SUMMARY REPORT

Title of project: Student Record Management System

Name of the Supervisor: Prof. Pragati .P .Patil.

Name of the Students:

1. ***Niranjan Janardhan Raje***
2. ***Pranav Milind Shinde***

Brief description:

Our projects aim is to help the teacher and students to fill students information and manage. By regular method we fill information but it takes time also it is filled by only teachers so it is quite difficult and after that student can't see information is it accurate data or not. So for filling and maintaining information of student in short time and accurate we develop this program . We develop this program to help the teacher and student. To complete this program we use class and object, inheritance, linked list , switch , array , file handling . By using this concepts we developed this program.

In this program admin and student can fill and both can update and see information it can store manage information like first name, last name, yea, branch, roll no and only admin can delete the record of student. this can help student and teacher to manage information of student.

Objective:

Manage data of student like roll no, branch, year, name. It provides capability for registering student. Providing platform for student and teacher to manage data of student, Allowing admin to modify data.

Outcomes:

There is constant advancement in the system as new techniques are introduced. Another new technique employed in education is the information management system. Nowadays, the information system is used by teachers to manage the students' information in class using only one application. The information management system has mostly taken root all over the world in recent years, replacing the traditional method of assessing the students information from file.

Snaps of Program:

Fig.1: Main Menu

```cpp
int main() {
    Student s;
    manage m;

    int choice;
    while (true) {
        system("cls");
        cout<<"\n \t\t\t\t \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\
        cout<<"\n \t\t\t\t |-- STUDENT RECORD MANAGEMENT SYSTEM --|";
        cout<<"\n \t\t\t\t \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\

        cout<<"\t\t\t\t|  1. Admin             |\n";

        cout<<"\t\t\t\t|  2. Student           |\n";

        cout<<"\t\t\t\t|  3. Exit the program                        |\n";
        cout<<"\n \t\t\t\t \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\

        cout<<"\n\t\t\t\tEnter your choice  ";
        cin>>choice;

        switch (choice) {
        case 1:
            m.menu("admin");
            break;

        case 2:
            m.menu("student");
            break;

        case 3:
            exit(0);
            break;

        }
    }
return 0;
}
```

Output

```
    ═══════════════════════════════════
    |-- STUDENT RECORD MANAGEMENT SYSTEM --|
    ═══════════════════════════════════
   |  1. Admin             |
   |  2. Student           |
   |  3. Exit the program                        |

    ═══════════════════════════════════
   Enter your choice  _
```

Fig.2: Admin Menu

```
void adminMenu()
{
    int choice;
    while (true) {
        system("cls");
        cout<<"\n  \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\
        cout<<"\n  |-- STUDENT RECORD MANAGEMENT SYSTEM --|";
        cout<<"\n  \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\

        cout<<"1. Add new Student to the database\n";

        cout<<"2. Show All Student Information\n";

        cout<<"3. Search Student by Roll Number\n";

        cout<<"4. Search Student by First Name\n";

        cout<<"5. Search Student by Year \n";

        cout<<"6. Count Total number of Students\n";

        cout<<"7. Delete the Student by Roll Number\n";

        cout<<"8. Update Student Details by Roll Number\n";

        cout<<"9. Exit the program\n";

        cout<<"Enter your choice\n";
        cin>>choice;

 switch (choice) {
 case 1:
     addStudent();
     break;

 case 2:
     showAllStudents();
     break;

 case 3:
     searchStudentById();
     break;

 case 4:
     searchStudentByFirstName();
     break;

 case 5:
     searchStudentByYear();
     break;

 case 6:
     findTotalStudents();
     break;

 case 7:
     deleteStudent();
     break;
```

```
class manage : public Student
{
    public:
void registerUser(string username, string password,string role) {
    User newUser;
    newUser.username = username;
    newUser.password = password;
    newUser.role = role;

    ofstream outfile;
    outfile.open("users.txt", ios::app);
    outfile << newUser.username << "," << newUser.password <<","<<newUser.role<< endl;
    outfile.close();

    cout << "Registration successful!" << endl;
}
```

```
bool login(string username, string password,string role) {
    User user;
    ifstream infile;
    infile.open("users.txt");

    while (infile >> user.username ) {
        if (user.username == username+"," +password +","+role) {
            cout << "Login successful!" << endl;
            return true;
        }
    }

    cout << "Incorrect username or password." << endl;
    return false;
}
```

