LAKSHMI NARAIN COLLEGE OF TECHNOLOGY, BHOPAL

Master of Computer Applications

LAB Assignment

MCA – 307 LAB IN C++

SUBMITTED TO---

DR. SANJAY VAJPAYEE

SUBMITTED BY---

NAME: ABHISHEK GUPTA
ENROLL NO: 0103CA191003

Q1: Write a program that defines an inline function containing two arguments first is for salary of an employee and second is the percent value by which salary is to be increased. Main program passes these two values to the function and receives increased salary for printing.

Program

```
1. #include <iostream>
2.
3. using namespace std;
4.
5. class emp{
6. float sal ,per;
7.
8. public:
9.     int cal(float,float);
10.  };
11.   inline int emp::cal(float _sal , float _per)
12.      {
13.
14.          sal=_sal;
15.          per=_per;
16.          return (sal+(sal*per/100));
17.      }
18.  int main()
19.  {
20.      emp e1;
21.      float salary,percent;
22.      cout<<"Enter the EMP salary :";
23.      cin>>salary;
24.      cout<<"inc. salary in per :";
25.      cin>>percent;
26.      cout<<"\nincreased salary ="<<e1.cal(salary,percent);
27.      return 0;
28.  }
```

Output

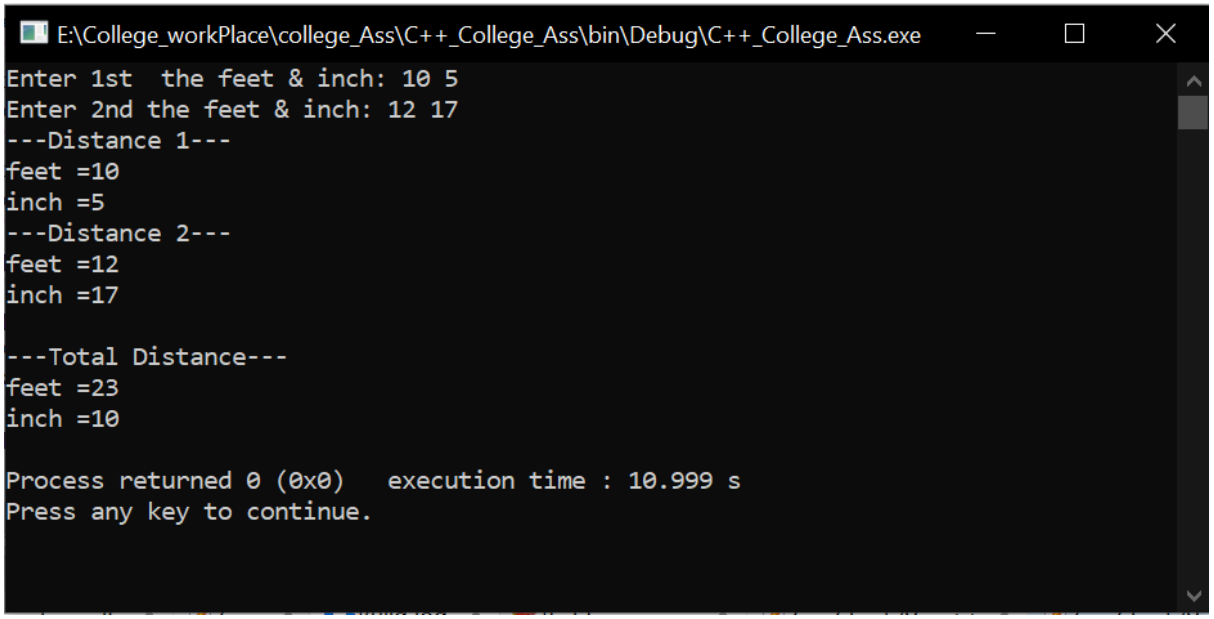
```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter the EMP salary :3050
inc. salary in per :
4.5

increased salary =3187
Process returned 0 (0x0)   execution time : 8.197 s
Press any key to continue.
```

Q2: Write a program in C + + that includes a class distance, with data members feet and inch. It should include a constructor with parameters, member function to display the distance and a friend function add ( ) that adds distances of two objects.

Program

```
1. #include <iostream>
2.  using namespace std;
3.
4. class Distance{
5.     int feet;
6.     int inch;
7. public:
8.     Distance(int _feet=0 ,int _inch=0):feet(_feet),inch(_inch){
9.     }
10.
11.      void display();
12.     friend void add_Distance(const Distance& , const Distance&
   ,Distance& );
13.  };
14.
15.  void Distance::display(){
16.      cout<<"feet ="<<feet<<endl;
17.      cout<<"inch ="<<inch<<endl;
18.  }
19.
20.  void add_Distance(const Distance& p ,const Distance& q ,Distance&
   temp){
21.      temp.feet=p.feet+q.feet;
22.      temp.inch=p.inch+q.inch;
23.
24.      if(temp.inch>=12){
25.          temp.feet=temp.feet+temp.inch/12;
26.          temp.inch=temp.inch%12;
27.      }
28.  }
29.
30.  int main(){
31.      int _feet,_inch;
32.      cout<<"Enter 1st  the feet & inch: ";
33.      cin>>_feet>>_inch;
34.      Distance D1(_feet,_inch);
35.      cout<<"Enter 2nd the feet & inch: ";
36.      cin>>_feet>>_inch;
37.      Distance D2(_feet,_inch);
38.      cout<<"---Distance 1---"<<endl;
39.      D1.display();
40.      cout<<"---Distance 2---"<<endl;
41.      D2.display();
42.      Distance D3;
43.      add_Distance(D1,D2,D3);
44.      cout<<"\n---Total Distance---"<<endl;
45.      D3.display();
46.
47.  return 0;
48.  }
```

Output


E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter 1st the feet & inch: 10 5
Enter 2nd the feet & inch: 12 17
---Distance 1---
feet =10
inch =5
---Distance 2---
feet =12
inch =17
---Total Distance---
feet =23
inch =10
Process returned 0 (0x0)   execution time : 10.999 s
Press any key to continue.

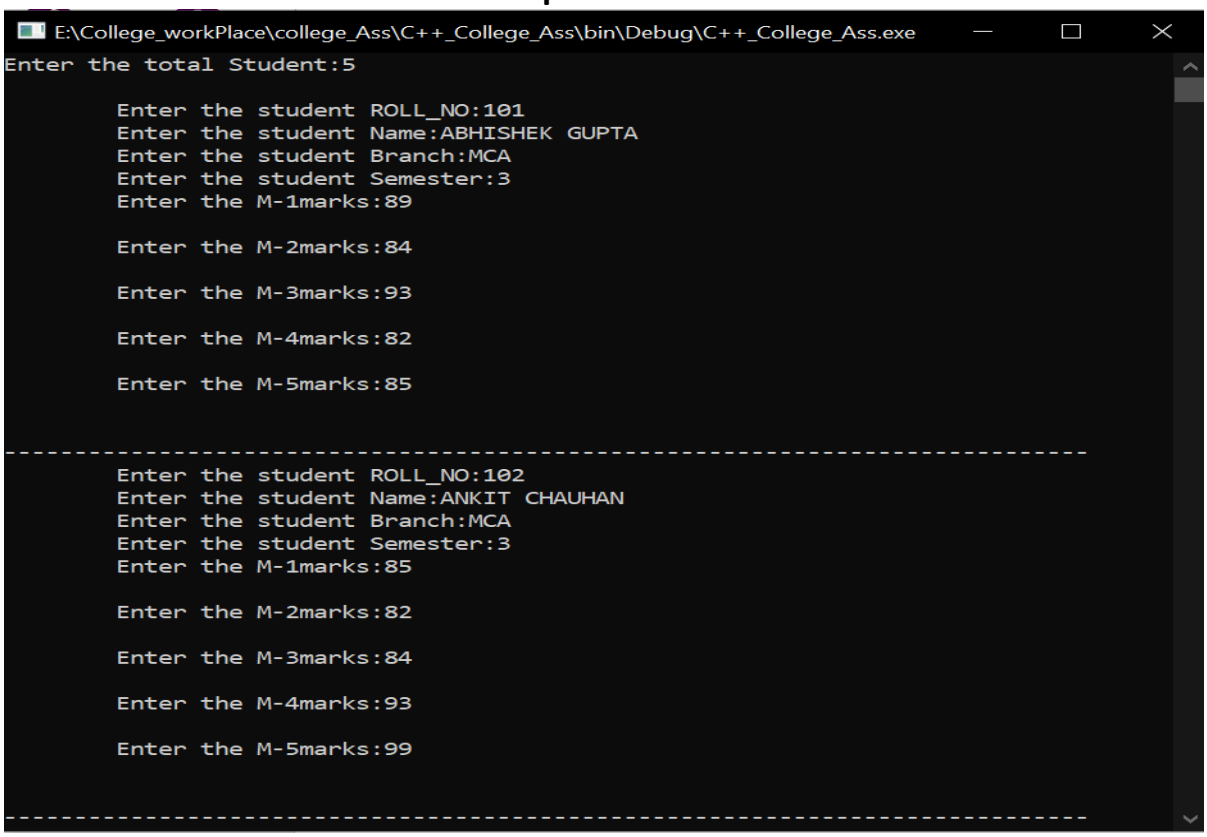
Q3: Write a program to read records of students with attributes Roll_No, Name, Branch, Semester and 5 theory marks in M1, M2, M3, M4 and M5. Calculate the percentage of each student and store in its record. Use array of objects to store data and display them record-wise.

