

BCS230

Lab8 Handout and Assignment

This lab handout provides two tasks that students are required to attempt during lab session.

Lab objectives are for students to practice with:

- Arrays and Arrays of Objects.
- Classes and Objects.
- Searching and Sorting.

Task1:

Charge account validation

Modify the provided shell code to create a program that lets the user enter a charge account number. The program should determine if the number is valid by checking for it in the following list:

5658845,	4520125,	7895122,
8777541,	8451277,	1302850,
8080152,	4562555,	5552012,
5050552,	7825877,	1250255,
1005231,	6545231,	3852085,
7576651,	7881200,	4581002,

Initialize a one-dimensional array with these values. Then use a simple linear search to locate the number entered by the user. If the user enters a number that is in the array, the program should display a message saying that number is valid. If the user enters a number not in the array, the program should display a message indicating it is invalid.

Complete:

ToDo 1: Read the input values.

ToDo 2: Search for a specific value

ToDo 3: Print if found

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

bool searchList(long [], int, long); // Function prototype

const int SIZE = 18;
int main()
{
    long accounts[SIZE] =
        {5658845, 4520125, 7895122, 8777541,
         8451277, 1302850, 8080152, 4562555,
         5552012, 5050552, 7825877, 1250255,
         1005231, 6545231, 3852085, 7576651,
         7881200, 4581002, };
    long acctNum;

    cout << "Please enter a 7-digit account number: ";
    //ToDo 1: Read the input value

    // ToDo 2: check if the account number is valid
    // ToDo 3: print if found
    return 0;
}

bool searchList(long list[], int numElems, long value)
{
    bool found = false;

    // ToDo provide implementation of the linear search here
    return found;
}
```

Task2:

Modify the selectionSort function presented in PPT lecture notes so it sorts an array of strings instead of an array of ints. Test the function as stated in the shell code below.

Complete:

ToDo 1: Using selectionSort complete the function implementation

ToDo 2: Convert the string to uppercase

ToDo 3: Print all elements in the array.

```
#include<iostream>
#include<string>
#include<cctype>
using namespace std;

// Function prototype
void selectionSort(string [], int);
string upperCaseIt(const string);
void displayNames(const string [], int);

int main(){
    const int SIZE = 20;
    // Set up the array of strings, in this case names
    string name[SIZE] =
        { "Collins, Bill", "Smith, Bart", "Michalski, Joe", "Griffin, Jim",
          "Sanchez, Manny", "Rubin, Sarah", "Taylor, Tyrone", "Johnson, Jill",
          "Allison, Jeff", "Moreno, Juan", "Wolfe, Bill", "Whitman, Jean",
          "Moretti, Bella", "Wu, Hong", "Patel, Renee", "Harrison, Rose",
          "Smith, Cathy", "Conroy, Pat", "Kelly, Sean", "Holland, Beth" };

    // Call a function to sort the strings
    selectionSort(name, SIZE);
    // Call a function to display the (now sorted) strings
    cout << "The names in sorted order are: \n\n";
    displayNames(name, SIZE);

    return 0;
}

void selectionSort(string name[], int numStrings ){
    int startScan, minIndex;
    string minValue;
    // ToDo 1: Sort the strings
}

string upperCaseIt(const string stringIn){
    string s = stringIn; // Local copy of stringIn we'll uppercase
    //ToDo 2: convert the string to uppercase
    return s;
}

void displayNames(const string name[], int numNames){
    //ToDo 3: Print all elements (numNames) of the string array (name)
}
```

Lab8 – Assignment.

What to hand-in

Problem:

Write a program that can be used by a ski resort to keep track of the local snow conditions for one week. It should have seven-element array of structures or class objects, where each structure or object holds a date and the number of inches of snow in the base on that date. The program should have the user input the name of the month, the starting and ending date of the seven-day period being measured, and then the seven base snow depths.

The program should then sort the data in the ascending order by base depth and display the results (two functions one for sorting and one for displaying the results). Here is the beginning of a sample report.

Snow Report December 12-18

Date	Base
13	42.3
12	42.5
14	42.8

Submission instructions

- Test your code before submission.
- You should have either a class or a structure.
- Add comments to explain your solution.
- Due date is Wed April/03/2018 @ 11:59pm
- You are allowed to submit late with -5% per day.
- Submission will not be available after Friday April/06/2018 @ 11:59pm.
- If you have more than one submission the late will be graded.
- Check rubric.