Output

```
1. Register
2. Login
3. Exit
Enter your choice:
```

```
1. Register
2. Login
3. Exit
Enter your choice: 1
Enter a username: 1212
Enter a password: 1212
Registration successful!
1. Register
2. Login
3. Exit
Enter your choice:
```

```
|-- STUDENT RECORD MANAGEMENT SYSTEM --|
1. Add new Student to the database
2. Show All Student Information
3. Search Student by Roll Number
4. Search Student by First Name
5. Search Student by Year
6. Count Total number of Students
7. Delete the Student by Roll Number
8. Update Student Details by Roll Number
9. Exit the program
Enter your choice
```

Fig.3: Student Menu

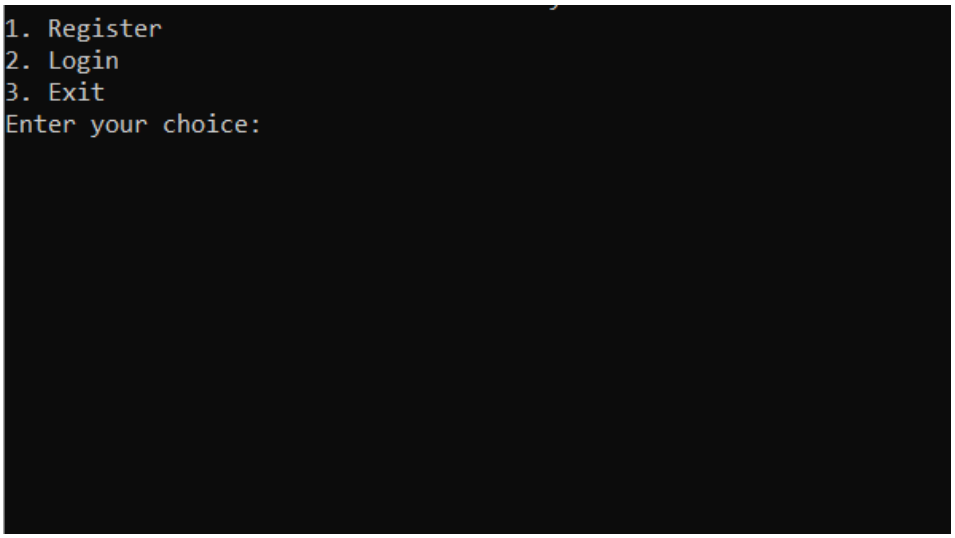
```
void studentMenu(char rollNo[])
{

    int choice;
    while (true) {
        system("cls");
        cout<<"\n \t\t\t\t \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\
        cout<<"\n \t\t\t\t |-- STUDENT RECORD MANAGEMENT SYSTEM --|";
        cout<<"\n \t\t\t\t \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\

        cout<<"\t\t\t\t|  1. Add Details                       |\n";

        cout<<"\t\t\t\t|  2. Update Details                    |\n";

        cout<<"\t\t\t\t|  3. show list                         |\n";

        cout<<"\t\t\t\t|  4. Exit the program                  |\n";
        cout<<"\n \t\t\t\t \xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\xcd\

        cout<<"\n\t\t\t\tEnter your choice  ";
        cin>>choice;
```

```
        switch (choice) {
        case 1:
            addStudent();
            break;

        case 2:
            updateStudent();
            break;

        case 3:
            showAllStudents();
            break;

        case 4:
            return;
            break;
        }
    }
```

```
class manage : public Student
{
    public:
void registerUser(string username, string password,string role) {
    User newUser;
    newUser.username = username;
    newUser.password = password;
    newUser.role = role;

    ofstream outfile;
    outfile.open("users.txt", ios::app);
    outfile << newUser.username << "," << newUser.password <<","<<newUser.role<< endl;
    outfile.close();

    cout << "Registration successful!" << endl;
}
```

```
bool login(string username, string password,string role) {
    User user;
    ifstream infile;
    infile.open("users.txt");

    while (infile >> user.username ) {
        if (user.username == username+"," +password +","+role) {
            cout << "Login successful!" << endl;
            return true;
        }
    }

    cout << "Incorrect username or password." << endl;
    return false;
}
```