Program

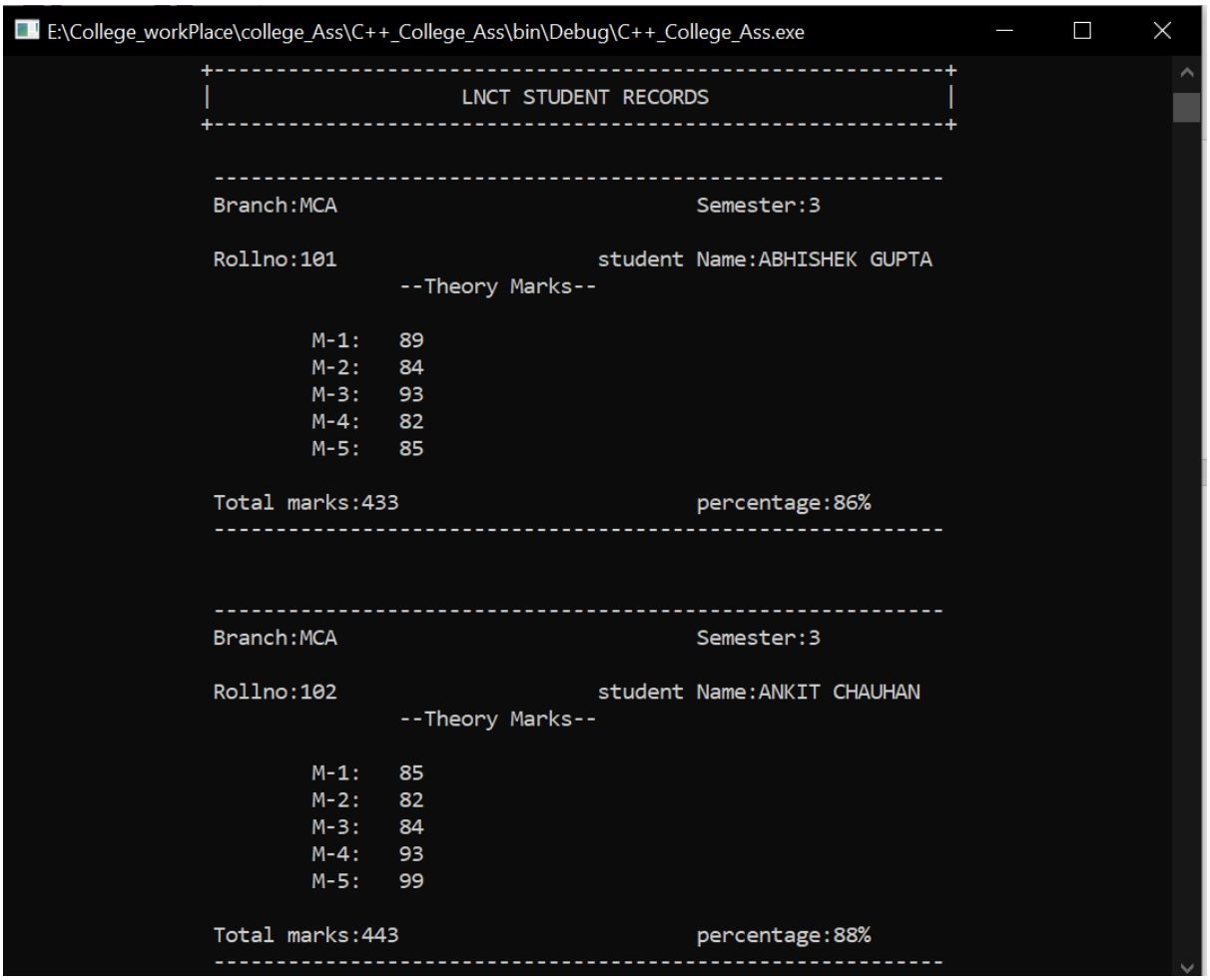
```
1. #include<iostream>
2. #include<cstring>
3. using namespace std;
4.
5. class student
6. {
7.  private:
8.    int Roll_No;
9.    char *Name;
10.   char *Branch;
11.   int Semester;
12.   int  theory_marks[5]={0};
13.   int total;
14.   float per;
15.   public:
16.
17.       void get();
18.       void show();
19.          ~student(){
20.           delete []Name;
21.           delete []Branch;
22.       }
23.  };
24.
25.  void student::get(){
26.      cout<<"\n\tEnter the student ROLL_NO:";
27.      cin>>Roll_No;
28.      char temp[30];
29.      cout<<"\tEnter the student Name:";
30.      {
31.      cin.ignore();
32.      cin.getline(temp,30);
33.      int x=strlen(temp);
34.      Name=new char[x+1];
35.      strcpy(Name,temp);
36.      }
37.      cout<<"\tEnter the student Branch:";
38.      {
39.      cin.getline(temp,30);
40.      int x=strlen(temp);
41.      Branch=new char[x+1];
42.      strcpy(Branch,temp);
43.      }
44.      cout<<"\tEnter the student Semester:";
45.      cin>>Semester;
46.      int temp_sum=0;
47.      for(int i=0;i<5;i++){
48.          cout<<"\tEnter the M-"<<i+1<<"marks:";
49.          cin>>theory_marks[i];
50.          cout<<"\n";
```

```
51.          temp_sum=temp_sum+theory_marks[i];
52.      }
53.      total=temp_sum;
54.      per=total/5;
55.  }
56.  void student::show(){
57.
58.      cout<<"\n\t\t ----------------------------------------------------
   ----------\n";
59.      cout<<"\t\t
   Branch:"<<Branch<<"\t\t\t\t"<<"Semester:"<<Semester;
60.      cout<<"\n\n\t\t Rollno:"<<Roll_No<<"\t\t\t"<<"student
   Name:"<<Name;
61.      cout<<"\n\t\t \t\t--Theory Marks--\n";
62.      for(int i=0;i<5;i++){
63.      cout<<"\n\t\t\t M"<<"-"<<i+1<<":    "<<theory_marks[i];
64.  }
65.      cout<<"\n\n\t\t Total
   marks:"<<total<<"\t\t\t"<<"percentage:"<<per<<"%";
66.      cout<<"\n\t\t ----------------------------------------------------
   ----------\n";
67.  }
68.
69.  int main(){
70.      int n;
71.      student *ptr;
72.      cout<<"Enter the total Student:";
73.      cin>>n;
74.      ptr=new student[n];
75.      if(ptr==0){
76.          cout<<"insuff. memory !";
77.          return 1;
78.      }
79.      for(int i=0;i<n;i++){
80.          (ptr+i)->get();
81.          cout<<"\n----------------------------------------------------
   ---------------------------";
82.      }
83.      cout<<"\n\t\t+----------------------------------------------------
   ----------+\n";
84.      cout<<"\t\t|                      LNCT STUDENT RECORDS
   |\n";
85.      cout<<"\t\t+----------------------------------------------------
   --------+\n";
86.      for(int i=0;i<n;i++){
87.          (ptr+i)->show();
88.          cout<<"\n";
89.      }
90.      delete []ptr;
91.  return 0;
92.  }
```

Output



```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter the total Student:5

        Enter the student ROLL_NO:101
        Enter the student Name:ABHISHEK GUPTA
        Enter the student Branch:MCA
        Enter the student Semester:3
        Enter the M-1marks:89

        Enter the M-2marks:84

        Enter the M-3marks:93

        Enter the M-4marks:82

        Enter the M-5marks:85


--------------------------------------------------------------------------------
        Enter the student ROLL_NO:102
        Enter the student Name:ANKIT CHAUHAN
        Enter the student Branch:MCA
        Enter the student Semester:3
        Enter the M-1marks:85

        Enter the M-2marks:82

        Enter the M-3marks:84

        Enter the M-4marks:93

        Enter the M-5marks:99


--------------------------------------------------------------------------------
```



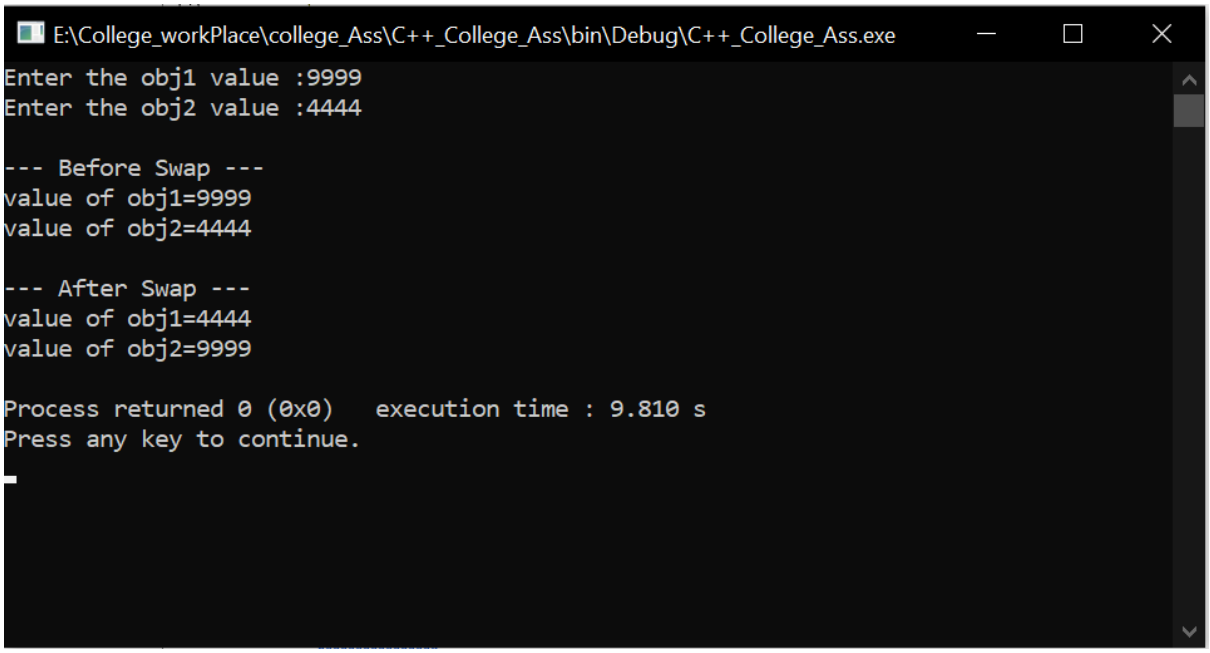
```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
              +-----------------------------------------------------------+
              |                    LNCT STUDENT RECORDS                   |
              +-----------------------------------------------------------+

               ------------------------------------------------------------
               Branch:MCA                              Semester:3

               Rollno:101                    student Name:ABHISHEK GUPTA
                             --Theory Marks--

                      M-1:   89
                      M-2:   84
                      M-3:   93
                      M-4:   82
                      M-5:   85

               Total marks:433                         percentage:86%
               ------------------------------------------------------------


               ------------------------------------------------------------
               Branch:MCA                              Semester:3

               Rollno:102                    student Name:ANKIT CHAUHAN
                             --Theory Marks--

                      M-1:   85
                      M-2:   82
                      M-3:   84
                      M-4:   93
                      M-5:   99

               Total marks:443                         percentage:88%
               ------------------------------------------------------------
```

Q4: Write a program to swap private data of two classes.

