

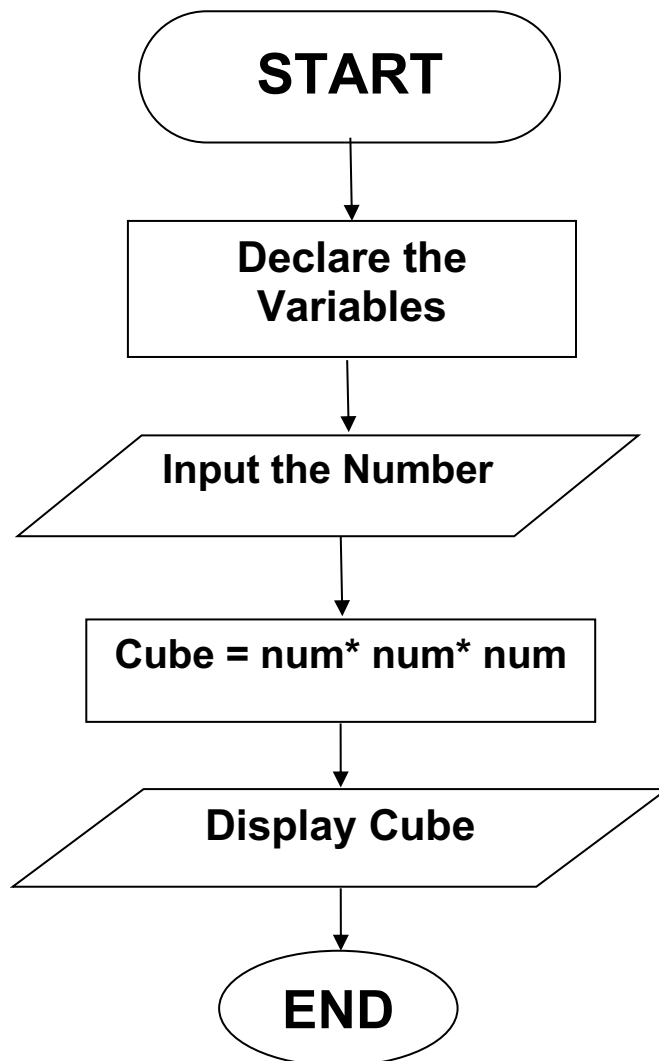
EXPERIMENT NUMBER – Practical 6.1

STUDENT’S NAME – RAJDEEP JAISWAL
STUDENT’S UID – 20BCS2761
CLASS AND GROUP – CSE 26 - B
SEMESTER – 2nd

TOPIC OF EXPERIMENT – POLYMORPHISM

AIM OF THE EXPERIMENT - WAP to calculate and display cube of an integer and float variable using function overloading.

FLOWCHART/ ALGORITHM -



PROGRAM CODE -

```
9 #include <iostream>
10
11 using namespace std;
12
13 int cube(int );
14
15 float cube(float);
16
17 int main() {
18     int a = 7;
19     float b = 7.5;
20
21     cout<< "Cube of integer number " << a << " is " << cube(a) <<endl;
22
23     cout<< "Cube of float number " << b << " is " << cube(b) <<endl;
24
25     return 0;
26 }
27
28 int cube(int x) {
29     return x*x*x;
30 }
31
32 float cube(float y){
33     return y*y*y;
34 }
35
36
37
38
39
40
41 }
```

OUTPUT -

```
Cube of integer number 7 is 343
Cube of float number 7.5 is 421.875

...Program finished with exit code 0
Press ENTER to exit console.
```

