2, "Hajra Sattar", "f/16", "03008123485", "hajra@0gmail.com", "Artificial Intelligence", "2", 'B', 80},
    {3, "Laibah Rauf", "PMO", "0398357821", "laibahrauf@gmail.com", "Computer Science", "3", 'A', 90}
};

// Function to calculate CGPA from marks based on new grading scale
float calculateCGPA(float marks) {
    if (marks >= 85) return 4.0;  // A
    if (marks >= 70) return 3.5;  // B
    if (marks >= 55) return 3.0;  // C
    if (marks >= 40) return 2.5;  // D
    return 0.0;  // F
}

// Function to insert a new student record
void insert_student() {
    if (studentCount >= 50) {
        cout << "Database is full. Cannot insert more students.\n";
        return;
    }

    cout << "Enter Student ID: ";
    cin >> students[studentCount].ID;
    cin.ignore();
```

Compiler | Resources | Compile Log | Debug | Find Results

Project Classes Debug | 1.student managment.cpp

```
33
34      cout << "Enter Student ID: ";
35      cin >> students[studentCount].ID;
36      cin.ignore();
37
38      cout << "Enter Student Name: ";
39      getline(cin, students[studentCount].name);
40
41      cout << "Enter Address: ";
42      getline(cin, students[studentCount].address);
43
44      cout << "Enter Phone: ";
45      cin >> students[studentCount].phone;
46
47      cout << "Enter Email: ";
48      cin >> students[studentCount].email;
49
50      cout << "Enter Course: ";
51      cin >> students[studentCount].course;
52
53      cout << "Enter Semester: ";
54      cin >> students[studentCount].semester;
55
56      cout << "Enter Marks (0-100): ";
57      cin >> students[studentCount].marks;
58
59      // Calculate CGPA based on marks
60      students[studentCount].cgpa = calculateCGPA(students[studentCount].marks);
61
62      // Assign grade based on CGPA
63      if (students[studentCount].cgpa == 4.0) {
64          students[studentCount].grade = 'A';
65      } else if (students[studentCount].cgpa == 3.5) {
66          students[studentCount].grade = 'B';
67      } else if (students[studentCount].cgpa == 3.0) {
68          students[studentCount].grade = 'C';
69      } else if (students[studentCount].cgpa == 2.5) {
```

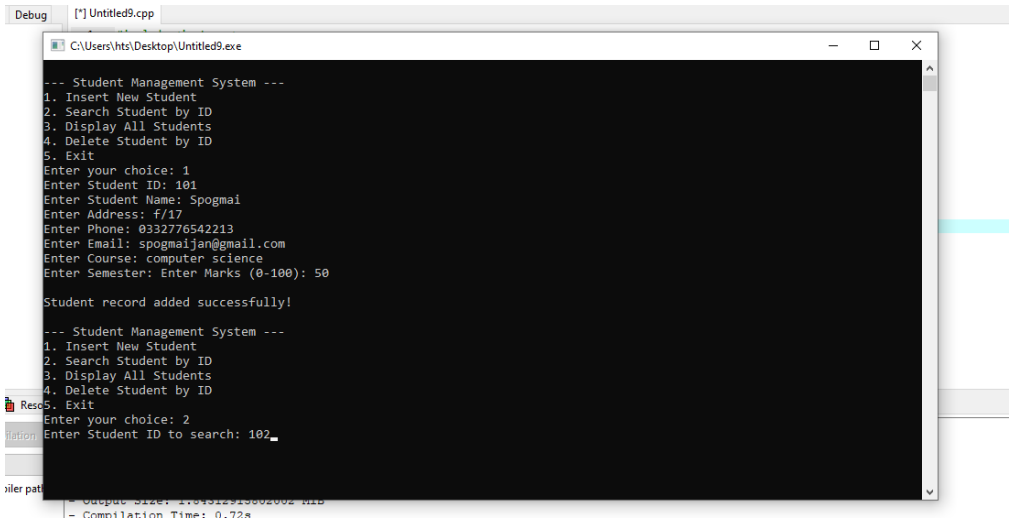