Program

```
1. #include <iostream>
2. using namespace std;
3.
4. class B;
5.
6. class A{
7. private:
8.     int x;
9. public:
10.      A(int _x ):x(_x){
11.
12.      }
13.      void show(){
14.      cout<<"value of obj1="<<x<<endl;
15.      }
16.      friend  void data_swap(A& , B&);
17.  };
18.
19.  class B{
20.  private:
21.      int y;
22.  public:
23.      B(int _y ):y(_y){
24.      }
25.      void show(){
26.      cout<<"value of obj2="<<y<<endl;
27.      }
28.      friend  void data_swap(A& , B&);
29.  };
30.
31.  void data_swap(A& O1 , B& O2){
32.      int temp;
33.      temp=O1.x;
34.      O1.x=O2.y;
35.      O2.y=temp;
36.  }
37.
38.  int main(){
39.      int z;
40.      cout<<"Enter the obj1 value :";
41.      cin>>z;
42.      A obj1(z);
43.      cout<<"Enter the obj2 value :";
44.      cin>>z;
45.      B obj2(z);
46.      cout<<"\n--- Before Swap ---"<<endl;
47.      obj1.show();
48.      obj2.show();
49.
50.      data_swap(obj1,obj2);
51.      cout<<"\n--- After Swap ---"<<endl;
52.      obj1.show();
53.      obj2.show();
54.      return 0;
55.  }
```

Output


E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter the obj1 value :9999
Enter the obj2 value :4444
--- Before Swap ---
value of obj1=9999
value of obj2=4444
--- After Swap ---
value of obj1=4444
value of obj2=9999
Process returned 0 (0x0)   execution time : 9.810 s
Press any key to continue.

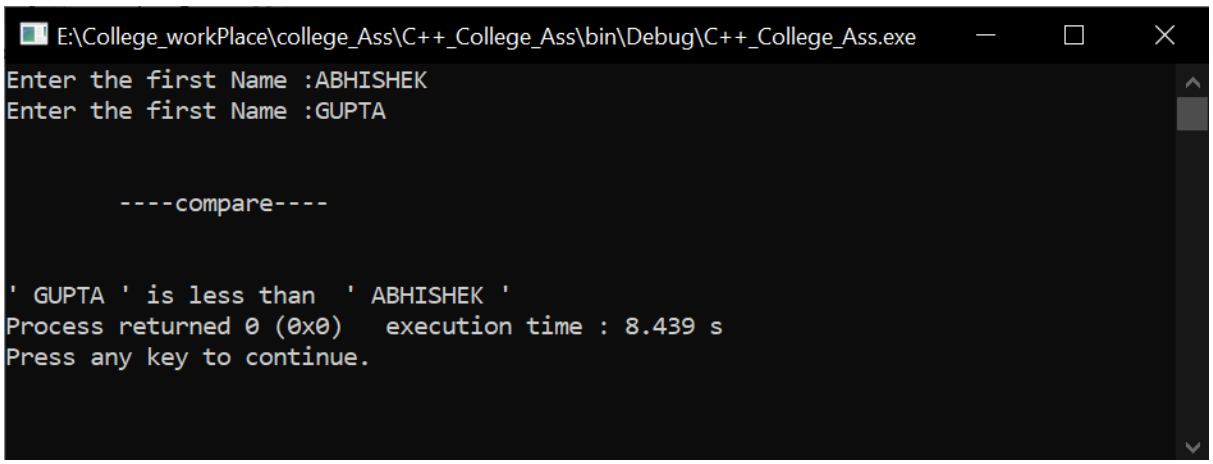
Q5: Write a program to read a string with the help of constructor using dynamic memory allocation. Then overload" < "operator to compare their length and display the smaller string first and larger string second. For e.g. If obj1 and obj2 are two objects having a string then compare by if(obj1<obj2). If condition is true then print string of object obj1 "is less than" followed by string of object obj2 otherwise vice versa.

Program

```
1. #include<iostream>
2. using namespace std;
3.
4. char* name_copy(char*,const char*);
5. int name_length(const char*);
6.
7. class Name{
8.     char* name=NULL;
9. public:
10.      Name();
11.      bool operator<(const Name&);
12.      void show(){
13.      cout<<name<<" ";
14.      }
15.      ~Name(){
16.      delete[]name;
17.      }
18.  };
19.
20.  Name::Name(){
21.      char temp[30];
22.      cout<<"Enter the first Name :";
23.      cin.getline(temp,30);
24.      int x=name_length(temp);
25.      name=new char[x+1];
26.      name_copy(name,temp);
27.  }
28.
29.  bool Name::operator<(const Name& _obj){
30.      int l1,l2;
31.      l1=name_length(name);
32.      l2=name_length(_obj.name);
33.      if(l1 < l2)
34.          return true;
35.      else
36.          return false;
37.  }
38.
39.  char* name_copy(char* str1, const char* str2){
40.      int i;
41.      for(i=0;str2[i]!='\0';i++){
42.          str1[i]=str2[i];
43.      }
44.      str1[i]='\0';
45.      return str1;
46.  }
47.
```

```
48.  int name_length(const char* ptr){
49.      int i;
50.      for(i=0;*(ptr+i)!='\0';i++);
51.      return i;
52.  }
53.
54.
55.  int main(){
56.      Name str1;
57.      Name str2;
58.
59.      cout<<"\n\n\t----compare---- \n\n"<<endl;
60.      if(str1 < str2){
61.          cout<<"' ";
62.          str1.show();
63.          cout<<"'";
64.          cout<<" "<<"is less than "<<" ";
65.          cout<<"' ";
66.          str2.show();
67.          cout<<"' ";
68.          }
69.      else if(str2 < str1){
70.          cout<<"' ";
71.          str2.show();
72.          cout<<"'";cout<<" "<<"is less than "<<" ";
73.           cout<<"' ";
74.          str1.show();
75.          cout<<"'";
76.          }
77.      else{
78.           cout<<"' ";
79.          str1.show();
80.          cout<<"'";cout<<" equal "<<" ";
81.           cout<<"' ";
82.          str2.show();
83.          cout<<"'";
84.          }
85.  return 0;
86.  }
```

Output



Q6 Create 2-D matrices dynamically using class and multiply them

Program

```
1. #include<iostream>
2. using namespace std;
3. class Matrix{
4.     int **ptr_arr;
5.     int size_arr;
6. public:
7.     Matrix(int);
8.     void get();
9.     void display();
10.      void multiply(const Matrix&,const Matrix&);
11.      ~Matrix();
12.  };
13.
14.  Matrix::Matrix(int _n):size_arr(_n){
15.      ptr_arr=new int*[size_arr];
16.      for(int i = 0; i < size_arr; i++)
17.      {
18.          ptr_arr[i] = new int[size_arr];
19.      }
20.  }
21.  void Matrix::get(){
22.    for(int i=0;i<size_arr;i++)
23.      {
24.          for(int j=0;j<size_arr;j++){
25.            //  cout<<"enter the value a["<<i+1<<"]""["<<j+1<<"]:";
26.              cin>>ptr_arr[i][j];
27.      }
28.    }
29.
30.  }
31.  void Matrix::display(){
32.      for(int i=0;i<size_arr;i++){
33.              for(int j=0;j<size_arr;j++){
34.                  cout<<"\t\t"<<ptr_arr[i][j]<<"   ";
35.      }
36.      cout<<"\n";
37.    }
38.  }
39.
40.  void Matrix::multiply(const Matrix& obj1 ,const Matrix& obj2){
41.      int sum;
42.      for(int i=0;i<size_arr;i++){
43.          for(int j=0;j<size_arr;j++){
44.              sum=0;
45.              for(int k=0;k<size_arr;k++)
46.                  sum=sum+obj1.ptr_arr[i][k]*obj2.ptr_arr[k][j];
47.              this->ptr_arr[i][j]=sum;
48.              }
49.      }
50.  }
51.
52.
53.
```

```
54.  Matrix::~Matrix(){
55.      for(int i = 0; i < size_arr; i++)
56.      {
57.          delete [] ptr_arr[i];
58.      }
59.      delete [] ptr_arr;
60.  }
61.
62.  int main()
63.  {
64.      int n;
65.      cout<<"Enter the Size of sq matrix:";
66.      cin>>n;
67.      Matrix M1(n);
68.      Matrix M2(n);
69.      Matrix M3(n);
70.      cout<<"\n\t\t---Enter the 1st array element---\n\n";
71.      M1.get();
72.      cout<<"\n\t\t---Enter the 2nd array element---\n\n";
73.      M2.get();
74.
75.      cout<<"\n\t\t---1st array element---\n";
76.      M1.display();
77.      cout<<"\n\t\t---2nd array element---\n";
78.      M2.display();
79.      M3.multiply(M1,M2);
80.      cout<<"\n\t\t---Multiply 1st matrix and 2nd matrix---\n";
81.      M3.display();
82.      return 0;
83.  }
```

Output

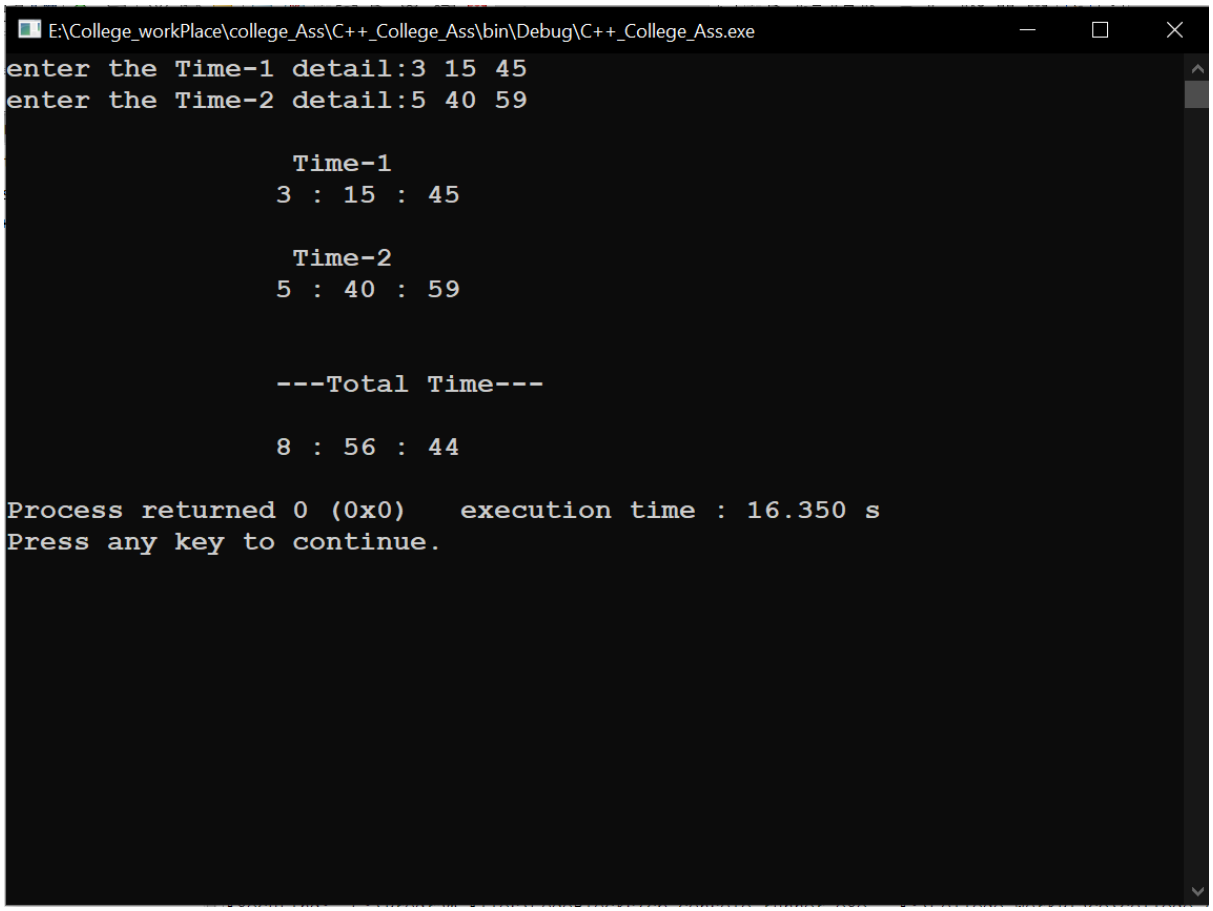
```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter the Size of sq matrix:3

                ---Enter the 1st array element---

                1 2 3
                4 5 6
                7 8 9

                ---Enter the 2nd array element---

                9 8 7
                6 5 4
                3 2 1

                ---1st array element---
                1               2               3
                4               5               6
                7               8               9

                ---2nd array element---
                9               8               7
                6               5               4
                3               2               1

                ---Multiply 1st matrix and 2nd matrix---
                30              24              18
                84              69              54
                138             114             90

Process returned 0 (0x0)   execution time : 67.140 s
Press any key to continue.
```