Output



```
1. Register
2. Login
3. Exit
Enter your choice: 1
Enter a username: 1212
Enter a password: 1212
Registration successful!
1. Register
2. Login
3. Exit
Enter your choice:
```

```
 ═══════════════════════════════════════
 |-- STUDENT RECORD MANAGEMENT SYSTEM --|
 ═══════════════════════════════════════
|  1. Add Details                       |
|  2. Update Details                    |
|  3. show list                         |
|  4. Exit the program                  |

 ═══════════════════════════════════════
Enter your choice
```

Fig.4: Adding student data

```
temp = new node;
temp->first_name = firstName;
temp->last_name = lastName;
temp->roll_number =rollNo;
temp->branch = branch;
temp->year = year;
temp->next=NULL;
if(head==NULL)
{
    head=temp;
    head->next=NULL;
}
else
{
    last=head;
    while(last->next!=NULL)
    {
        last=last->next;
    }
    last->next=temp;
}
cout<<"\n\t\t\t\t\t_____DATA ADDED_____\n";
 cout<<"\t\t\t\t\t_____PRESS ANY KEY_____\n";
 getch();
 number_of_students++;
```

Output

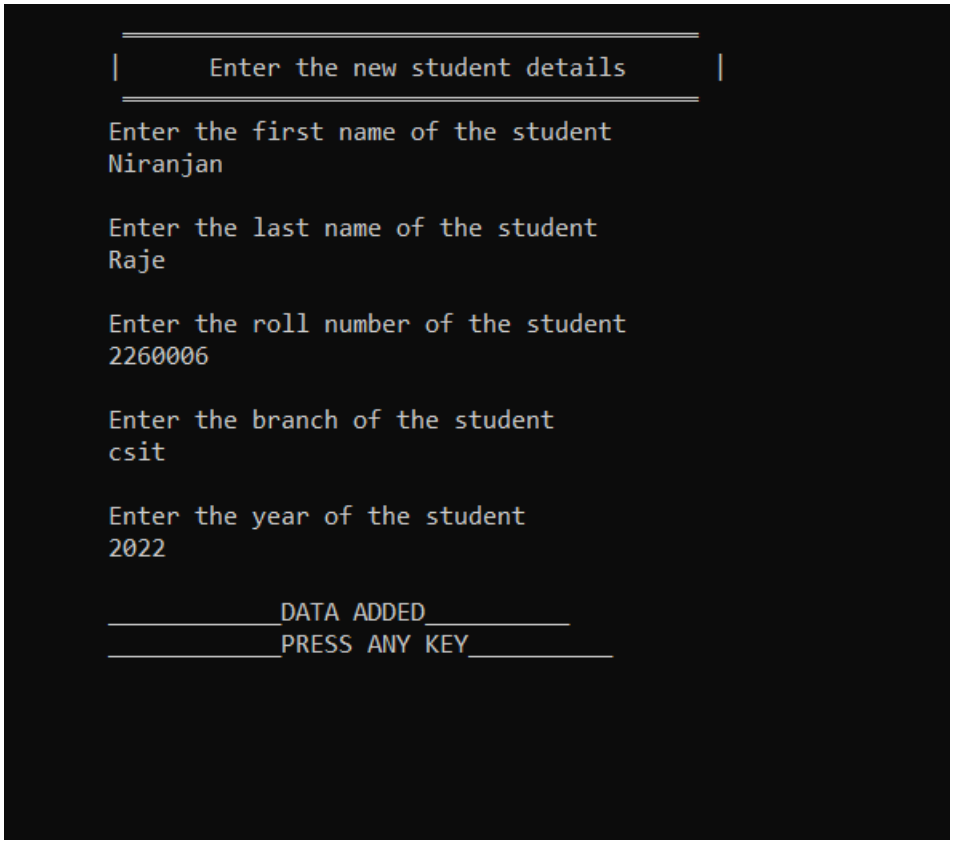


Fig.5: Displaying Student records

```cpp
void showAllStudents() {
    struct node* temp = head;
     system("cls");
     printLabel();
     cout<<"\n\t\t\tROLL NO \t FIRST NAME \t LAST NAME \t YEAR  \t BRANCH";

     while (temp!= NULL) {

             cout<<"\n\t\t\t"<<temp->roll_number<<"\t\t"<<temp->first_name<<"\t\t"<<temp->last_name<<"\t
         temp = temp->next;
     }
     cout<<"\n\t\t\t\t____________PRESS ANY KEY__________\n";
     getch();
}
```

Output

```
        |       Enter the new student details       |

