

COP 3014 Honors: Spring 2017

Homework 5

Total Points: 150

Due: Thursday 03/09/2017 11:59:59 PM

1 Objective

The purpose of this assignment is to test your familiarity with C++ functions and arrays. You'll also need to do fundamental algorithmic error checking. These programs have to be tested on `linprog` before they are turned in.

Email your files `arrayops.cpp` and `norm.cpp` to `jayarama@cs.fsu.edu`

2 Program 1 - Array Operations

For this program, we're going to perform a range of standard array operations to familiarize yourself once and for all, with arrays and how they work. You're required to write functions to insert and delete elements from an array, sort the array and search for elements in the array. We're also going to write a command line menu to do these tasks repeatedly.

Please make sure you conform to the following requirements:

1. Create a global constant integer called `CAPACITY` and set it to 100 (5 points)
2. Use the class examples and exercises to write functions to initialize the array by reading values from the user, and to print the array. These functions take the array and its current size as parameters. You can use the code in the examples. (5 points)
3. Write a function called `insert` that takes the array, a number, a position and the current size of the array as parameters. Insert the number at the given position. (15 points)
4. Write a function called `remove` that takes the array, a number and the current size of the array, and removes the first instance of the number from the array. (15 points)
5. Write a function called `sort` that takes the array and its current size as parameters and sorts the array in ascending order. You need to use insertion sort, which is given below. (15 points)
6. Write a function called `binarysearch` that accepts the array, a number, the start position, the end position and the array's current size as parameters and returns the position of the number in the array. You need to use binary search algorithm, which will be discussed in class. (20 points)
7. In the main function, have the user initialize the array. Then, write a command line menu with the following options:
 - Print
 - Insert

- Remove
- Sort
- Search
- Exit

Call the appropriate functions when the user chooses a particular option. Print an error message and continue if the user enters a wrong option. (20 points)

8. Please make sure your code is appropriately commented. (5 points)

Algorithm for insertion sort

```

loop i from 1 to Length(A)
  x = A[i]
  j = i - 1
  loop as long as j >= 0 and A[j] > x
    A[j+1] = A[j]
    j = j - 1
  end loop
  A[j+1] = x
end loop

```

2.1 Sample Run

```

Enter the number of elements you want to enter (Max 100): 5
Enter 5 numbers
2.5 6 -7 10.7 0.8

```

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element
6. Exit

```

Enter your option: 1
Enter the number: 3.1
Enter the position: 2
Element Inserted.

```

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element
6. Exit

```

Enter your option: 3

```

```

The array is:
2.5 6 3.1 -7 10.7 0.8

```

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array

5. Search for an element
6. Exit
Enter your option: 4

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element
6. Exit
Enter your option: 3

The array is:
-7 0.8 2.5 3.1 6 10.7

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element
6. Exit
Enter your option: 5

Enter the element: 6
The element is in position 4

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element
6. Exit
Enter your option: 5

Enter the element: 8
Element not found.

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element
6. Exit
Enter your option: 2

Enter the element: 0.8
Element deleted.

1. Insert an element
2. Remove an element
3. Print an element
4. Sort the array
5. Search for an element

```
6. Exit
Enter your option: 3
```

```
The array is:
-7 2.5 3.1 6 10.7
```

3 Program 2 - Matrix Norms

The Forbenius norm of a matrix is defined as square root of the sum of the squares of all the elements of a matrix. Write a C++ program to compute the Forbenius norm.

Please make sure you conform to the following requirements:

1. Declare global constant integers for ROWCAP and COLCAP. Set both to 100 (5 points)
2. Use the class examples and exercises to write functions to initialize the matrix by reading values from the user, and to print the matrix. These functions take the matrix and its current number of rows and columns as parameters. You can use the code in the examples. (5 points)
3. Write a function called `findnorm` that accepts the matrix, its current number of rows and columns and return its Forbenius norm. (20 points)
4. In the main function, initialize the matrix, read in its values and compute and print its norm. (15 points)
5. Please make sure your code is commented (5 points).

3.1 Sample Run

```
Enter the number of rows: 3
Enter the number of columns: 3
Enter the matrix:
-3 5 7
2 6 4
0 2 8
```

```
The Forbenius Norm is: 14.3875
```

4 Generic Guidelines

1. Please make sure you're only using the concepts already discussed in class. That is, please try and restrict yourself to input/output statements, variables, selection statements and loops, functions, and arrays.
2. The first program is worth 100 points. The second program is worth 50 points.
3. Please make sure that you're conforming to specifications (program name, print statements, expected inputs and outputs etc.).
4. Please make sure your code is readable.
5. Please make sure you've compiled and run your program before you turn it in. Compilation errors can be quite costly.

6. The functions you have been asked to write are very standard things. Which means, you will find them on the Internet. I will be watching for this. so please do not get into trouble by copying stuff off the Internet or your friends.