Q7 Declare a class Time having its members as hour, minute and second. Define functions to read and display the data and add the contents of two objects into third object like T3 = T1 + T2 where T1, T2 and T3 are objects of class Time.

Program

```
1. #include<iostream>
2. using namespace std;
3.
4. class _time{
5.     int hour,minute,second;
6. public:
7.     _time(int=0 ,int=0 ,int=0);
8.     void show__time();
9.     _time operator+(const _time&);
10. };
11. _time::_time(int h,int m,int s):hour(h),minute(m),second(s){
12. }
13. void _time::show__time(){
14.     cout<<"\t\t"<<hour<<" : "<<minute<<" : "<<second<<endl;
15. }
16. _time _time::operator+(const _time& _t1){
17.     _time temp;
18.     temp.second=second+_t1.second;
19.     temp.minute=minute+_t1.minute+(temp.second/60);
20.     temp.hour=hour+_t1.hour+(temp.minute/60);
21.     temp.second=temp.second%60;
22.     temp.minute=temp.minute%60;
23.     return temp;
24. }
25.
26. int main(){
27.     int hh,mm,ss;
28.     cout<<"enter the Time-1 detail:";
29.     cin>>hh>>mm>>ss;
30.     _time T1(hh,mm,ss);
31.     cout<<"enter the Time-2 detail:";
32.     cin>>hh>>mm>>ss;
33.     _time T2(hh,mm,ss);
34.     _time T3;
35.     T3=T1+T2;
36.     cout<<"\n\t\t Time-1\n";
37.     T1.show__time();
38.     cout<<"\n\t\t Time-2\n";
39.     T2.show__time();
40.     cout<<"\n\n\t\t---Total Time---\n\n";
41.     T3.show__time();
42. return 0;
43. }
```

Output

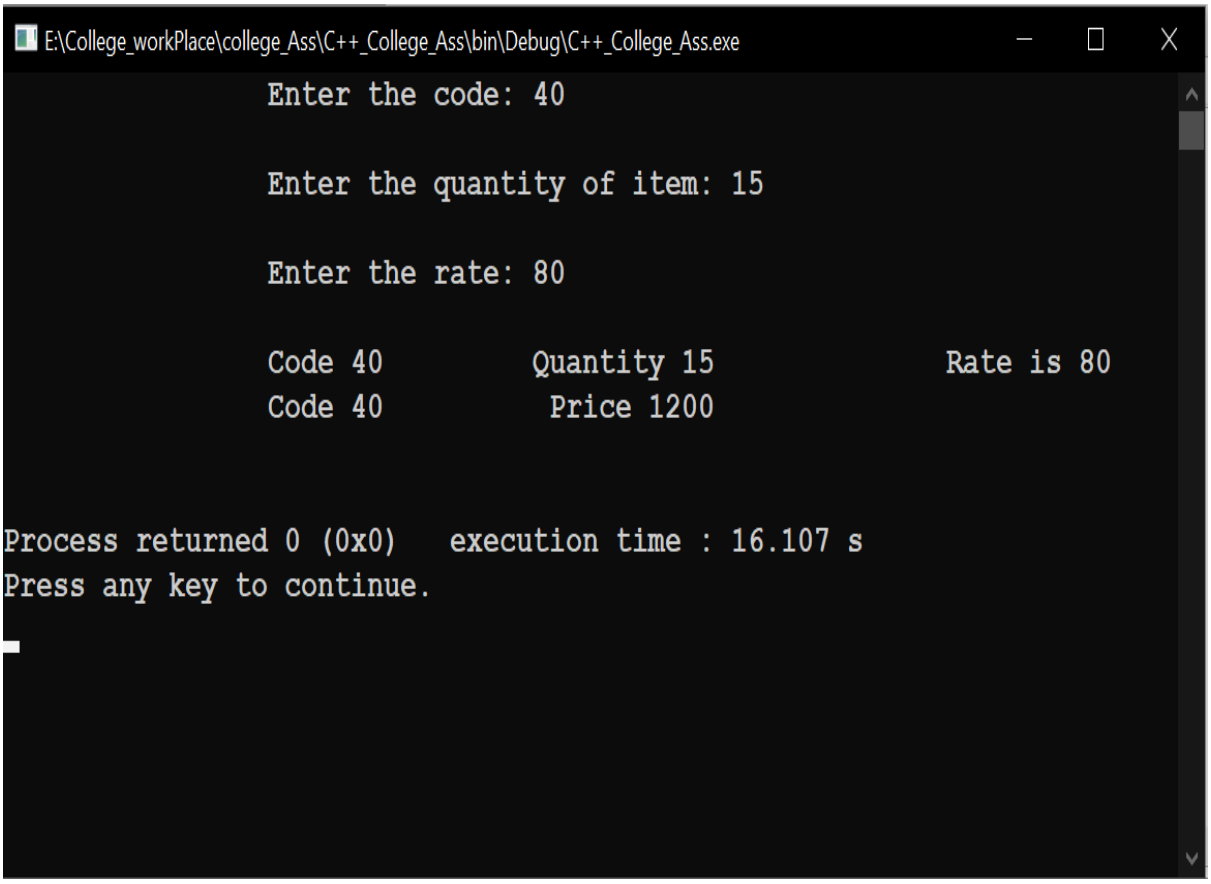

E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
enter the Time-1 detail:3 15 45
enter the Time-2 detail:5 40 59
Time-1
3 : 15 : 45
Time-2
5 : 40 : 59
---Total Time---
8 : 56 : 44
Process returned 0 (0x0) execution time : 16.350 s
Press any key to continue.

Q8 Let class items have code, qty and rate as data members which are read by the appropriate functions. Another class product is having code and price as its data members and its values are obtained by the object of class item like obj1 = obj2; where obj2 is the object of class item and obj1 is the object of class product.

Program

```
1. #include<iostream>
2. using namespace std;
3. class product;
4. class Items{
5.     int code;
6.     int qty;
7.     int rate;
8. public:
9.     Items(int c, int q ,int r):code(c),qty(q),rate(r){
10.      }
11.       void show()
12.        {
13.            cout<<"\n\t\tCode "<<code<<"\t\tQuantity "<<qty<<"\t\t
   Rate is "<<rate;
14.        }
15.        friend product;
16.  };
17.  class product{
18.     int code;
19.     int price;
20.  public:
21.     product(int c=0,int p=0):code(c),price(p){
22.     }
23.     void operator=(const Items &temp ){
24.          code = temp.code;
25.          price = temp.qty*temp.rate;
26.        }
27.        void show()
28.        {
29.            cout<<"\n\t\tCode "<<code<<"\t\t Price "<<price<<"\n\n";
30.        }
31.  };
32.  int main()
33.  {
34.       int _code,_qty,_rate;
35.       cout<<"\t\tEnter the code: ";
36.       cin>>_code;
37.       cout<<"\n\t\tEnter the quantity of item: ";
38.       cin>>_qty;
39.       cout<<"\n\t\tEnter the rate: ";
40.       cin>>_rate;
41.       Items obj_Item(_code,_qty,_rate);
42.       obj_Item.show();
43.       product obj_product;
44.       obj_product=obj_Item;
45.       obj_product.show();
46.       return 0;
47.  }
```

Output


E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter the code: 40
Enter the quantity of item: 15
Enter the rate: 80
Code 40 Quantity 15 Rate is 80
Code 40 Price 1200
Process returned 0 (0x0) execution time : 16.107 s
Press any key to continue.

Q9 Overload insertion and extraction operator in your program and show the results.

