```cpp
// Question_2.cpp : This file contains the 'main' function. Program execution begins and ends there.
//// /
//------------------------------------------------------------------------------------------------
//Name                          Sai Chaitanya Kilambi
//Course                        CPSC 131 Data Structures, Fall, 2022
//Assignment                    No.8 question:2
//Due date                      10/26/2022
// Purpose:
// This program stores the data in ordered linked list from a text file and displays the data in order
//-------------------------------------------------------------------------------------------------
// list of libraries
//
//importing the required libraries


#include <iostream>
#include<string>
#include<fstream>

using namespace std;
template <class T>
class ORDER
{
private:
    struct node
    {
        T info;
        node* next;
    };
    node* order;
public:
    ORDER() { order = NULL; }// constructore
    bool emptyOrder()
    {
        return (order == NULL) ? true : false;
    }
    void pushOrder(T x)//
    {
        //insert x in the list and keep the list sorted
        node* r = new node; r->info = x;
        r->next = NULL;
        //find the insertion place;
        node* p = order; node* q = order;
        if (order == NULL)
            order = r;
```

```
45          else
46          {
47              while (p != NULL && x > p->info)
48              {
49                  q = p; p = p->next;
50              }
51              if (p == q)
52              { //insert in front
53                  r->next = p; order = r;
54              }
55              else
56              { //insert at the rear
57                  r->next = p; q->next = r;
58              }
59          }
60      }
61      void displayOrder()
62      {
63          node* p = order;
64          while (p != NULL)
65          {
66              cout << p->info << "-->"; p = p->next;
67          }
68          cout << "NULL\n";
69      }
70      T popOrdere()
71      {
72          //return the info of the first node and then
73          //delete that node
74          T popedElement;
75          node* p = order;
76          popedElement = p->info;
77          order = p->next;
78          delete p;
79          return popedElement;
80      }
81  };
82  int main()
83  {
84      string presidents[6];
85      std::fstream f;
86      //opening the file to only read
87      f.open("data.txt", std::ios::in);
88
89      // copying the file data into an array
90      for (int i = 0; i < 6; i++) {
91          f >> presidents[i];
92      }
93          //display a in sorted form
```

```
 94          ORDER<string> ord;
 95          for (int i = 0; i < 6; ++i)
 96              ord.pushOrder(presidents[i]);
 97          //display the list
 98          cout << "This is the ordered list\n";
 99          ord.displayOrder();
100  }
```