**DR. AKHILESH DAS GUPTA INSTITUTE OF TECHNOLOGY & MANAGEMENT**



**LAB MANUAL**

**Object Oriented Programming**

ETCS-210

**Name: Kunal Kashyap**

**Roll no: 01015607221**

**Branch: CSE LE 1st shift**

**Submitted to: MS. Garima**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Experiments | Date | Sign |
| 1 | Write a program to create an object in c++ |  |  |
| 2 | Write a program to calculate simple interest. |  |  |
| 3 | Write a program to multiply two matrices. |  |  |
| 4 | Write a program to add two complex number using friend function. |  |  |
| 5 | Write a program for constructor and destructor. |  |  |
| 6 | Write a program for unary operator overloading. |  |  |
| 7 | Write a program for binary operator overloading. |  |  |
| 8 | Write a program to create a function template. |  |  |
| 9 | Write a program to create a class template in. |  |  |
| 10 | Write a program for writing and reading file. |  |  |

**EXPERIMENT – 1**

**AIM:-** Write a program to create an object in c++.

#include <iostream>

using std::string;

using namespace std;

class empolyee

{

string name;

string department;

int emp\_id;

public:

void get()

{

cout<< "enter the employee name" <<endl; cin>> name;

cout<< "enter the department"<<endl; cin>> department;

cout<< "enter the empolyee id"<<endl; cin>> emp\_id;

}

void dis()

{

cout <<"empolyee name :"<< name<<endl; cout <<"department :"<< department<<endl; cout <<"employee id :"<< emp\_id<<endl;

}

};

int main()

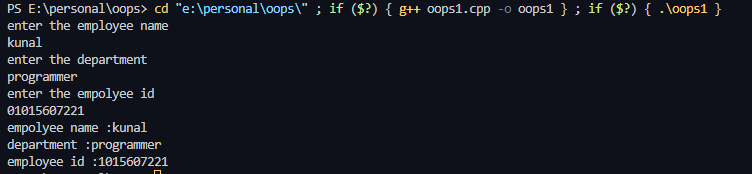
{

empolyee a1;

a1.get();

a1.dis();

}

****

**EXPERIMENT – 2**

**AIM:-** Write a program to calculate simple interest.

#include <iostream>

using namespace std;

class simple

{

int a, r, n;

public:

void get()

{

cout << "enter the ammount" << endl; cin >> a;

cout << "enter the rate of intrest" << endl; cin >> r;

cout << "enter the time in months" << endl; cin >> n;

}

void dis()

{

float simp = (a \* n \* r) / (12 \* 100);

cout << "you simple intrest is :" << simp << endl;

}

};

int main()

{

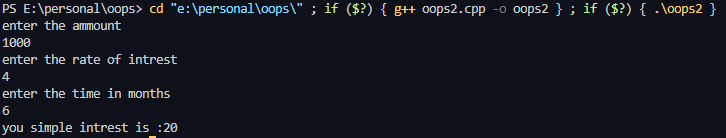
simple intrest;

intrest.get();

intrest.dis();

return 0;

}



**EXPERIMENT – 3**

**AIM:-** Write a program to multiply two matrices**.**

#include <iostream>

using namespace std;

class multiply

{

int arr[3][3];

int arr1[3][3];

int arr3[3][3];

int a = 0;

public:

void get()

{

cout << "first matrix" << endl;

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

cout << "enter the element for row" << i + 1 << " and column " << j + 1 << ":";

cin >> arr[i][j];

}

}

cout << "second matrix" << endl;

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

cout << "enter the element for row" << i + 1 << " and column " << j + 1 << ":";

cin >> arr1[i][j];

}

}

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

for (int k = 0; k < 3; k++)

{

a += arr[i][k] \* arr1[k][j];

}

arr3[i][j] = a;

a = 0;

}

}

}

void dis()

{

cout << "multiplication of first and second matrices is :" << endl;

for (int j = 0; j < 3; j++)

{

for (int i = 0; i < 3; i++)

{

cout << arr3[j][i] << " ";

}

cout << endl;

}

}

};

int main()

{

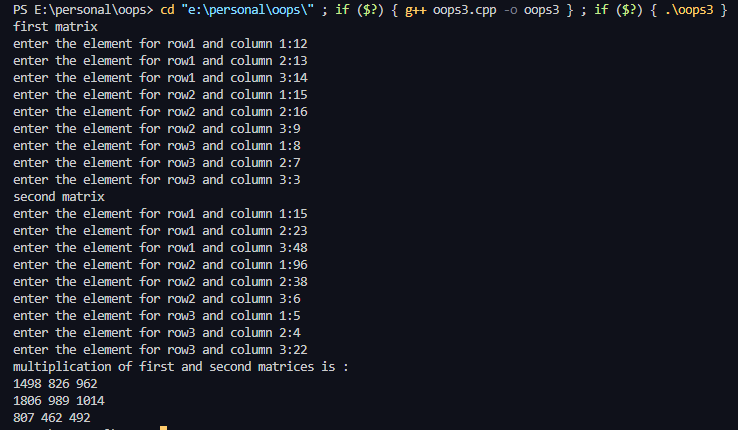
multiply matrix;

matrix.get();

matrix.dis();

return 0;

}



**EXPERIMENT – 4**

**AIM:-** Write a program to add two complex number using friend function

#include <iostream>

using namespace std;

class addition

{

int a, b, c, d,e ,f;

public:

void get()

{

cout << "enter the real and imaginary part of first complex number " << endl;

cin >> a >> b;

cout << "enter the real and imaginary part of second complex number " << endl;

cin >> c >> d;

}

void dis()