Program

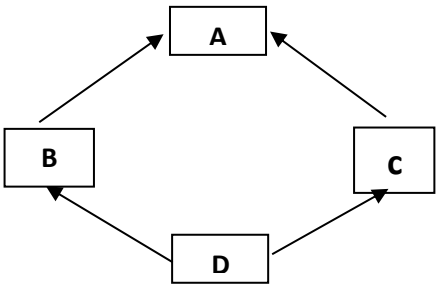
```
1. #include<iostream>
2. using namespace std;
3.
4. class Emp{
5.     int age;
6.     char name[20];
7.     float sal;
8. public:
9.     friend istream& operator>>(istream& ,Emp& );// insertion operator
10.      friend ostream& operator<<(ostream& ,Emp& );//extraction
11.  };
12.
13.  istream& operator>>(istream& in, Emp& E){
14.      in>>E.age>>E.name>>E.sal;
15.      return in;
16.  }
17.
18.  ostream& operator<<(ostream& out, Emp& E){
19.      out<<E.age<<"  "<<E.name<<"  "<<E.sal<<endl;
20.      return out;
21.  }
22.
23.  int main(){
24.      Emp E1,E2;
25.      cout<<"\n\n\t\tEnter the age ,name and sal:";
26.      cin>>E1;
27.      cout<<"\n\n\t\tEnter the age ,name and sal:";
28.      cin>>E2;
29.      cout<<"\n\n\t\t--details of 1st Emp--\n\t\t";
30.      cout<<E1;
31.      cout<<"\n\n\t\t--details of 2nd Emp--\n\t\t";
32.      cout<<E2;
33.  return 0;
34.  }
```

Output

```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe

                Enter the age ,name and sal: 22 Abhishek 75000


                Enter the age ,name and sal: 23 Amit 68000


                --details of 1st Emp--
                22  Abhishek  75000


                --details of 2nd Emp--
                23  Amit  68000

Process returned 0 (0x0)   execution time : 28.944 s
Press any key to continue.
```

Q10 Write a program to illustrate the concept of hybrid inheritance.
Ans:

Hybrid Inheritance or Multipath Inheritance



Program

```
1. #include<iostream>
2. using namespace std;
3. class Base
4. {
5.   public:
6.           int a;
7. };
8. class drv1 : virtual public Base
9. {
10.          public:
11.                          int b;
12.  };
13.  class drv2: virtual public Base
14.  {
15.          public:
16.                  int c;
17.  };
18.  class drv3 : public drv1, public drv2
19.  {
20.          public:
21.                  int d;
22.  };
23.  int main()
24.  {
25.          drv3 obj;
26.          obj.a =10;
27.          obj.b =20;
28.          obj.c =30;
29.          obj.d= obj.a + obj.b + obj.c;
30.          cout<<"\t\t Total Sum ="<<obj.d;
31.  return 0;
32.  }
```

Output

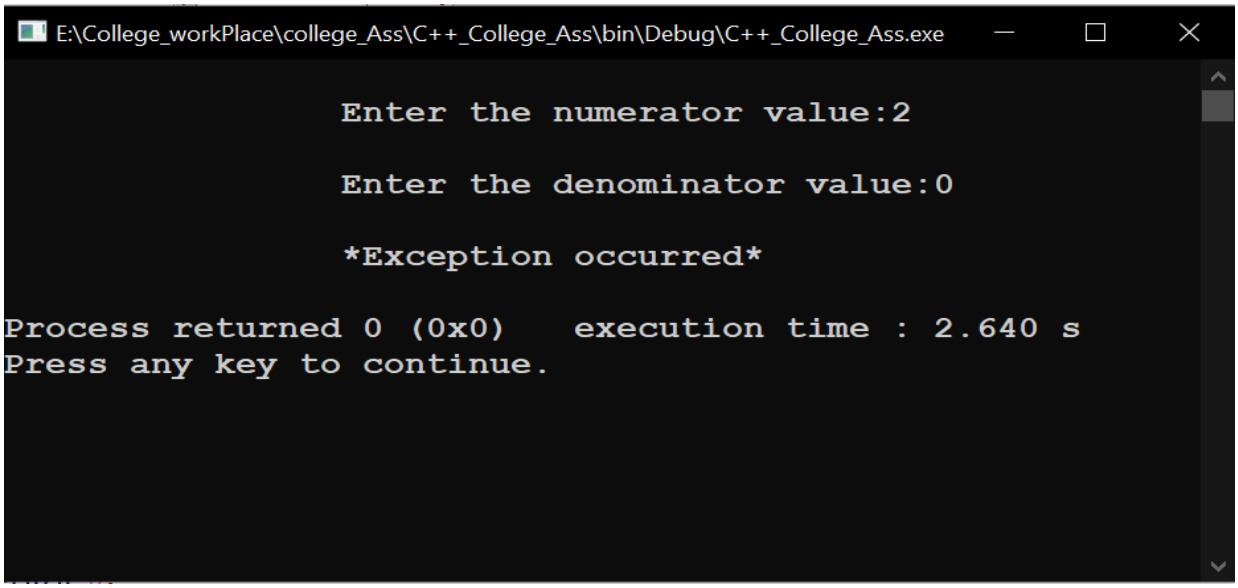
```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
                Total Sum =60
Process returned 0 (0x0)   execution time : 0.016 s
Press any key to continue.
```

Q11 Write a program for performing division. If divisor is zero then take the appropriate action at run time using the concept of try/catch block.

Program

```
1.  #include <iostream>
2.  using namespace std;
3.
4. float Check_Denominator(float den)
5. {
6.     if (den == 0)
7.         throw "Error";
8.     else
9.         return den;
10.  }
11.
12.  int main()
13.  {
14.      float a, b, result;
15.      cout<<"\n\t\tEnter the numerator value:";
16.      cin>>a;
17.      cout<<"\n\t\tEnter the denominator value:";
18.      cin>>b;
19.      try {
20.          if (Check_Denominator(b)) {
21.
22.              result = (a / b);
23.              cout << "\n\t\tThe Result is = "
24.                   << result << endl;
25.          }
26.      }
27.      catch (...) {
28.          cout <<"\n\t\t*Exception occurred*" << endl;
29.      }
30.   return 0;
31.  }
```

Output



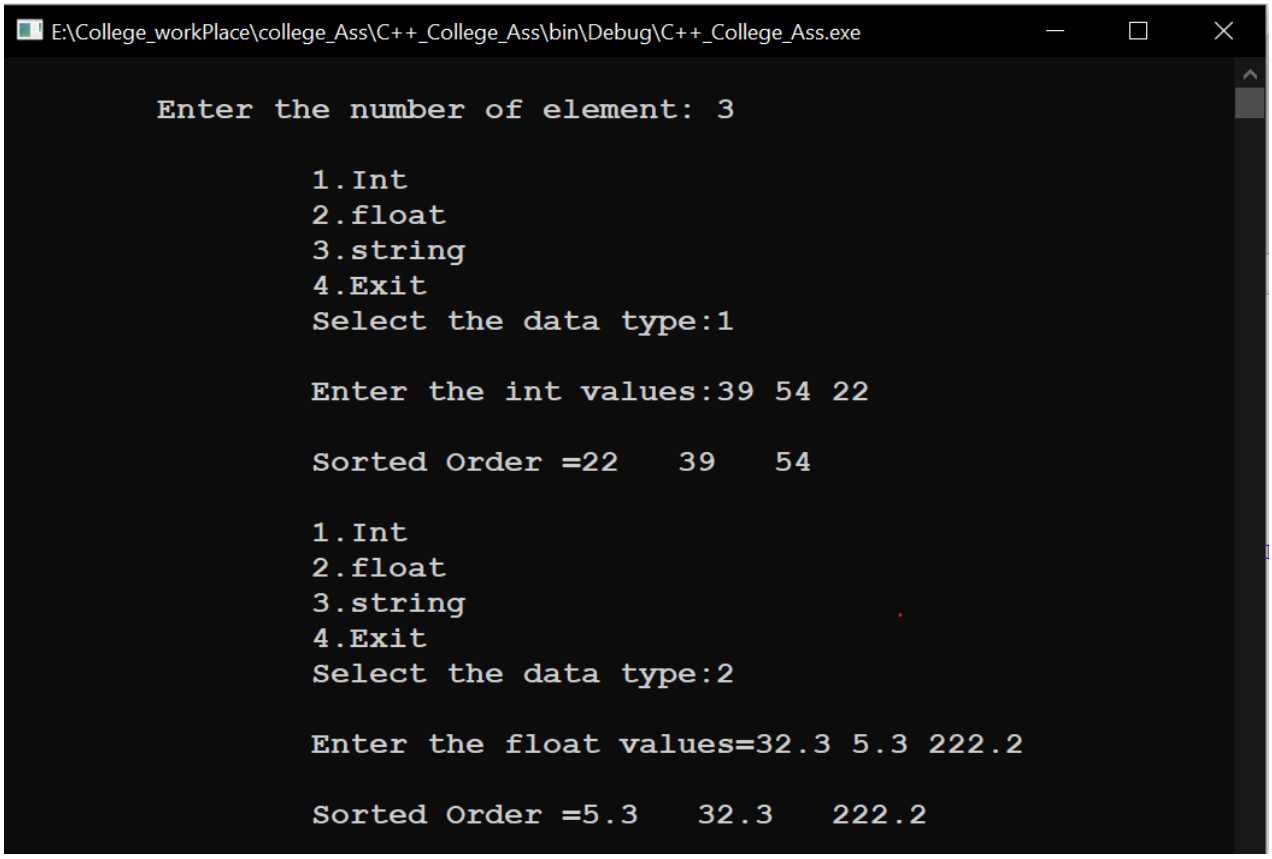
Q12 Using the generic programming approach, write a program to sort data of different data types.