    ROLL NO         FIRST NAME      LAST NAME       YEAR    BRANCH
    2260006         Niranjan                Raje            2022            scsit
    2260008         Pranav          shinde          2022            scsit
    2260009         Sarang          Mane            2022            scsit
    2260001         Shubham         Mane            2022            scsit
            ____________PRESS ANY KEY__________
```

Fig.6: Updating student record

```cpp
void updateStudent() {
    //int rollNo;
    char rollNo[10];
    system("cls");
    cout << "\t\t\t\tEnter roll no of student to update:\n\t\t\t\t ";
    cin >> rollNo;

    node* temp = head;
    while (temp != NULL) {
        if (temp->roll_number == rollNo) {
            int error = 0;
           do
           {
            int error = 0;
             cout<<"\n\t\t\t\tEnter new first name of the student\n\t\t\t\t";
             cin >> firstName;
             for (int i = 0; i < strlen(firstName); i++)
             {
                if(isdigit(firstName[i]))
                {
                    system("cls");
                    cout << "\n\t\t\t\tPlease enter a valid first name which does not contain any number" << endl;
                    error = 1;
                    break;
                }
             }
           }while(error == 1);
```

```
        temp->first_name = firstName;
        temp->last_name = lastName;
        temp->roll_number =rollNo;
        temp->branch = branch;
        temp->year = year;
            return;
        }
        temp = temp->next;
    }
    cout << "\n\t\t\t\tStudent not found" << endl;
}
```

