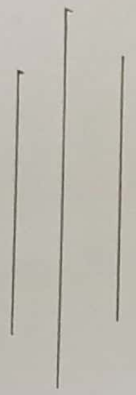
KATHMANDU UNIVERSITY

DHULIKHEL, KAVRE



A

Lab Report On

Object oriented Programming (COMP116)

Lab Report No:

Submitted by:

Ashraya Kadel

CE I/II

Rollno: 25

Submitted to:

Rajani Chulyadyo

Department of Computer Science and Engineering

SUBMISSION DATE: 24 / 12 / 2023

Q.17: Implement bubble sort algorithm using function template.

Ans:

*) Source Code

```
#include <iostream>
using namespace std;
template <typename T>
T sort (T A[], int n)
{
   int i, j, temp;
   T temp;
   for (i=0; i<n-1; i++)
   {
     for (j=0; j<n-1-i; j++)
     {
       if (A[j] > A[j+1])
       {  temp = A[j];
          A[j] = A[j+1];
          A[j+1] = temp; } } }
int main()
{ int A[9] = {12, 31, 24, 23, 234, 13, 2, 41};
  sort (A, 9);
  for (int i=0; i<9; i++) { cout << A[i] << endl; }
}
```

(Q.2): Convert ArrayQueue and IQueue into generic classes to enable for all datatype.

Ans

(*) Source code:

```
#include <iostream>
using namespace std;
template <typename T>
Class IQueue
{
   public:
      virtual ~IQueue(){}
      virtual bool insert (T element) = 0;
      virtual bool remove (T &element) = 0;
      virtual bool front (T &element) = 0;
      virtual bool rear (T &element) = 0;
};
class ArrayQueue: public IQueue
{
  private:
     int topindex;
     int size;
     T *data;
  public:
    ArrayQueue (int size): topindex(-1), size(size),
                  data (new T[size]) {}
```



```
bool insert (T element)
{
   if (topindex < size-1)
   {  topindex++;
      data[topindex] = element;
      return true; } else {return false}
}

bool remove (T &element)
{
   if (topindex >= 0)
   {  int i;
      for (i=0; i< topindex; i++)
      {  T temp;
         temp = data[i+1];
         data[i] = data[i+1];
      }
      topindex--;
      return true; } else {return false;}
   }

bool front (T &element)
{  if (topindex >= 0)
   { element = data[0];
     return true; } else { return false; }

bool rear (T &element)
{  if (topindex >= 0)
     { element = data[topindex];
       return true; } else {return false; }
};
```

```
int main()
{
    IQueue <int> * I = new ArrayQueue (10);
    I -> insert (5);
    I -> insert (6);
    T element;
    I -> rear (element);
    cout << "Last element is " << element << endl;
    I -> front (element);
    cout << " Front element is " << element << endl;
}.
```