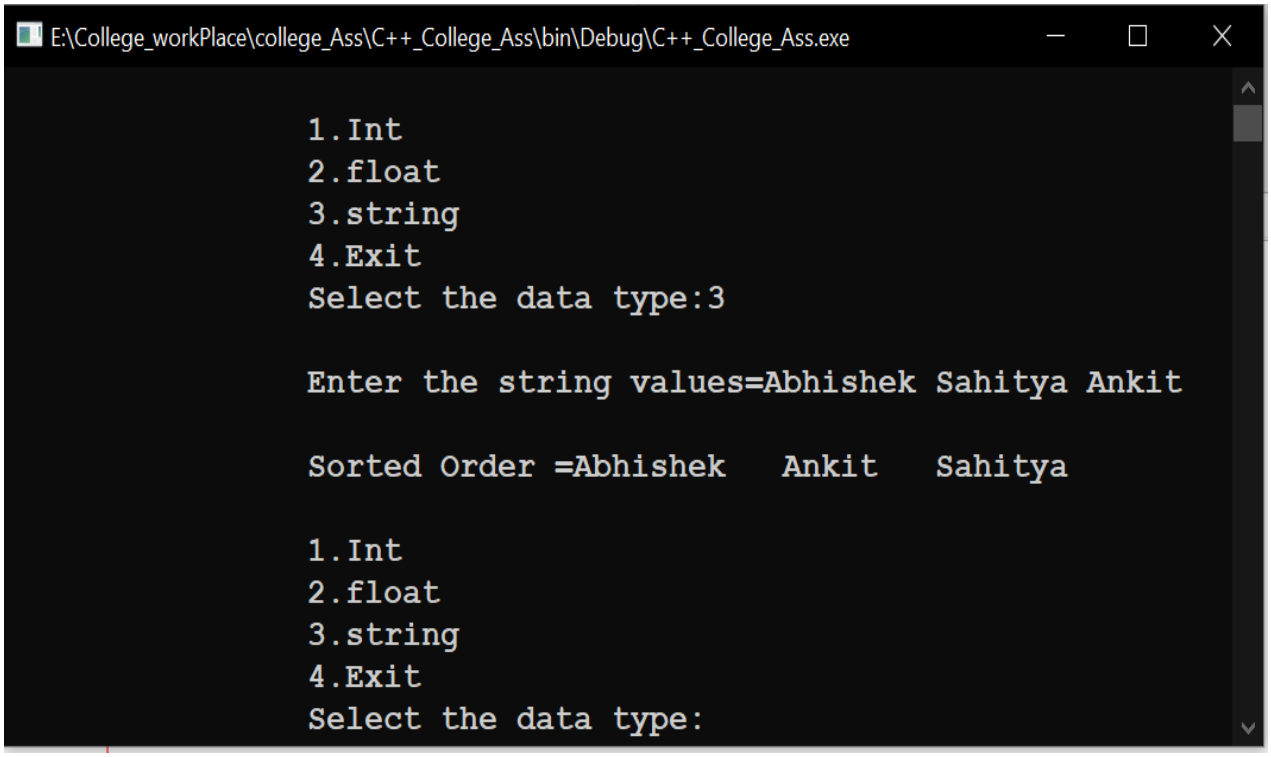
Program

```
1. #include <iostream>
2. #include<string>
3. using namespace std;
4.
5. template <class T>
6. void sortArray(T a[], int n)
7. {
8.     bool b = true;
9.     while (b) {
10.          b = false;
11.          for (size_t i=0; i<n-1; i++) {
12.              if (a[i] > a[i + 1]) {
13.                  T temp = a[i];
14.                  a[i] = a[i + 1];
15.                  a[i + 1] = temp;
16.                  b = true;
17.              }
18.          }
19.      }
20. }
21.
22. template <class T>
23. void printArray(T a[], int n)
24. {
25.     cout<<"\n\t\tSorted Order =";
26.     for (size_t i = 0; i < n; ++i)
27.         cout << a[i] << "   ";
28.     cout << endl;
29. }
30.
31. int main()
32. {
33.     int n,choice;
34.     cout<<"\n\tEnter the number of element: ";
35.     cin>>n;
36.     do
37.     {
38.     cout<<"\n\t\t1.Int\n\t\t2.float\n\t\t3.string\n\t\t4.Exit";
39.     cout<<"\n\t\tSelect the data type:";
40.     cin>>choice;
41.     switch(choice)
42.     {
43.     case 1:
44.         {
45.
46.         int* ptr_arr=new int[n];
47.         cout<<"\n\t\tEnter the int values:";
48.         for(int i=0;i<n;i++){
49.             cin>>ptr_arr[i];
50.         }
51.         sortArray(ptr_arr, n);
52.         printArray(ptr_arr, n);
53.         delete []ptr_arr;
54.         break;
```

```
55.          }
56.          case 2:
57.          {
58.          float* ptr_arr=new float[n];
59.          cout<<"\n\t\tEnter the float values=";
60.          for(int i=0;i<n;i++){
61.              cin>>ptr_arr[i];
62.          }
63.          sortArray(ptr_arr, n);
64.          printArray(ptr_arr, n);
65.          delete []ptr_arr;
66.          break;
67.              }
68.      case 3:
69.          {
70.          string* ptr_arr=new string[n];
71.          cout<<"\n\t\tEnter the string values=";
72.          for(int i=0;i<n;i++){
73.              cin>>ptr_arr[i];
74.          }
75.          sortArray(ptr_arr, n);
76.          printArray(ptr_arr, n);
77.          delete []ptr_arr;
78.          break;
79.          }
80.      case 4:
81.          exit(0);
82.  }
83.
84.  }while(1);
85.  return 0;
86.  }
```

Output


E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
Enter the number of element: 3
1.Int
2.float
3.string
4.Exit
Select the data type:1
Enter the int values:39 54 22
Sorted Order =22 39 54
1.Int
2.float
3.string
4.Exit
Select the data type:2
Enter the float values=32.3 5.3 222.2
Sorted Order =5.3 32.3 222.2


E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
1.Int
2.float
3.string
4.Exit
Select the data type:3
Enter the string values=Abhishek Sahitya Ankit
Sorted Order =Abhishek Ankit Sahitya
1.Int
2.float
3.string
4.Exit
Select the data type:

Q13 Write a program to find the size of a file.

Program

```
1. #include <bits/stdc++.h>
2. using namespace std;
3. long int findSize(char file_name[])
4. {
5.     FILE* fp = fopen(file_name, "r");
6.     if (fp == NULL) {
7.         printf("File Not Found!\n");
8.         return -1;
9.     }
10.     fseek(fp,0L,SEEK_END);
11.      long int res = ftell(fp);
12.       fclose(fp);
13.       return res;
14.  }
15.  int main()
16.  {
17.     char text[100];
18.     fstream file;
19.     file.open("abhi.txt",ios::out);
20.     if(!file)
21.     {
22.         cout<<"Error in creating file!!!";
23.         return 0;
24.     }
25.     cout<<"File created successfully."<<endl;
26.     cout << "\n\n\t\t---Write text to be written on file---\n\n" <<
   endl;
27.     cin.getline(text, sizeof(text));
28.     file << text << endl;
29.     file.close();
30.     char file_name[] = {"abhi.txt" };
31.     long int res = findSize(file_name);
32.      if (res != -1)
33.          printf("\n\n\n\n\n\t\t----Size of the file is %ld bytes---
   \n", res);
34.     return 0;
35.  }
```

Output

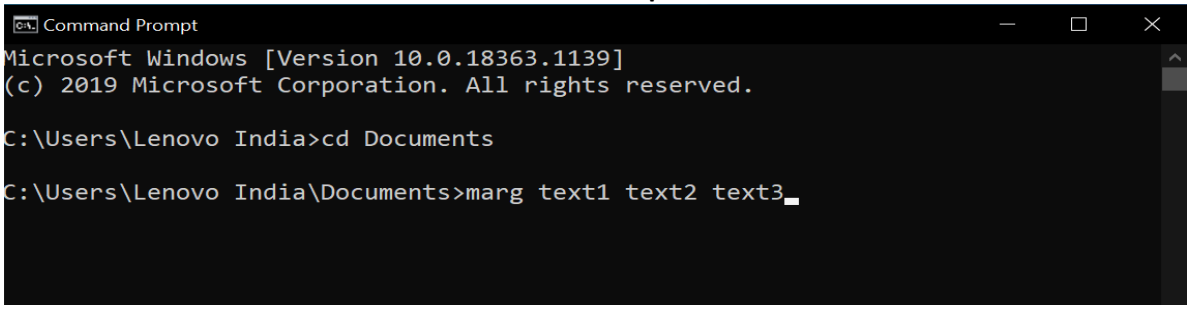
```
E:\College_workPlace\college_Ass\C++_College_Ass\bin\Debug\C++_College_Ass.exe
File created successfully.

                ---Write text to be written on file---

My Name is Abhishek Gupta . I am Student of LNCT,Bhopal,MP.




                ----Size of the file is 61 bytes---

Process returned 0 (0x0)   execution time : 115.285 s
Press any key to continue.
```

Q14 Write a program to merge two files into a third file using command line arguments.

Program

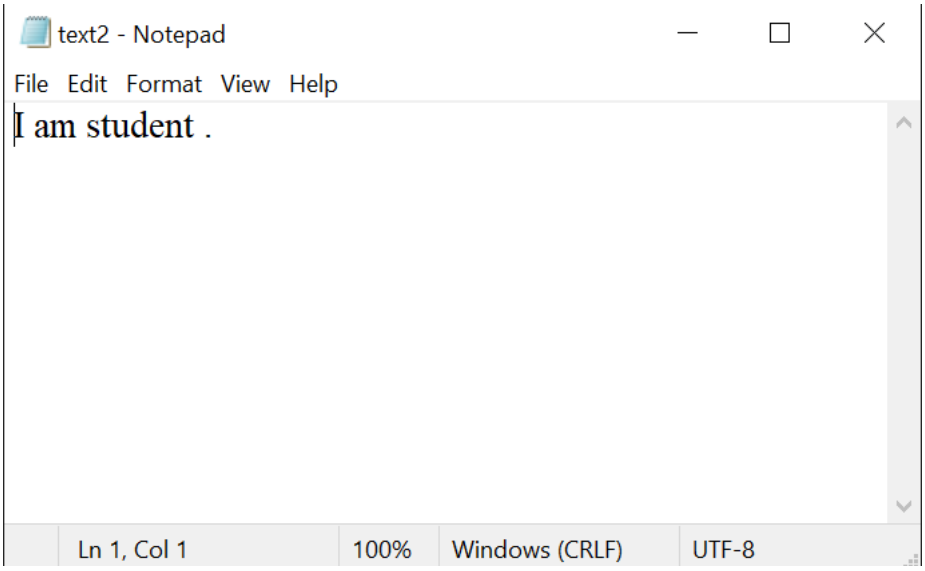
```
1. #include <fstream>
2. #include <iostream>
3. #include <stdlib.h>
4. #include <process.h>
5. using namespace std;
6. int main(int argc, char* argv[] )
7. {
8.
9.     char ch;
10.      ifstream infile;
11.      infile.open( argv[1] );
12.      if( !infile )
13.      {
14.          cerr << "\nCan't open " << argv[1];
15.          exit(-1);
16.      }
17.      ofstream outfile;
18.      outfile.open( argv[3] );
19.      if( !outfile )
20.      {
21.          cerr << "\nCan't open " << argv[3];
22.          exit(-1);
23.      }
24.      while( infile )
25.      {
26.          infile.get(ch);
27.          outfile.put(ch);
28.      }
29.      infile.close();
30.
31.      infile.open( argv[2] );
32.      if( !infile )
33.      {
34.          cerr << "\nCan't open " << argv[2];
35.          exit(-1);
36.      }
37.
38.      while( infile )
39.      {
40.          infile.get(ch);
41.          outfile.put(ch);
42.      }
43.      infile.close();
44.      outfile.close();
45.      return 0;
46.  }
```

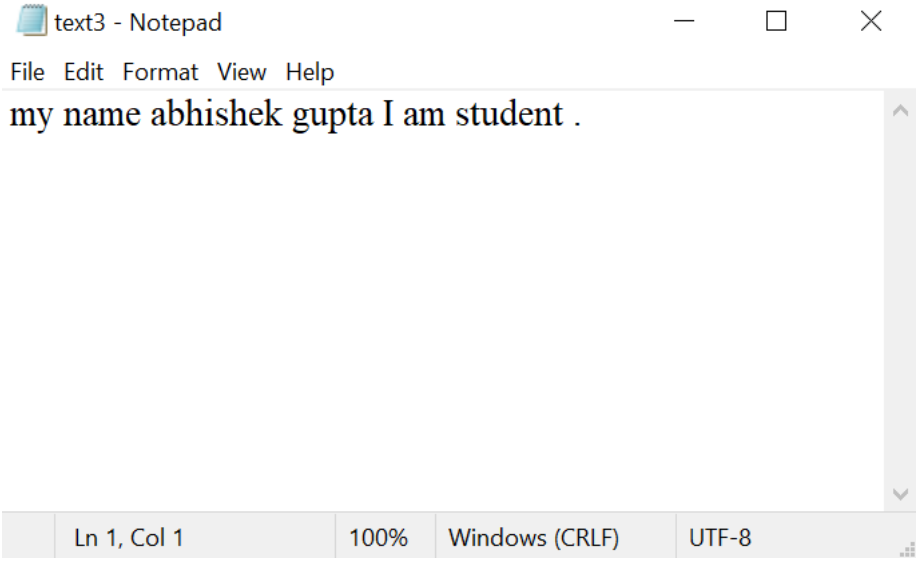
Command Prompt



Output


text1 - Notepad
File Edit Format View Help
my name abhishek gupta
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8


text2 - Notepad
File Edit Format View Help
I am student .
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8


text3 - Notepad
File Edit Format View Help
my name abhishek gupta I am student .
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8

Q15 Make a menu driven program to insert, retrieve, update and delete the records of students in file. Programs should be able to update either the complete record or some of the attributes of the records.

