EXPERIMENT 10

(1)
```
#include <iostream>
using namespace std;
template <class T>
T sumArray (T arr [], int n)
{
  T sum = 0;
for (int i=0; i<n; i++)
  sum += arr [i];
  return sum;
}
int main ()
{
int intarr [5] = {1,2,3,4,5};
float floarr [5] = {1.2, 2.3, 3.4, 4.5, 5.0};
double arr [5] = {10.5, 20.5, 30.5, 40.5, 50.5};
cout << "Sum of integer array :" << sumArray
                    (intarr, 5) << endl;
cout << "Sum of float array :" << sumArray
                    (floatarr, 5) << endl;
cout << "Sum of double array :" << sumArray
                    (doublearr, 5) << endl;
return 0;
}
```

Output : Sum of integer array : 15
Sum of float array : 17
Sum of double array : 157.5

②
```
#include <iostream>
#include <string>
using namespace std;
template <class T>
T square (T x)
{
 return x*x;
}
template <>
string square <string> (string s)
{
 return s+s;
}
int main()
{
 int i = 5;
 string str = "Hi Anushka";
cout << " Square of integer :" << square(i)<<endl;
cout << " Square of string :" << square (str)<<endl;
 return 0;
}
```

Output : Square of integer : 59
Square of string : Hi Anushka Hi Anushka

③
```
#include <iostream>
using namespace std;
template <class T>
class calculator
{
```

```
T num1, num2;
public:
  Calculator (T n1, T n2)
  {
     num1 = n1;
     num2 = n2;
}
void display()
  {
cout<<"Numbers:"<<num1<<"and"<<num2
                              <<endl;
cout<<"Addition:"<<num1+num2<<endl;
cout<<"Substraction:"<<num1-num2<<endl;
cout<<"Multiplication:"<<num1*num2<<endl;
cout<<"Division:"<<num1/num2<<endl;
  }
};
int main()
{
 Calculator<int> int calc (10,5);
  cout<<"Integer Calculator:\n";
  int calc.display();
Calculator <float> float calc (5.5,2.2);
cout<<"\nfloat Calculator:\n";
float calc.display();
  return 0;
}
```

Output : Integer Calculator :
Numbers : 10 and 5
Addition : 15

Substraction : 5
Multiplication : 50
Division 2

Float calculator :
Numbers : 5.5 and 2.2
Addition : 7.7
Substraction : 3.3
Multiplication : 12.1
Division : 2.5

(4)
```
#include <iostream>
using namespace std;
template <class T>
class stack
{
  T stack[10];
  int top;
public:
 Stack()
 {
  top = -1;
 }
void push(T item)
 {
  if (top == 9)
cout << "Stack Overflow!" << endl;
else
  stack[++top] = item;
}
void display()
```

```
{
if (top == -1)
cout << "Stack is empty." << endl;
else
{
cout << "Stack Elements :"
for (int i = 0; i <= top; i++)
 cout << stack [i] << " ";
 cout << endl;
}
}
};
int main()
{
 Stack <int> S;
  S.push (10);
  S.push (20);
  S.push (30);
  S.display();
  S.pop();
  S.display();
  return 0;
}
```

12/11

Output : Stack elements : 10 20 30
Popped : 30
Stack elements : 10 20

× ——— × ——— ×