Compiler | Resources | Compile Log | Debug | Find Results

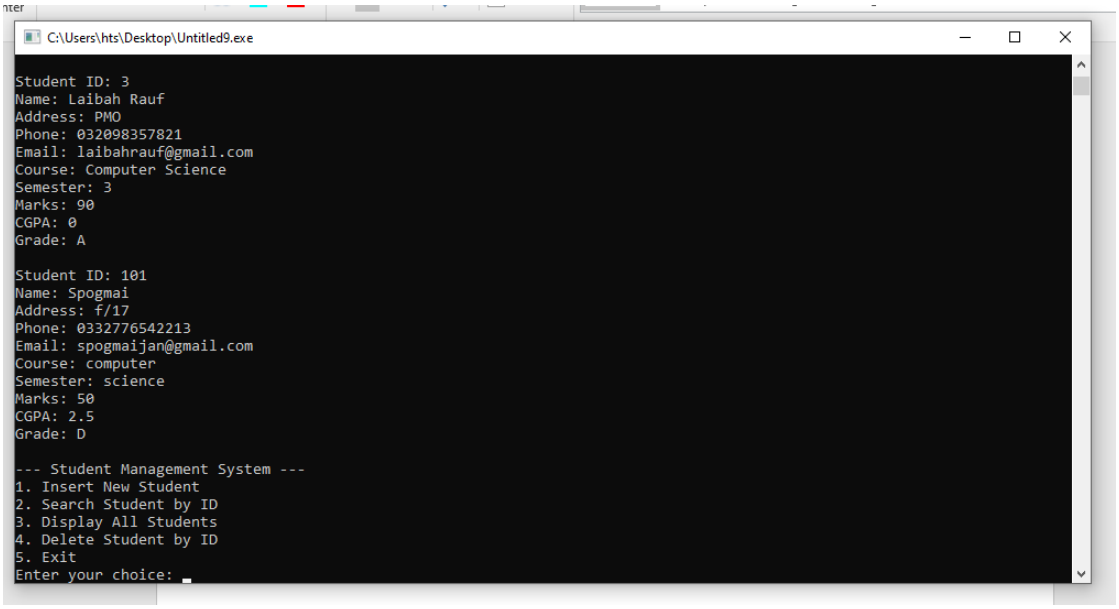
Project Classes Debug | 1.student managment.cpp

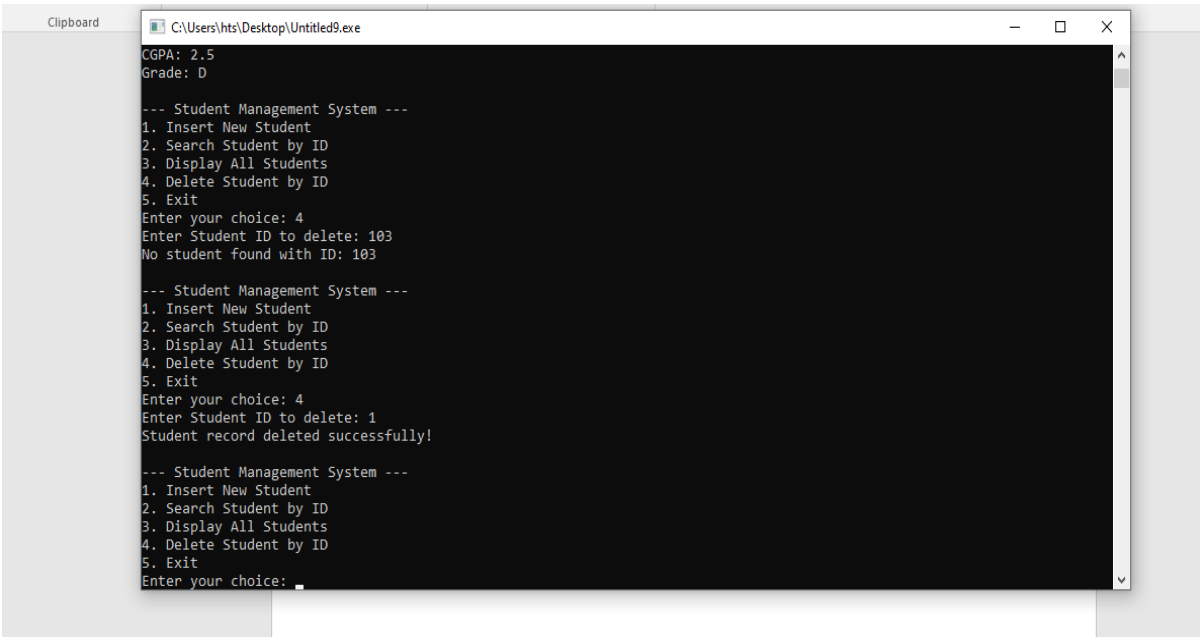
```
69      } else if (students[studentCount].cgpa == 2.5) {
70          students[studentCount].grade = 'D';
71      } else {
72          students[studentCount].grade = 'F';
73      }
74
75      studentCount++;
76      cout << "\nStudent record added successfully!\n";
77  }
78
79  // Function to display all student records
80  void display_all_students() {
81      if (studentCount == 0) {
82          cout << "No student records available.\n";
83          return;
84      }
85
86      for (int i = 0; i < studentCount; i++) {
87          cout << "\nStudent ID: " << students[i].ID;
88          cout << "\nName: " << students[i].name;
89          cout << "\nAddress: " << students[i].address;
90          cout << "\nPhone: " << students[i].phone;
91          cout << "\nEmail: " << students[i].email;
92          cout << "\nCourse: " << students[i].course;
93          cout << "\nSemester: " << students[i].semester;
94          cout << "\nMarks: " << students[i].marks;
95          cout << "\nCGPA: " << students[i].cgpa;
96          cout << "\nGrade: " << students[i].grade << "\n";
97      }
98  }
99
100 // Function to search for a student by ID
101 void search_student_by_id() {
102     int id;
103     cout << "Enter Student ID to search: ";
104     cin >> id;
```

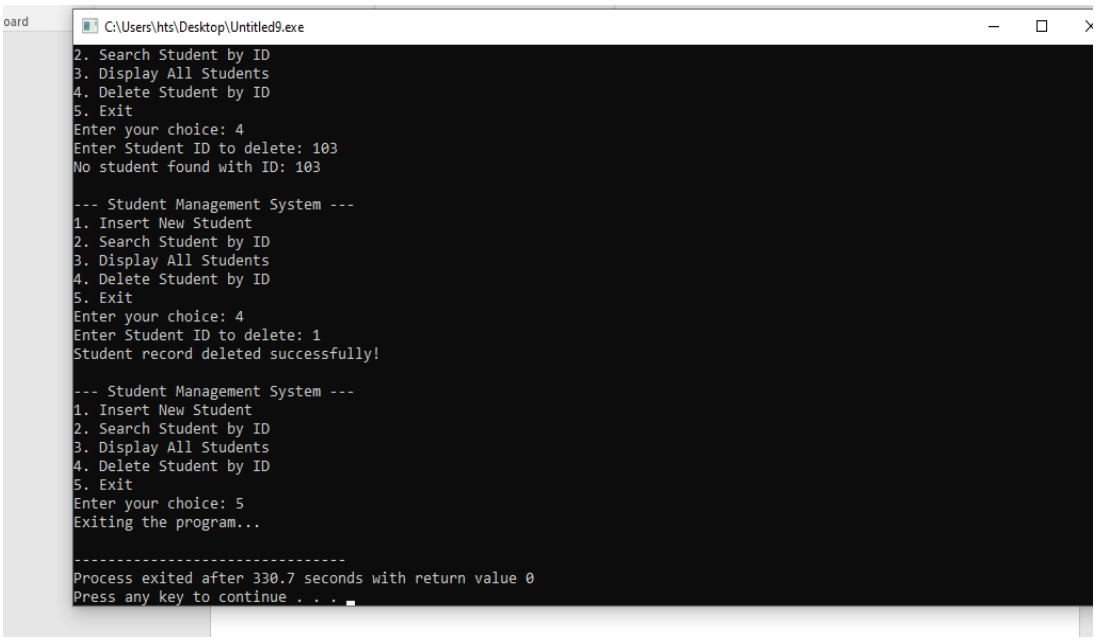
```
105
106     bool found = false;
107     for (int i = 0; i < studentCount; i++) {
108         if (students[i].ID == id) {
109             cout << "\nStudent ID: " << students[i].ID;
110             cout << "\nName: " << students[i].name;
111             cout << "\nAddress: " << students[i].address;
112             cout << "\nPhone: " << students[i].phone;
113             cout << "\nEmail: " << students[i].email;
114             cout << "\nCourse: " << students[i].course;
115             cout << "\nSemester: " << students[i].semester;
116             cout << "\nMarks: " << students[i].marks;
117             cout << "\nCGPA: " << students[i].cgpa;
