Question –

Implement Bubble Sort using templates.

Code –

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

template <typename T>
void bubbleSort(T arr[], int n)
{
    for (int i = 0; i < n - 1; ++i)
        for (int j = 0; j < n - i - 1; ++j)
            if (arr[j] > arr[j + 1])
                swap(arr[j], arr[j + 1]);
}

template <typename T>
void printArray(T arr[], int n)
{
    for (int i = 0; i < n; ++i)
        cout << arr[i] << " ";
    cout << endl;
}

int main()
{
    int intArr[] = {5, 2, 9, 1, 5, 6};
    int n1 = sizeof(intArr) / sizeof(intArr[0]);
    bubbleSort(intArr, n1);
    printArray(intArr, n1);

    double doubleArr[] = {3.2, 1.5, 4.8, 2.9};
    int n2 = sizeof(doubleArr) / sizeof(doubleArr[0]);
    bubbleSort(doubleArr, n2);
    printArray(doubleArr, n2);

    return 0;
}
```

Output –

```
1 2 5 5 6 9
1.5 2.9 3.2 4.8
```

Question –

Implement Stack operation using templates

(i) Push
(ii) Pop
(iii) Display Stack contents

Code –

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

template <typename T>
class Stack
{
    T *arr;
    int top;
    int capacity;

public:
    Stack(int size)
    {
        capacity = size;
        arr = new T[capacity];
        top = -1;
    }

    ~Stack()
    {
        delete[] arr;
    }

    void push(T value)
    {
        if (top == capacity - 1)
```

```
        {
            cout << "Stack Overflow" << endl;
            return;
        }
        arr[++top] = value;
    }

    void pop()
    {
        if (top == -1)
        {
            cout << "Stack Underflow" << endl;
            return;
        }
        --top;
    }

    void display()
    {
        if (top == -1)
        {
            cout << "Stack is empty" << endl;
            return;
        }
        for (int i = top; i >= 0; --i)
            cout << arr[i] << " ";
        cout << endl;
    }
};

int main()
{
    Stack<int> s(5);

    s.push(10);
    s.push(20);
    s.push(30);
    s.display();

    s.pop();
    s.display();

    s.push(40);
    s.push(50);
    s.push(60);
```

```
    s.push(70);
    s.display();

    return 0;
}
```

Output –

```
30 20 10
20 10
Stack Overflow
60 50 40 20 10
```