

Team 6
Nathan Lai
Danny Nguyen

Source Code

Lab3.cpp

```
/*
    CECS 282 Lab 3: Bubble Sort
    Team 6: Nathan Lai and Danny Nguyen
*/
#include<iostream>
#include<fstream>

using namespace std;

//PROBLEM 1 AND 2
//Reads data from a text file and adds it to array using pointers
int readData(int * &arr){
    int size;
    ifstream inputFile;
    inputFile.open("data.txt");
    inputFile >> size;
    for(int i = 0; i < size; i++){
        inputFile >> *(arr + i);
    }
    inputFile.close();
    return size;
}

//PROBLEM 1 AND 2
//Prints the array to console
void writeToConsole(int *arr, int last){
    cout << "Sorted array:";
    for(int i = 0; i < last; i++){
        cout << " " << *(arr + i);
    }
    cout << endl;
}

//PROBLEM 1
//Bubble sort algorithm sorting array from pointer
void bsort(int *arr, int last){
```

Team 6

Nathan Lai

Danny Nguyen

```
    bool sorted = false;
    while(!sorted){
        sorted = true;
        for(int i = 0; i < last - 1; i++){
            if(*(arr + i) > *(arr + i + 1)){
                sorted = false;
                int temp = *(arr + i);
                *(arr + i) = *(arr + i + 1);
                *(arr + i + 1) = temp;
            }
        }
    }
}
```

//PROBLEM 2

//Boolean function that returns true if two values are in ascending order

```
int ascending(int first, int second){
    if(first <= second){
        return true;
    }
    return false;
}
```

//PROBLEM 2

//Boolean function that returns true if two values are in descending order

```
int descending(int first, int second){
    if(first >= second){
        return true;
    }
    return false;
}
```

//PROBLEM 2

//Bubble sort function, that can now sort in both ascending and descending order through the use of function pointers

```
void bubble_sort(int *array, int size, int(*order)(int, int)){
    bool sorted = false;
    while(!sorted){
        sorted = true;
        for(int i = 0; i < size - 1; i++){
            if(!(*order)(*(array + i), *(array + 1 + i))){
                sorted = false;
            }
        }
    }
}
```

Team 6
Nathan Lai
Danny Nguyen

```
        int temp = *(array + i);
        *(array + i) = *(array + i + 1);
        *(array + i + 1) = temp;
    }
}
}
```

//MAIN: used for testing methods

```
int main(){
    //PROBLEM 1: Basic ascending order with bubble sort
    cout << "Testing bsort" << endl;
    int * arr1 = new int;
    cout << "Reading data.txt..." << endl;
    int size1 = readData(arr1);
    cout << "Performing bubble sort..." << endl;
    bsort(arr1, size1);
    writeToConsole(arr1, size1);

    //PROBLEM 2: Bubble sort with both ascending and descending order through function pointers
    cout << endl << "Testing bubble_sort with descending function pointer" << endl;
    int * arr2 = new int;
    cout << "Reading data.txt..." << endl;
    int size2 = readData(arr2);
    cout << "Performing bubble sort..." << endl;
    bubble_sort(arr2, size2, descending);
    writeToConsole(arr2, size2);

    cout << endl << "Testing bubble_sort with ascending function pointer" << endl;
    int * arr3 = new int;
    cout << "Reading data.txt..." << endl;
    int size3 = readData(arr3);
    cout << "Performing bubble sort..." << endl;
    bubble_sort(arr3, size3, ascending);
    writeToConsole(arr3, size3);

    return 0;
}
```

Runtime Output

```
D:\CSULB Code\CECS 282 Lab\Lab3\Lab3.exe
Testing bsort
Reading data.txt...
Performing bubble sort...
Sorted array: 1 2 3 4 5 6 7 8 9

Testing bubble_sort with descending function pointer
Reading data.txt...
Performing bubble sort...
Sorted array: 9 8 7 6 5 4 3 2 1

Testing bubble_sort with ascending function pointer
Reading data.txt...
Performing bubble sort...
Sorted array: 1 2 3 4 5 6 7 8 9

-----
Process exited after 0.1527 seconds with return value 0
Press any key to continue . . .
```