118             cout << "\nGrade: " << students[i].grade << "\n";
119             found = true;
120             break;
121         }
122     }
123
124     if (!found) {
125         cout << "No student found with ID: " << id << "\n";
126     }
127 }
128
129 // Function to delete a student by ID
130 void delete_student_by_id() {
131     int id;
132     cout << "Enter Student ID to delete: ";
133     cin >> id;
134
135     bool found = false;
136     for (int i = 0; i < studentCount; i++) {
137         if (students[i].ID == id) {
138             // Shift records to delete the student
139             for (int j = i; j < studentCount - 1; j++) {
140                 students[j] = students[j + 1];
141             }
142             studentCount--; // Decrease the student count
143             cout << "Student record deleted successfully!\n";
144             found = true;
145             break;
146         }
147     }
148
149     if (!found) {
150         cout << "No student found with ID: " << id << "\n";
151     }
152 }
153
154 // Main function
155 int main() {
156     int choice;
157     while (true) {
158         cout << "\n--- Student Management System ---\n";
159         cout << "1. Insert New Student\n";
160         cout << "2. Search Student by ID\n";
161         cout << "3. Display All Students\n";
162         cout << "4. Delete Student by ID\n";
163         cout << "5. Exit\n";
164         cout << "Enter your choice: ";
165         cin >> choice;
166
167         switch (choice) {
168             case 1:
169                 insert_student();
170                 break;
171             case 2:
172                 search_student_by_id();
173                 break;
174             case 3:
175                 display_all_students();
176                 break;
```

```cpp
int main() {
    int choice;
    while (true) {
        cout << "\n--- Student Management System ---\n";
        cout << "1. Insert New Student\n";
        cout << "2. Search Student by ID\n";
        cout << "3. Display All Students\n";
        cout << "4. Delete Student by ID\n";
        cout << "5. Exit\n";
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice) {
