UNIVERSITY OF ENGINEERING AND TECHNOLOGY

(NAROWAL CAMPUS)

Object-Oriented Programming Lab Manual

Created by: Muhammad Abdullah

Registration Number: 2022-CS-525

Topics: Copy Constructor, Array of pointers to Object,

Dynamic Objects and their Destruction, Static Data Member and

Constant Member Functions

Lab Manual

(Object-Oriented Programming Lab)

- **Task 1:**

Create a class and make copy constructor in its definition in the C++ program

Program:

```
#include <iostream>
#include <string>
using namespace std;
class person
{
private:
int age;
string name, fname, city;
public:
person(string = "Abdullah", string = "Zahid", int = 19, string = "lahore");
person(person&);
~person(){}
void get();
void print();
};
int main()
{
    person Abdullah;
    Abdullah.get();
    person abdullah(Abdullah);
    Abdullah.print();
    abdullah.print();
    return 0;
}
person :: person(string a, string b, int c, string d) {
    name = a;
    fname = b;
    age = c;
    city = d;
}
person :: person(person &per){
    name = per.name;
    fname = per.fname;
    age = per.age;
    city = per.city;
}
void person :: print(){
    cout<<"Name: "<<name<<endl;
    cout<<"Father Name: "<<fname<<endl;
    cout<<"Age: "<<age<<endl;
    cout<<"City: "<<city<<endl;
}
void person :: get(){
    cout<<"Enter Name: ";
    getline(cin>>ws,name);
    cout<<"Enter Father Name: ";
    getline(cin>>ws,fname);
    cout<<"Enter City Name: ";
    getline(cin>>ws,city);
    cout<<"Enter Age: ";
    cin>>age;
}
```

Output:

```
Enter Name: Abdullah
Enter Father Name: Zahid
Enter City Name: Lahore
Enter Age: 19
Name: Abdullah
Father Name: Zahid
Age: 19
City: Lahore
Name: Abdullah
Father Name: Zahid
Age: 19
City: Lahore

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 7080) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

- ## Task 2:

Create an array of pointers that points to objects in the C++ program

Program:

```
#include <iostream>
using namespace std;
class client{
public:
client(int = 00, string = "Ali");
~client();
void get(int,string);
void print();
private:
int clientNo;
string project;
};
int main(){
    client** ptr = new client*[5];
    for(int i=0 ; i<5;i++){
        ptr[i] = new client[3];
    }
    for(int i = 0; i<5;i++){
        for(int j =0; j<3;j++){
            if((i+j)%2==0){
                ptr[i][j].get(i+j,"Web Development");
            }
            else{
                ptr[i][j].get(i+j,"Machine Learning");
            }
        }
    }
    for(int i = 0; i<5;i++){
        for(int j =0; j<3;j++){
            ptr[i][j].print();
        }
    }
    for(int i =0;i<5;i++){
        delete[] ptr[i];
    }
    delete ptr;
    ptr = NULL;
    return 0;
}
```

```
40  client :: client(int a , string b){
41      clientNo = a;
42      project = b;
43  }
44  void client :: print(){
45      cout<<" --: Client Order :-- "<<endl<<endl;
46      cout<<"Client No: "<<clientNo<<endl;
47      cout<<"Project Type: "<<project<<endl<<endl;
48  }
49  void client :: get (int a, string s){
50      clientNo = a;
51      project = s;
52  }
53  client :: ~client(){
54      cout<<"Object Destroyed Successfully"<<endl;
55  }
```

Output:

```
 --: Client Order :--

Client No: 5
Project Type: Machine Learning

 --: Client Order :--

Client No: 4
Project Type: Web Development

 --: Client Order :--

Client No: 5
Project Type: Machine Learning

 --: Client Order :--

Client No: 6
Project Type: Web Development

Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully
Object Destroyed Successfully

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 5612) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
 --: Client Order :--

Client No: 3
Project Type: Machine Learning

 --: Client Order :--

Client No: 4
Project Type: Web Development
```

- **Task 3:**

Create a dynamic object using a pointer and also perform its destruction in a C++ program

Program:

```cpp
#include <iostream>
#include <string>
using namespace std;
class Student{
private:
int age, id;
string name,fname;
public:
Student(string = "Abdullah", string= "Zahid", int=19, int=525);
~Student();
void get();
void print();
};
int main(){
    Student *ptr = new Student[2];
    for(int i=0;i<2;i++){
        ptr[i].get();
        cout<<endl;
    }
    for(int i=0;i<2;i++){
        ptr[i].print();
        cout<<endl;
    }
    delete[] ptr;
    ptr = NULL;
    return 0;
}
Student :: Student(string a, string b, int c, int d){
    name = a;
    fname = b;
    age = c;
    id = d;
}
Student :: Student(string Name, string Fname, int Age, int Id){
    name = Name;
    fname = Fname;
    age = Age;
    id = Id;
}
void Student :: get(){
    cout<<" --: Enter Student Information :--"<<endl;
    cout<<"Enter Name: ";
    getline(cin>>ws,name);
    cout<<"Enter Father Name: ";
    getline(cin>>ws,fname);
    cout<<"Enter Age: ";
    cin>>age;
    cout<<"Enter Id Number: ";
    cin>>id;
}
void Student :: print(){
    cout<<" --: Student Information :--"<<endl;
    cout<<"Name: "<<name<<endl;
    cout<<"Father Name: "<<fname<<endl;
    cout<<"Age: "<<age<<endl;
    cout<<"Id Number: "<<id<<endl;
}
Student :: ~Student(){
    cout<<"Object Destroyed Successfully";
}
```

Output:

```
 --: Enter Student Information :--
Enter Name: Muhammad Abdullah
Enter Father Name: Zahid Mehmood
Enter Age: 19
Enter Id Number: 525

 --: Enter Student Information :--
Enter Name: Muhammad Abu-Bakar
Enter Father Name: Zahid Mehmood
Enter Age: 14
Enter Id Number: 500

 --: Student Information :--
Name: Muhammad Abdullah
Father Name: Zahid Mehmood
Age: 19
Id Number: 525

 --: Student Information :--
Name: Muhammad Abu-Bakar
Father Name: Zahid Mehmood
Age: 14
Id Number: 500

Object Destroyed SuccessfullyObject Destroyed Successfully
D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 15740) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

- ## Task 4:

Create a class that contains static data members in a C++ program

Program:

```
#include <iostream>
using namespace std;
class resident
{
private:
    static int age;
public:
    resident();
    int display();
};
int resident :: age = 0;
int main()
{
    resident r1;
    cout<<"Employee Age of Resident 1: "<<r1.display()<<endl;
    resident r2;
    cout<<"Employee Age of Resident 2: "<<r2.display()<<endl;
    resident r3;
    cout<<"Employee Age of Resident 3: "<<r3.display()<<endl;
    return 0;
}
resident :: resident(){
    age++;
}
int resident :: display(){
    return age;
}
```

Output:

```
Employee Age of Resident 1: 1
Employee Age of Resident 2: 2
Employee Age of Resident 3: 3

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 8092) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

- ## Task 5:

Create a class that contains constant member functions and create constant objects in the C++ program

Program:

```cpp
#include <iostream>
#include <string>
using namespace std;
class user
{
private:
    int age,balance;
    string name, fname,city;
public:
    user(string = "Abdullah", string = "Zahid", int = 19, int = 1000, string = "Lahore");
    void getInfo();
    int getAmount() const;
    void deposit(int);
    void debit(int);
    void print() const;
};
int main()
{
    user Abdullah;
    Abdullah.getInfo();
    Abdullah.print();
    Abdullah.deposit(500);
    Abdullah.debit(200);
    Abdullah.print();
    const user Ali("Ali","Ahamd", 20);
    Ali.getAmount();
    Ali.print();
    return 0;
}
user :: user(string a, string b, int c, int d, string e){
    name = a;
    fname = b;
    age = c;
    balance = d;
    city = e;
}
```

```
37  void user :: getInfo(){
38      cout<<"Enter Name: ";
39      getline(cin>>ws,name);
40      cout<<"Enter Father Name: ";
41      getline(cin>>ws,fname);
42      cout<<"Enter Age: ";
43      cin>>age;
44      cout<<"Enter Balance: ";
45      cin>>balance;
46      cout<<"Enter City Name: ";
47      getline(cin>>ws,city);
48  }
49  int user :: getAmount() const{
50      return balance;
51  }
52  void user :: deposit(int value){
53      balance += value;
54  }
55  void user :: debit(int value){
56      balance -= value;
57  }
58  void user :: print() const{
59      cout<<endl<<" --: User Information :-- "<<endl;
60      cout<<"Name:         "<<name<<endl;
61      cout<<"Father Name: "<<fname<<endl;
62      cout<<"Age:          "<<age<<endl;
63      cout<<"Balance:      "<<balance<<endl;
64      cout<<"City:         "<<city<<endl;
65  }
```

Output:

```
Enter Name: Abdullah
Enter Father Name: Zahid
Enter Age: 19
Enter Balance: 30000
Enter City Name: Lahore

 --: User Information :--
Name:         Abdullah
Father Name: Zahid
Age:          19
Balance:      30000
City:         Lahore

 --: User Information :--
Name:         Abdullah
Father Name: Zahid
Age:          19
Balance:      30300
City:         Lahore

 --: User Information :--
Name:         Ali
Father Name: Ahamd
Age:          20
Balance:      1000
City:         Lahore

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 14368) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```
