Course code: CSC-284
Lab Report: 05

Submitted To

Tanzina Tasnim Bithi

Department of Computer, Science and Engineering

Submitted By

Name: Md.Pranto Ali
ID: 23303101
Program: BCSE
Section: A
Semester: Fall 2024

Submission date: 15/12/2024

Teacher signature

1. Default Constructor:

main.cpp

Share Run

```cpp
#include <iostream>
using namespace std;
class Students {
private:
    string name;
    int age;
public:
    Students() {
        name = "Md.Pranto";
        age = 20;
        cout << "Default constructor called!" << endl;
    }
    Students(string n, int a) {
        name = n;
        age = a;
        cout << "Parameterized constructor called here parameter is n and a" << endl;
    };
    Students(const Students &p) {
        age = p.age;
        cout << "Copy constructor called this function name = p.name and age = p.age" << endl;
    };
    void display() {
        cout << "Name: " << name << ", Age: " << age << endl;
    };
};
int main() {
    cout << "Name: MD.Pranto Ali" << endl << "ID: 23303101" << endl;
```

Output Clear

```
Name: MD.Pranto Ali
ID: 23303101
Default constructor called!
Name: Md.Pranto, Age: 20
Parameterized constructor called here parameter is n and a
Name: Santo, Age: 16
Copy constructor called this function name = p.name and age = p.age
Name: , Age: 16


=== Code Execution Successful ===
```

2. Copy Constructor:

main.cpp Share Run

```cpp
#include <iostream>
using namespace std;
class Example {
    int x;
public:
    Example(int val){
        x = val;
    };
    Example(Example &obj){
        x = obj.x;
        cout << "Copy Constructor Called" << endl;
    };
    void display() {
        cout << "Value of x: " << x << endl;
    };
};
int main() {
    cout << "Name: MD.Pranto Ali" << endl << "ID: 23303101" <<
        endl;
    Example obj1(10);  // Calls parameterized constructor
    Example obj2 = obj1;  // Calls copy constructor
    cout << "Object 1: " << endl;
    obj1.display();
    cout << "Object 2: " << endl;
    obj2.display();
    return 0;
}

```

Output Clear

```
Name: MD.Pranto Ali
ID: 23303101
Copy Constructor Called
Object 1:
Value of x: 10
Object 2:
Value of x: 10


=== Code Execution Successful ===
```

3. Parameterized Constructor:

main.cpp | Share | Run | Output | Clear

```cpp
#include <iostream>
#include <string>
using namespace std;
class students {
    private:
        string name;
        int age;
        int id;
    public:
        students(string nam, int ag, int ro) {
            name = nam;
            age = ag;
            id = ro;
            cout << "Name: " << name << endl << "Age: " << age
                << endl;
            cout << "ID: " << id << endl;
        };
};
int main() {
    cout << "Name: MD.Pranto Ali" << endl << "ID: 23303101" <<
        endl;
    students one("Md.Pranto Islam", 20, 23303101);
    return 0;
};
```

```
Name: MD.Pranto Ali
ID: 23303101
Name: Md.Pranto Islam
Age: 20
ID: 23303101


=== Code Execution Successful ===
```

4. Multiple Constructor:

main.cpp

Share Run

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

class Student {
private:
    string name;
    int age;
public:
    Student() {
        name = "Md.Pranto Islam";
        age = 20;
        cout << "Default constructor called: " << name << ", Age
            : " << age << endl;
    }
    Student(string studentName) {
        name = studentName;
        age = 0;
        cout << "Constructor with name called: " << name << ",
            Age: " << age << endl;
    }
    Student(string studentName, int studentAge) {
        name = studentName;
        age = studentAge;
        cout << "Constructor with name and age called: " << name
            << ", Age: " << age << endl;
    }
};
int main() {
    cout << "Name: MD.Pranto Ali" << endl << "ID: 23303101" <<
        endl;
```

Output Clear

```
Name: MD.Pranto Ali
ID: 23303101
Default constructor called: Md.Pranto Islam, Age: 20
Constructor with name called: Md.Turjo islam, Pranto, Age: 0
Constructor with name and age called: Turjo, Age: 200


=== Code Execution Successful ===
```

5. Constructor with default argument:

main.cpp

Share Run

```cpp
#include <iostream>
using namespace std;
class Student {
private:
    string name;
    int age;

public:
    Student(string studentName = "Md.Pranto Islam", int
        studentAge = 20) {
        name = studentName;
        age = studentAge;
        cout << "Constructor called: " << name << ", Age: " <<
            age << endl;
    }
};

int main() {
    cout << "Name: MD.Pranto Ali" << endl << "ID: 23303101" <<
        endl;
    Student student1;
    Student student2("Md.Ikbal Hossain");
    Student student3("MD.Irfan Islam", 18);
    return 0;
}

```

Output Clear

```
Name: MD.Pranto Ali
ID: 23303101
Constructor called: Md.Pranto Islam, Age: 20
Constructor called: Md.Ikbal Hossain, Age: 20
Constructor called: MD.Irfan Islam, Age: 18


=== Code Execution Successful ===
```

6. Copy Constructor:

main.cpp

```cpp
#include <iostream>
using namespace std;
class Example {
    int x;
public:
    Example(int val){
        x = val;
    };
    Example(Example &obj){
        x = obj.x;
        cout << "Copy Constructor Called" << endl;
    };
    void display() {
        cout << "Value of x: " << x << endl;
    };
};
int main() {
    cout << "Name: MD.Pranto Ali" << endl << "ID: 23303101" <<
        endl;
    Example obj1(10);
    Example obj2 = obj1;
    cout << "Object 1: " << endl;
    obj1.display();
    cout << "Object 2: " << endl;
    obj2.display();
    return 0;
}
```

Output

```
Name: MD.Pranto Ali
ID: 23303101
Copy Constructor Called
Object 1:
Value of x: 10
Object 2:
Value of x: 10


=== Code Execution Successful ===
```