Output

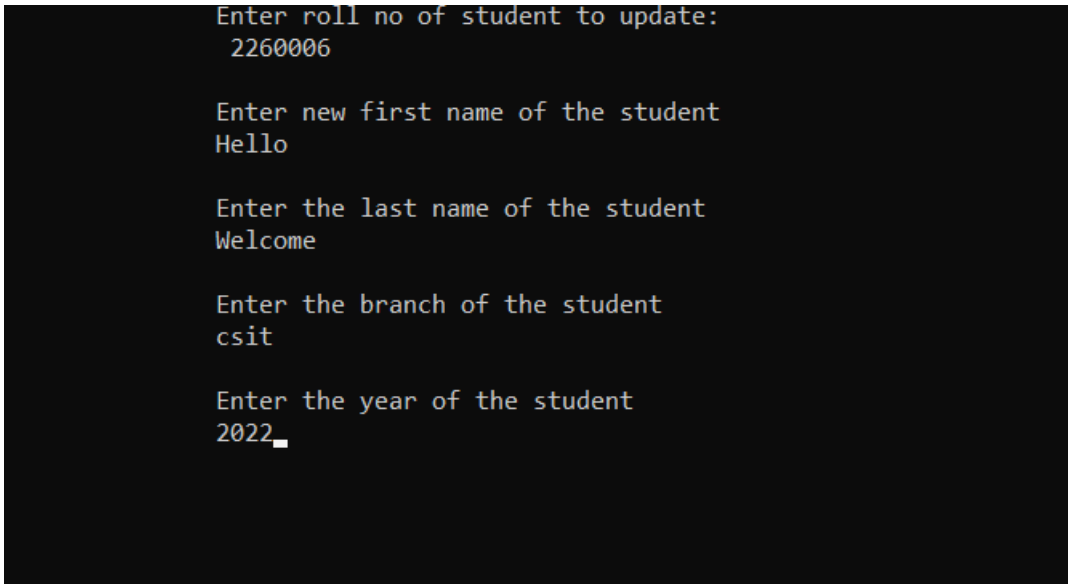


Fig.7: Deleting student record

```
void deleteStudent() {
    //int rollNo;
    //string rollNo;
    system("cls");
    cout << "\t\t\t\tEnter roll no of student to delete:\n\t\t\t\t ";
    cin >> rollNo;


    node* temp = head;
    node* prev = NULL;
    while (temp != NULL) {
        if (temp->roll_number == rollNo) {
            cout<<"deleted";
            if (prev == NULL) {
                head = temp->next;
            } else {
                prev->next = temp->next;
            }
            delete temp;
            number_of_students--;
            getch();
            return;
        }
        prev = temp;
        temp = temp->next;
    }
    cout << "\n\t\t\t\tStudent not found" << endl;
}
```

Output

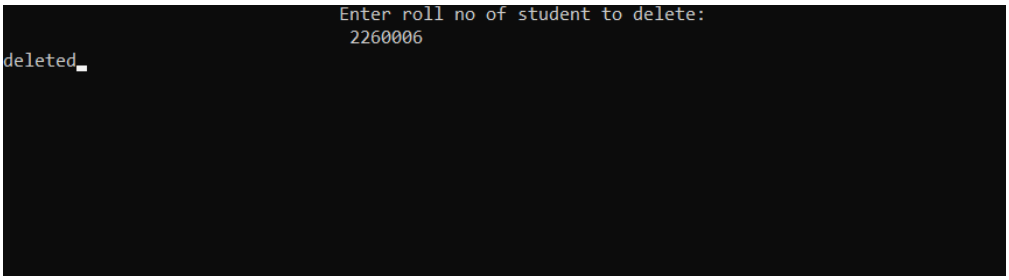


Fig.8: Searching student record by roll no

```
void searchStudentById() {
    //int rollNo;
    string rollNo;
    system("cls");
    cout << "\t\t\t\tEnter roll no of student to search:\n\t\t\t\t ";
    cin >> rollNo;

    node* temp = head;
    while (temp != NULL) {
        if (temp->roll_number == rollNo) {
            cout<<"\n"<<temp->roll_number<<"\t"<<temp->first_name<<temp->last_name<<"\t"<<temp->year<<"\t"<<temp->branch;
            cout<<"\n\t\t\t\t\t__________PRESS ANY KEY__________\n";
            getch();
            return;
        }
        temp = temp->next;
    }
    cout << "\n \t\t\t\tStudent not found" << endl;
    cout<<"\n\t\t\t\t\t__________PRESS ANY KEY__________\n";
    getch();
}
```