Program

```
1. #include<iostream>
2. #include<fstream>
3. #include<iomanip>
4. #include<stdlib.h>
5.
6. using namespace std;
7.
8. class Student
9. {
10.   int admno;
11.   char name[20];
12.   char gender;
13.   int std;
14.   float marks;
15.   float percentage;
16.
17.   public:
18.    void getData();
19.    void showData();
20.    int getAdmno(){return admno;}
21.  }s;
22.
23.  void Student::getData()
24.  {
25.   cout<<"\t\n\nEnter Student Details......\n";
26.   cout<<"\tEnter Admission No.     : "; cin>>admno;
27.   cout<<"\tEnter Full Name         : "; cin.ignore();
   cin.getline(name,20);
28.   cout<<"\tEnter Gender (M/F)      : "; cin>>gender;
29.   cout<<"\tEnter Standard          : "; cin>>std;
30.   cout<<"\tEnter Marks (out of 500): "; cin>>marks;
31.   cout<<endl;
32.   percentage=marks*100.0/500.00;
33.  }
34.
35.  void Student::showData()
36.  {
37.   cout<<"\n\n.......Student Details......\n";
38.   cout<<"\tAdmission No.     : "<<admno<<endl;
39.   cout<<"\tFull Name         : "<<name<<endl;
40.   cout<<"\tGender            : "<<gender<<endl;
41.   cout<<"\tStandard          : "<<std<<endl;
42.   cout<<"\tMarks (out of 500): "<<marks<<endl;
43.   cout<<"\tPercentage        : "<<percentage<<endl;
44.   cout<<endl;
45.  }
46.
47.  void addData()
48.  {
49.   ofstream fout;
50.   fout.open("Students.dat",ios::binary|ios::out|ios::app);
51.   s.getData();
```

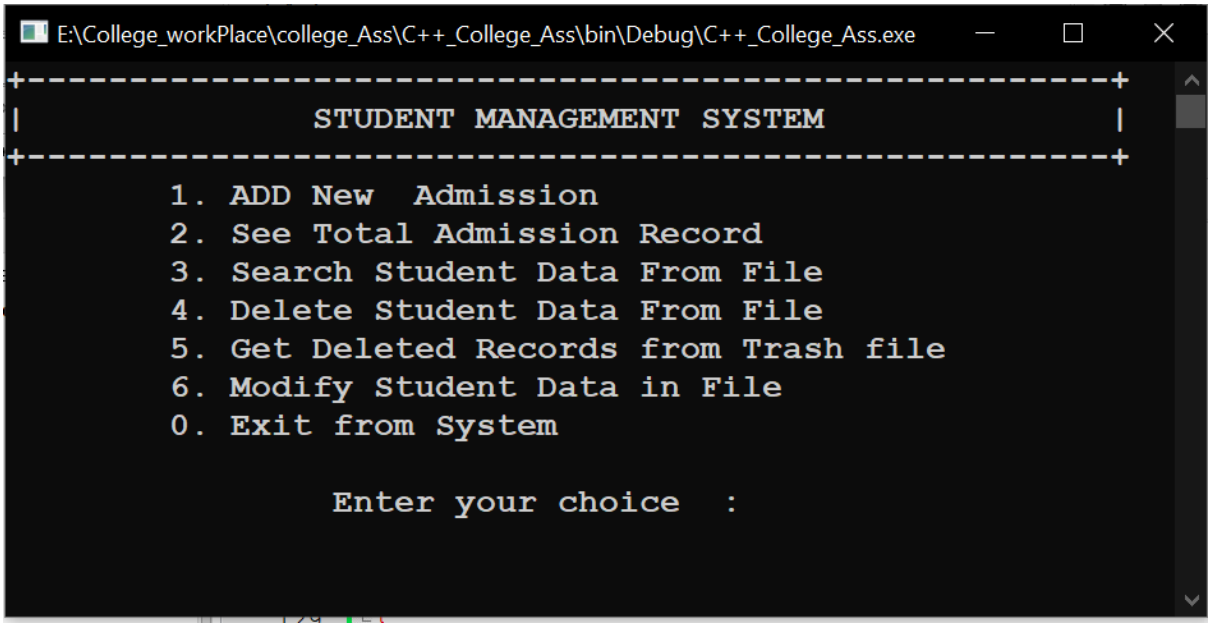
```
52.   fout.write((char*)&s,sizeof(s));
53.   fout.close();
54.   cout<<"\t\n\nData Successfully Saved to File....\n";
55.  }
56.
57.  void displayData()
58.  {
59.   ifstream fin;
60.   fin.open("Students.dat",ios::in|ios::binary);
61.   while(fin.read((char*)&s,sizeof(s)))
62.   {
63.    s.showData();
64.   }
65.   fin.close();
66.   cout<<"\t\n\nData Reading from File Successfully Done....\n";
67.  }
68.
69.  void searchData()
70.  {
71.   int n, flag=0;
72.   ifstream fin;
73.   fin.open("Students.dat",ios::in|ios::binary);
74.   cout<<"\n\tEnter Admission Number you want to search : ";
75.   cin>>n;
76.
77.   while(fin.read((char*)&s,sizeof(s)))
78.   {
79.    if(n==s.getAdmno())
80.    {
81.     cout<<"The Details of Admission No. "<<n<<" shown herewith:\n";
82.     s.showData();
83.     flag++;
84.    }
85.   }
86.   fin.close();
87.   if(flag==0)
88.    cout<<"The Admission No. "<<n<<" not found....\n\n";
89.   cout<<"\n\nData Reading from File Successfully Done....\n";
90.  }
91.
92.  void deleteData()
93.  {
94.   int n, flag=0;
95.   ifstream fin;
96.   ofstream fout,tout;
97.
98.   fin.open("Students.dat",ios::in|ios::binary);
99.   fout.open("TempStud.dat",ios::out|ios::app|ios::binary);
100.  tout.open("TrashStud.dat",ios::out|ios::app|ios::binary);
101.
102.
103.
104.
105.  cout<<"Enter Admission Number you want to move to Trash : ";
106.  cin>>n;
107.
108.  while(fin.read((char*)&s,sizeof(s)))
109.  {
110.   if(n==s.getAdmno())
111.   {
```

```
112.    cout<<"The Following Admission No. "<<n<<" has been moved to
   Trash:\n";
113.    s.showData();
114.    tout.write((char*)&s,sizeof(s));
115.    flag++;
116.   }
117.   else
118.   {
119.    fout.write((char*)&s,sizeof(s));
120.   }
121.  }
122.  fout.close();
123.  tout.close();
124.  fin.close();
125.  if(flag==0)
126.   cout<<"The Admission No. "<<n<<" not found....\n\n";
127.  remove("Students.dat");
128.  rename("tempStud.dat","Students.dat");
129. }
130.
131. void getTrash()
132. {
133.  ifstream fin;
134.  fin.open("TrashStud.dat",ios::in|ios::binary);
135.  while(fin.read((char*)&s,sizeof(s)))
136.  {
137.   s.showData();
138.  }
139.  fin.close();
140.  cout<<"\t\n\nData Reading from Trash File Successfully
   Done....\n";
141. }
142.
143. void modifyData()
144. {
145.  int n, flag=0, pos;
146.  fstream fio;
147.
148.  fio.open("Students.dat",ios::in|ios::out|ios::binary);
149.
150.  cout<<"\n\tEnter Admission Number you want to Modify : ";
151.  cin>>n;
152.
153.  while(fio.read((char*)&s,sizeof(s)))
154.  {
155.   pos=fio.tellg();
156.   if(n==s.getAdmno())
157.   {
158.    cout<<"The Following Admission No. "<<n<<" will be modified with
   new data:\n";
159.    s.showData();
160.    cout<<"\n\nNow Enter the New Details....\n";
161.    s.getData();
162.    fio.seekg(pos-sizeof(s));
163.    fio.write((char*)&s,sizeof(s));
164.    flag++;
165.   }
166.  }
167.  fio.close();
168.
169.  if(flag==0)
```