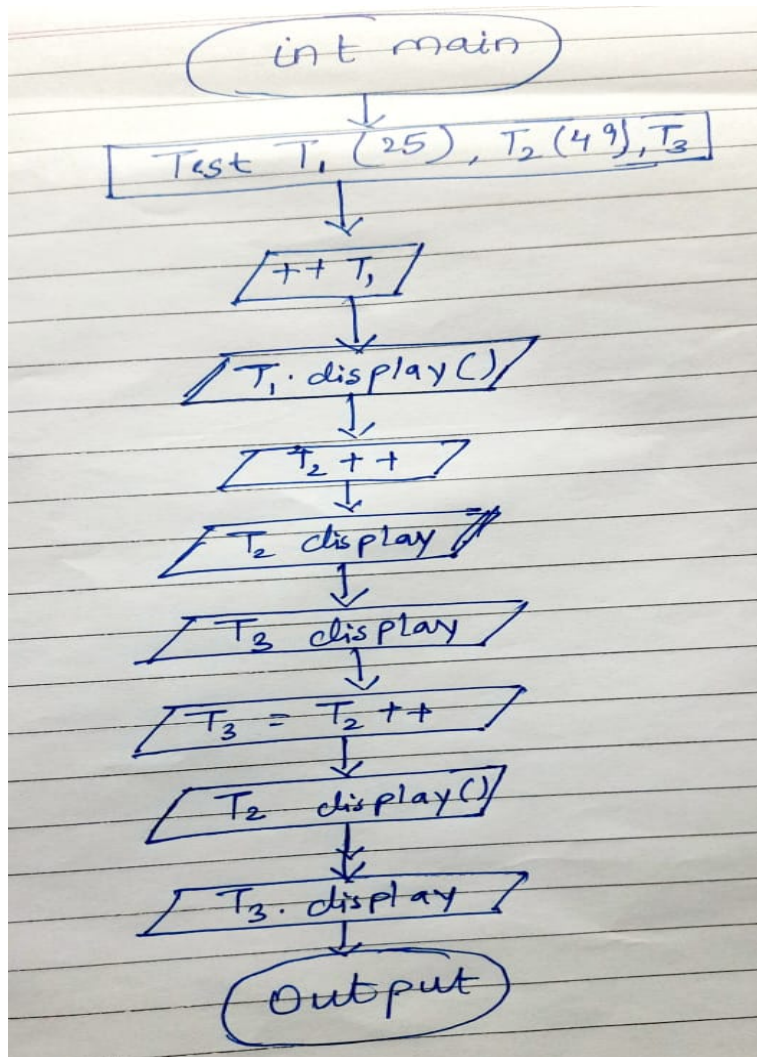
EXPERIMENT NUMBER – Practical 6.2

STUDENT’S NAME – RAJDEEP JAISWAL
STUDENT’S UID – 20BCS2761
CLASS AND GROUP – CSE 26 - B
SEMESTER – 2nd

TOPIC OF EXPERIMENT – POLYMORPHISM

AIM OF THE EXPERIMENT - WAP to create a class Pairs. Objects of type Pairs can be used in any situation where ordered pairs are needed. Our Task is to overload operator >> and << so that objects of class Pairs are to be input and output in the form (5,3) (5,-6) (-5,6) or (-5,-3). There is no need to implement any constructor/method .

FLOWCHART/ ALGORITHM -



PROGRAM CODE -

```
8
9 #include <iostream>
10
11 using namespace std;
12
13 class Test {
14
15 private:
16
17 int num;
18
19
20
21 public:
22
23 Test() {
24
25 num = 0;
26
27 }
28
29 Test(int n) {
30
31 num = n;
32
33 }
34
35 void display() {
36
37 cout<< "Number: " <<num<<endl;
38
39 }
40
41 Test operator++ () {
42
43 ++num;
44
45 return Test(num);
46
47 }
48
49 Test operator++( int ) {
50
51 Test t(num);
52 ++num;
53
54 return t;
55
56 }
57
58 };
59
60 int main() {
61
62 Test T1(25), T2(49), T3;
63
64 ++T1;
65
66 T1.display();
67
68 T2++;
```

```
T2.display();  
  
T3.display();  
    T3 = T2++;  
T2.display();  
T3.display();  
  
return 0;  
  
}
```

OUTPUT :-

```
Number: 26  
Number: 50  
Number: 0  
Number: 51  
Number: 50  
  
...Program finished with exit code 0  
Press ENTER to exit console.□
```



LEARNING OUTCOMES

- Identify situations where computational methods would be useful.
- Approach the programming tasks using techniques learnt and write pseudo-code.
- Choose the right data representation formats based on the requirements of the problem.
- Use the comparisons and limitations of the various programming constructs and choose the right one for the task.

EVALUATION COLUMN (To be filled by concerned faculty only)

Sr. No.	Parameters	Maximum Marks	Marks Obtained
1.	Worksheet Completion including writing learning objective/ Outcome	10	
2.	Post Lab Quiz Result	5	
3.	Student engagement in Simulation/ Performance/ Pre Lab Questions	5	
4.	Total Marks	20	