{

cout <<"the sum of two complex number is :"<<e<<" + "<<f<<"i"<<endl;

}

friend void add(addition &ad);

};

void add(addition &ad)

{

ad.e = ad.a + ad.c;

ad.f = ad.b + ad.d;

}

int main()

{

addition a1;

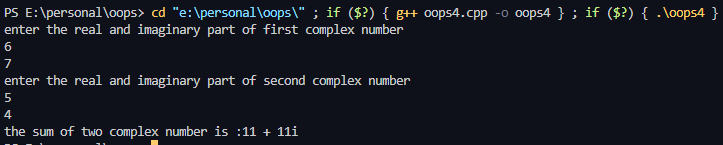
a1.get();

add(a1);

a1.dis();

return 0;

}



**EXPERIMENT – 5**

**AIM:-** Write a program for constructor and destructor.

#include<iostream>

using namespace std;

class omi

{

int a;

public:

omi()

{

a = 10;

cout<< "constructor invoked"<<endl;

}

~omi()

{

cout<<"destructor invoked"<< endl;

}

};

int main()

{

omi a1;

return 0;

}



**EXPERIMENT – 6**

**AIM:-** Write a program for unary operator overloading.

#include <iostream>

using namespace std;

class overload

{

int a;

public:

void get()

{

cout << "enter the element" << endl; cin >> a;

}

void operator++()

{

a = a + 10;

}

void dis()

{

cout << "element after overloading" << endl; cout << a << endl;

}

};

int main()

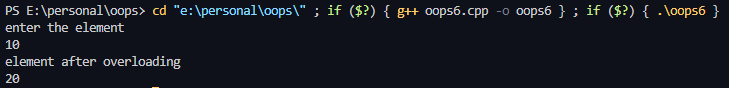
{

overload a1;

a1.get();

++a1;

a1.dis();

}

**EXPERIMENT – 7**

**AIM:-** Write a program for binary operator overloading.

#include <iostream>

using namespace std;

class addition

{

int a, b;

public:

void get()

{

cout << "enter the real and imaginary part of first complex number " << endl;

cin >> a >> b;

}

void dis()

{

cout << "the sum of two complex number is :" << a << " + " << b << "i" << endl;

}

friend addition operator+(addition &ad, addition &ad2);

};

addition operator+(addition &ad, addition &ad2)

{

addition ad3;

ad3.a = ad.a + ad2.a;

ad3.b = ad.b + ad2.b;

return ad3;

}

int main()

{

addition a1, a2, a3;

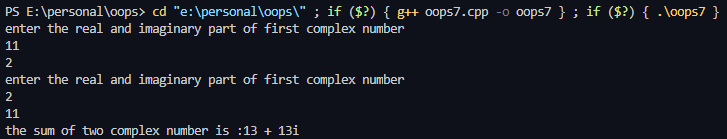
a1.get();

a2.get();

a3 = a1 + a2;

a3.dis();

}



**EXPERIMENT – 8**

**AIM:-** Write a program to create a function template in c++.

#include <iostream>

using namespace std;

template<class a>

a fun(a x, a y)

{

a res = x + y;

return res;

}

int main()

{

float a,b;

cout<< "enter two values in float"<< endl; cin>>a>>b;

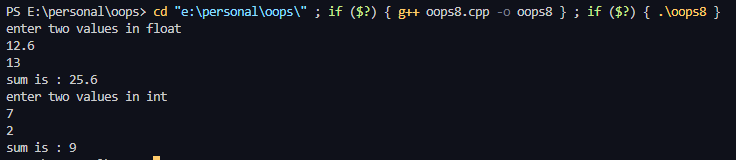
cout<<"sum is : "<<fun(a,b)<<endl;

int e,f;

cout<< "enter two values in int"<< endl; cin>>e>>f;

cout<<"sum is : "<<fun(e,f)<<endl;

}



**EXPERIMENT – 9**

**AIM:-** Write a program to create a class template in c++.

#include<iostream>

using namespace std;

template<class a>

class fun

{

private:

a x, y;

public:

void get()

{

cout<< "enter two values"<<endl;

cin>>x>>y;

}

void sum()

{

cout<<"sum is : "<<x+y<<endl;

}

};

int main()

{

fun<int> a1;

a1.get();

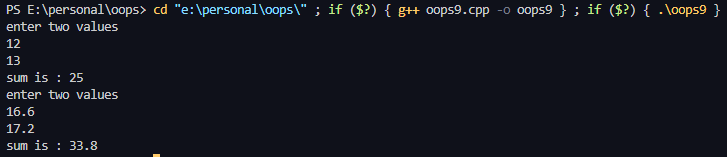
a1.sum();

fun<float> a2;

a2.get();

a2.sum();

}

****

**EXPERIMENT – 10**

**AIM:-** Write a program for writing and reading file in c++.

#include <iostream>

#include <fstream>

using namespace std;

int main()

{

string str0;

string str1;

string str2;

string str3;

ofstream ichigo("bleach.txt");

cout << "enter your first name " << endl; cin >> str0;

cout << "enter your last name " << endl; cin >> str1;

cout << "enter your age " << endl;

cin >> str2;

cout << "enter your occupation " << endl; cin >> str3;

ichigo << "Name:-" << str0 << " " << str1; ichigo << endl;

ichigo << "Age:-" << str2;

ichigo << endl;

ichigo << "Occupation:-" << str3;

ichigo.close();

string str4;

string str5;

string str6;

ifstream rukia("beach.txt");

getline(rukia, str4);

getline(rukia, str5);

getline(rukia, str6);

cout << endl;

cout << str4 << endl;

cout << str5 << endl;

cout << str6 << endl;

rukia.close();

}

