**Experiment – 16**

**Aim:** Write a program to demonstrate the use of special functions constructor and destructor in the class template.

**SOURCE CODE:**

#include <iostream>

using namespace std;

template <class T> class BiggerNum {

public :

T num;

BiggerNum (T a) {

num = a;

cout<<"Constructor called"<<endl;}

~ BiggerNum () {

cout << "Destructor called." << endl;}};

int main() {

int a, b;

cout << "Two numbers : ";

cin >> a >> b;

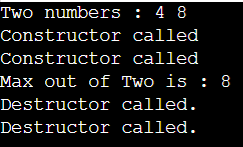
BiggerNum<int> obj1(a);

BiggerNum<int> obj2(b);

cout << "Max out of Two is : ";

cout << max(obj1.num, obj2.num) << endl;}

**OUTPUT:**

****

**Experiment – 17**

**Aim:** Write a program to demonstrate how template function can be overloaded.

**SOURCE CODE:**

#include <iostream>

using namespace std;

template<class T>

void f(T x, T y) {

cout << "Template\t" <<x<<" "<<y<<endl; }

void f(int w, int z) {

cout << "Non-template\t" <<w<<" "<<z<<endl; }

int main() {

f( 9 , 5 );

f('a', 'b');

f( 3 , 'b');

}

**OUTPUT:**

Graphical user interface, text

Description automatically generated