            case 1:
                insert_student();
                break;
            case 2:
                search_student_by_id();
                break;
            case 3:
                display_all_students();
                break;
            case 4:
                delete_student_by_id();
                break;
            case 5:
                cout << "Exiting the program...\n";
                return 0;
            default:
                cout << "Invalid choice. Please try again.\n";
        }
    }

    return 0;
}
```




C:\Users\hts\Desktop\Untitled9.exe
Student ID: 3
Name: Laibah Rauf
Address: PMO
Phone: 032098357821
Email: laibahrauf@gmail.com
Course: Computer Science
Semester: 3
Marks: 90
CGPA: 0
Grade: A
Student ID: 101
Name: Spogmai
Address: f/17
Phone: 0332776542213
Email: spogmaijan@gmail.com
Course: computer
Semester: science
Marks: 50
CGPA: 2.5
Grade: D
--- Student Management System ---
1. Insert New Student
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice:


Clipboard
C:\Users\hts\Desktop\Untitled9.exe
CGPA: 2.5
Grade: D
--- Student Management System ---
1. Insert New Student
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice: 4
Enter Student ID to delete: 103
No student found with ID: 103
--- Student Management System ---
1. Insert New Student
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice: 4
Enter Student ID to delete: 1
Student record deleted successfully!
--- Student Management System ---
1. Insert New Student
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice:


oard
C:\Users\hts\Desktop\Untitled9.exe
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice: 4
Enter Student ID to delete: 103
No student found with ID: 103
--- Student Management System ---
1. Insert New Student
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice: 4
Enter Student ID to delete: 1
Student record deleted successfully!
--- Student Management System ---
1. Insert New Student
2. Search Student by ID
3. Display All Students
4. Delete Student by ID
5. Exit
Enter your choice: 5
Exiting the program...
Process exited after 330.7 seconds with return value 0
Press any key to continue . . .