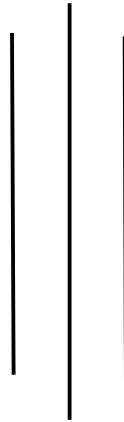


Object Oriented Programming Laboratory

Assignment – 5



Submitted By:

Yishap Khanal

21052960

CSE-34

Date: 20th October, 2022

Question 1:

```
1  // Write a program to demonstrate hybrid inheritance.
2
3  #include <iostream>
4  using namespace std;
5
6  class Base
7  {
8      public:
9          int a;
10 };
11
12 class Derived1 : virtual public Base
13 {
14     public:
15         int b;
16 };
17
18 class Derived2 : virtual public Base
19 {
20     public:
21         int c;
22 };
23
24 class Derived3 : public Derived2, public Derived1
25 {
26     public:
27         int total;
28 };
29
30 int main()
31 {
32     Derived3 obj;
33     obj.a = 10;
34     obj.b = 20;
35     obj.c = 30;
36     obj.total = obj.a + obj.b + obj.c;
37     cout << "Total = " << obj.total << endl;
38     return 0;
39 }
```

Output:

```
PS C:\Users\KIIT01\Desktop\programming> cd "c:\Users\KIIT01\Desktop\programming"
; if ($?) { .\q1_hybrid }
Total = 60
PS C:\Users\KIIT01\Desktop\programming\OOP\20 october>
```

Question 2:

```
1 // Write a program to demonstrate multiple inheritance.
2
3 #include <iostream>
4 using namespace std;
5
6 class Base1
7 {
8     public:
9         int a;
10 };
11
12 class Base2
13 {
14     public:
15         int b;
16 };
17
18 class Derived : public Base1, public Base2
19 {
20     public:
21         int total;
22 };
23
24 int main()
25 {
26     Derived obj;
27     obj.a = 10;
28     obj.b = 20;
29     obj.total = obj.a + obj.b;
30     cout << "Total = " << obj.total << endl;
31     return 0;
32 }
```

Output:

```
PS C:\Users\KIIT01\Desktop\programming> cd "c:\Users\KIIT01\Desktop\programming"
PS C:\Users\KIIT01\Desktop\programming> g++ q4_multiple_inhrtnc.cpp -o q4_multiple_inhrtnc
PS C:\Users\KIIT01\Desktop\programming> .\q4_multiple_inhrtnc
Total = 30
PS C:\Users\KIIT01\Desktop\programming\OOP\20 october>
```

Question 3:

```
1  // Write a program to demonstrate multilevel inheritance.
2
3  #include <iostream>
4  using namespace std;
5
6  class Base
7  {
8      public:
9          int a;
10 };
11
12 class Derived1 : public Base
13 {
14     public:
15         int b;
16 };
17
18 class Derived2 : public Derived1
19 {
20     public:
21         int total;
22 };
23
```

```
24 int main()
25 {
26     Derived2 obj;
27     obj.a = 10;
28     obj.b = 20;
29     cout << "a = " << obj.a << endl;
30     cout << "b = " << obj.b << endl;
31     obj.total = obj.a + obj.b;
32     cout << "Total = " << obj.total << endl;
33     return 0;
34 }
```

Output:

```
PS C:\Users\KIIT01\Desktop\programming> cd "c:\Users\KIIT01\Desktop\programming\OOP\20 october"
q5_multilevel_inhrtnc } ; if ($?) { .\q5_multilevel_inhrtnc }
a = 10
b = 20
Total = 30
PS C:\Users\KIIT01\Desktop\programming\OOP\20 october>
```

Question 4:

```
1  // Write a program to demonstrate ++ unary operator overloading.
2
3  #include <iostream>
4  using namespace std;
5
6  class Test{
7      int a,b,c;
8      public:
9          void read_data(int x, int y, int z){
10              a = x;
11              b = y;
12              c = z;
13          }
14          void operator ++(){
15              a++;
16              b++;
17              c++;
18          }
19          void display(){
20              cout << "a = " << a << endl;
21              cout << "b = " << b << endl;
22              cout << "c = " << c << endl;
23          }
24  };
```

```
26  int main(){
27      Test obj;
28      obj.read_data(10,20,30);
29      cout<<"Before incrementing: "<<endl;
30      obj.display();
31      ++obj;
32      cout<<"After incrementing: "<<endl;
33      obj.display();
34      return 0;
35  }
```

Output:

```
PS C:\Users\KIIT01\Desktop\programming> cd "c:\Users\KIIT01\Desktop\programming\OOP\20 october"
_q2_operator_ovrldng } ; if ($?) { .\q2_operator_ovrldng }
Before incrementing:
a = 10
b = 20
c = 30
After incrementing:
a = 11
b = 21
c = 31
PS C:\Users\KIIT01\Desktop\programming\OOP\20 october>
```

Question 5:

```
1  // Write a program to demonstrate * binary operator overloading.
2
3  #include <iostream>
4  using namespace std;
5
6  class Test{
7      int a,b,c;
8      public:
9          void read_data(int x, int y, int z){
10              a = x;
11              b = y;
12              c = z;
13          }
14          void operator *(Test obj){
15              a = a * obj.a;
16              b = b * obj.b;
17              c = c * obj.c;
18          }
19          void display(){
20              cout << "a = " << a << endl;
21              cout << "b = " << b << endl;
22              cout << "c = " << c << endl;
23          }
24  };
```

```
26  int main(){
27      Test obj1, obj2;
28      obj1.read_data(10,20,30);
29      obj2.read_data(2,3,4);
30      cout<<"Before multiplication: "<<endl;
31      obj1.display();
32      obj2.display();
33      obj1 * obj2;
34      cout<<"After multiplication: "<<endl;
35      obj1.display();
36      return 0;
37  }
```

Output:

```
PS C:\Users\KIIT01\Desktop\programming> cd "c:\Users\KIIT01\Desktop\programming\OO
f ($?) { .\q3_binary_overloading }
Before multiplication:
a = 10
b = 20
c = 30
a = 2
b = 3
c = 4
After multiplication:
a = 20
b = 60
c = 120
PS C:\Users\KIIT01\Desktop\programming\OOP\20 october>
```