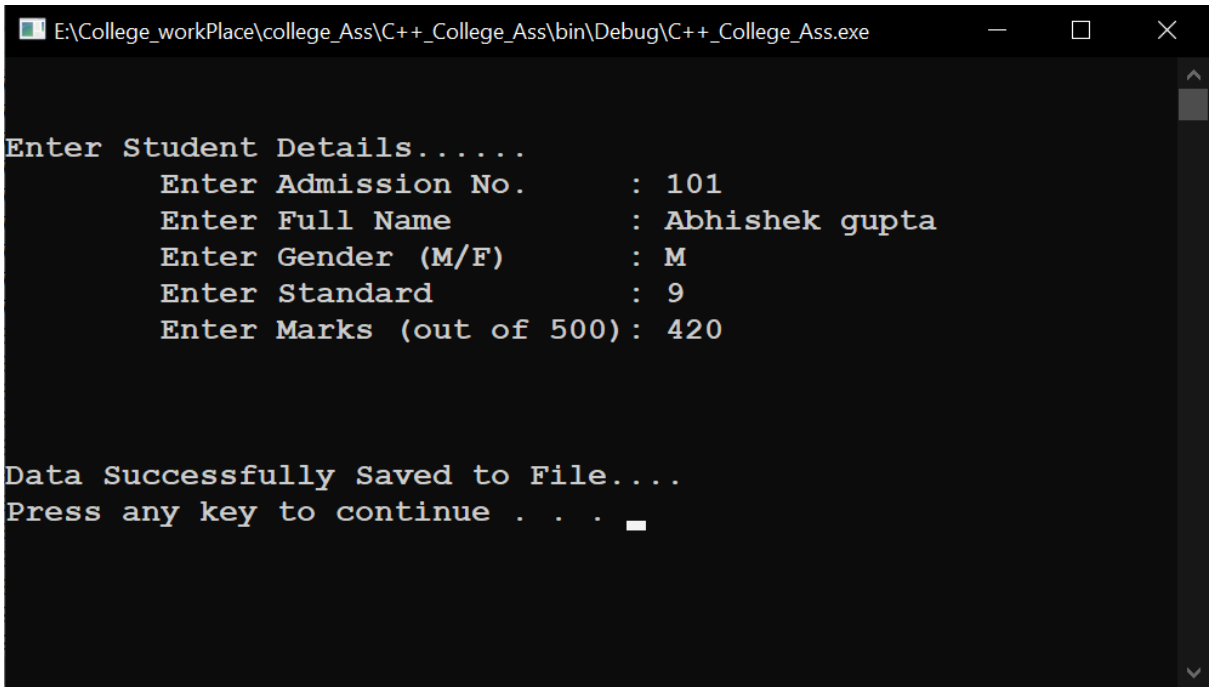
```
170.   cout<<"The Admission No. "<<n<<" not found....\n\n";
171. }
172.
173.
174. int main()
175. {
176. int ch;
177.  do
178.  {
179.   system("cls");
180.   cout<<"+-------------------------------------------------------
   +\n";
181.   cout<<"|              STUDENT MANAGEMENT SYSTEM
   |\n";
182.   cout<<"+-------------------------------------------------------
   +\n";
183.   cout<<"\t1. ADD New  Admission \n";
184.   cout<<"\t2. See Total Admission Record\n";
185.   cout<<"\t3. Search Student Data From File\n";
186.   cout<<"\t4. Delete Student Data From File\n";
187.   cout<<"\t5. Get Deleted Records from Trash file\n";
188.   cout<<"\t6. Modify Student Data in File\n";
189.   cout<<"\t0. Exit from System\n";
190.   cout<<"\n\t\tEnter your choice  : ";
191.   cin>>ch;
192.   system("cls");
193.   switch(ch)
194.   {
195.    case 1: addData(); break;
196.    case 2: displayData(); break;
197.    case 3: searchData(); break;
198.    case 4: deleteData(); break;
199.    case 5: getTrash(); break;
200.    case 6: modifyData(); break;
201.   }
202.   system("pause");
203.  }while(ch);
204. }
```

Output

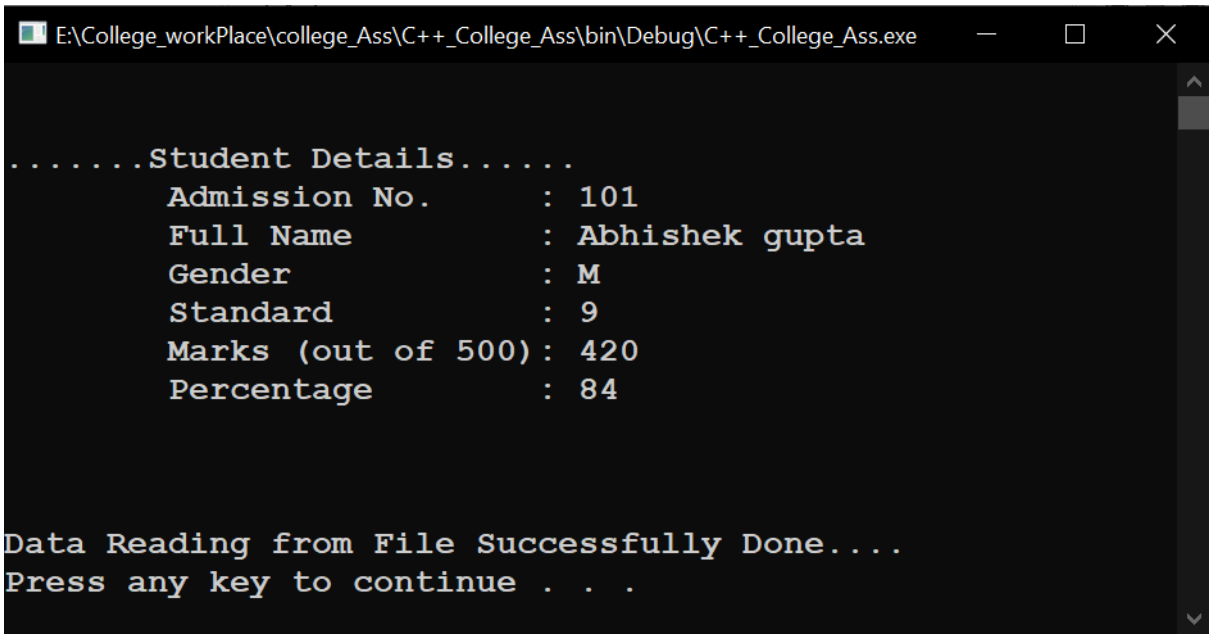
1 > Menu Driven Window



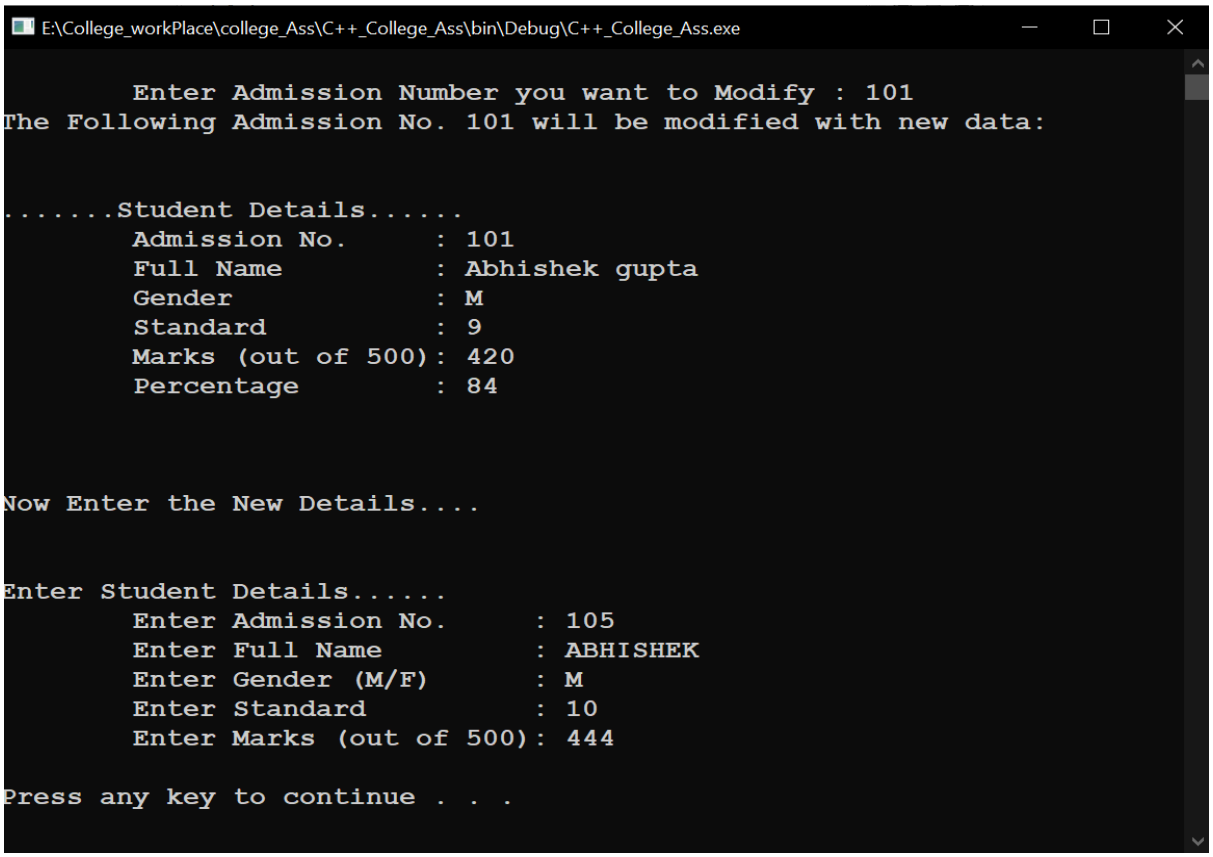
2> Insert Data Window



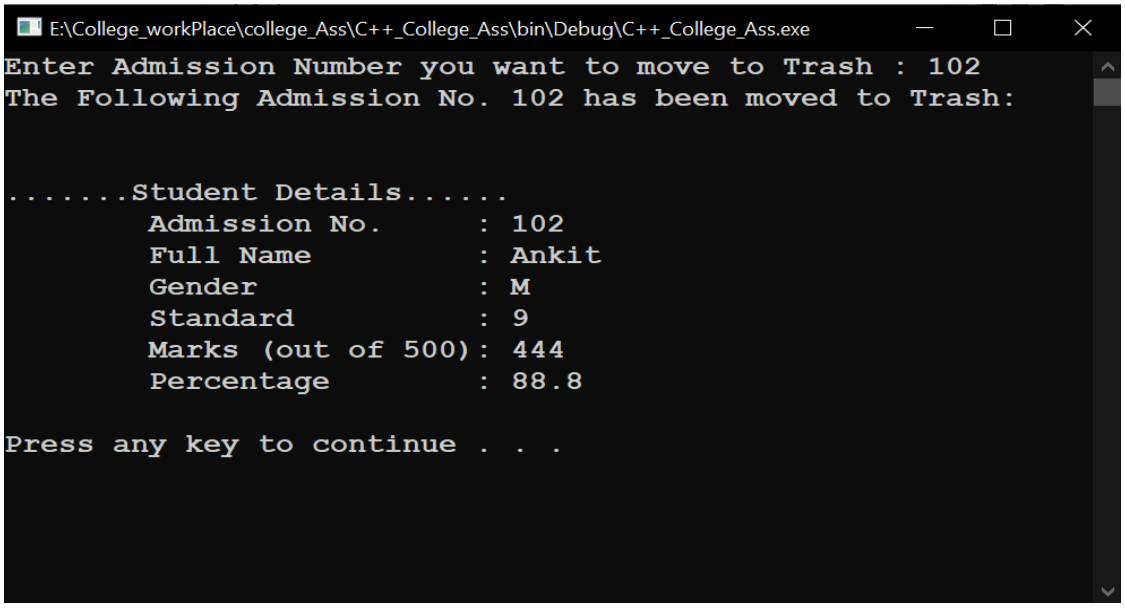
3> Retrieve Window



4> Update Window



5> Delete Window



The End