Output

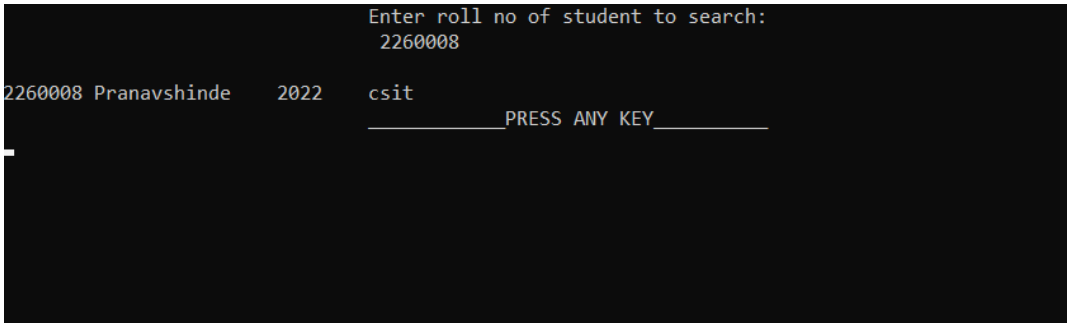


Fig.9: Searching student record by name

```
void searchStudentByFirstName() {
     string firstName;
     system("cls");
    cout << "\t\t\t\tEnter name of student to search:\n\t\t\t\t\t ";
    cin >>firstName;
    node* temp = head;
    while (temp != NULL) {
        if (temp->first_name == firstName) {
            cout<<"\n"<<temp->roll_number<<"\t"<<temp->first_name<<temp->last_name<<"\t"<<temp->year<<"\t"<<temp->branch;
            cout<<"\n\t\t\t\t\t_____________PRESS ANY KEY_____________\n";
            getch();
            return;
        }
        temp = temp->next;
    }
    cout << "\n\t\t\t\t\tStudent not found" << endl;
    cout<<"\n\t\t\t\t\t_____________PRESS ANY KEY_____________\n";
    getch();
}
```

Output

```
                              Enter name of student to search:
                               Pranav

2260008 Pranavshinde    2022    csit
                              _____________PRESS ANY KEY_____________
```

Fig.10: Searching student record by year

```
void searchStudentByYear() {
     //int year;
     string year;
     system("cls");
    cout << "\t\t\t\tEnter year of student to search:\n\t\t\t\t\t ";
    cin >> year;
    node* temp = head;
    while (temp != NULL) {
        if (temp->year == year) {
            cout<<"\n"<<temp->roll_number<<"\t"<<temp->first_name<<temp->last_name<<"\t"<<temp->year<<"\t"<<temp->branch;
            cout<<"\n\t\t\t\t\t_____________PRESS ANY KEY_____________\n";
            getch();
            return;
        }
        temp = temp->next;
    }
    cout << "\t\t\t\t\tStudent not found" << endl;
    cout<<"\n\t\t\t\t\t_____________PRESS ANY KEY_____________\n";
    getch();
}
```

Output

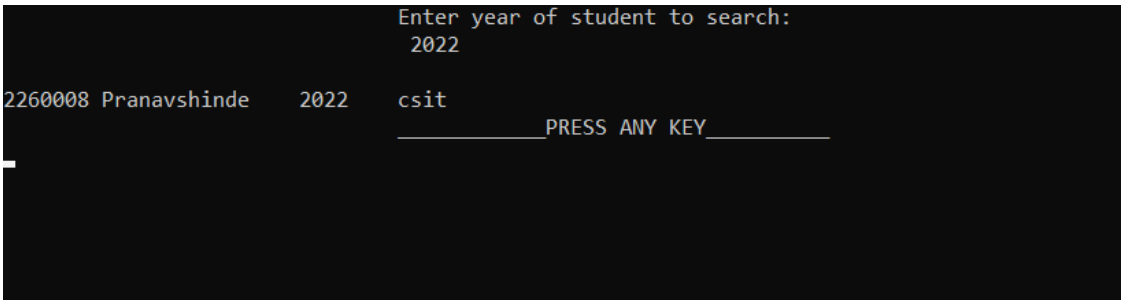
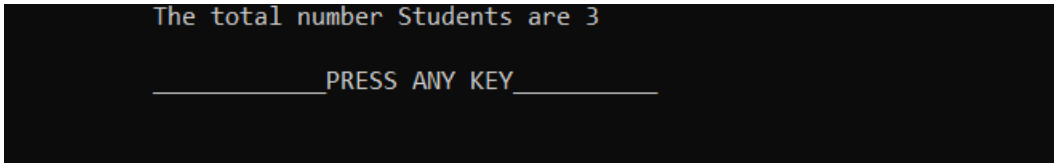


Fig.11: Displaying Total Number of Student

```
void findTotalStudents()
{
  system("cls");
  cout<<"\t\t\t\tThe total number Students are "<< number_of_students <<"\n";
  cout<<"\n\t\t\t\t\t_____PRESS ANY KEY_____\n";
  getch();
}
```

Output



Signature of Supervisor/s:_____