

HomeWork 2: Dynamically Allocating Memory

BASIC PART: Sort and Modify a Dynamically Allocated Array

(25 pts)

Specification:

In this assignment, you'll dynamically allocate memory for a number of integer variables to be read from the keyboard. You'll be able to find mean, median and standard deviation of the integers. You need to implement the following steps in the attached test file (**SortModifyArray_test.c**).

1. You will dynamically allocate memory to read into a number of elements. At first, you have to determine how many numbers (here, 'n') to be read (The program asks you to type 'n'). Next, you need to dynamically allocate memory for 'n' integers. As you see, 'x' is an integer pointer which needs to dynamically allocate memory to read 'n' integers into it from the keyboard. 2 pts
2. Read **n** integers into the allocated block using **scanf**. 2 pts
3. Complete the **printArray** function to display 'n' integers. 2 pts
void printArray(int *array, const int size);
4. Complete the following functions to find the average (mean), median and the standard deviation of 'n' integers.
 - a) double findMean(int *array, const int size); 2 pts
 - b) double findMedian(int *array, const int size); 4 pts
 - c) double findStandardDeviation(int *array, const int size, double average); 6 pts

Use the attached cheat sheet "**SimpleStats.pdf**" to find mean, median and standard deviation of numbers. **If the user types only one integer, the standard deviation should be set to 0.**

5. Note, you need to have the array sorted to find the median value. So, you have to write the following **sortArray** function:
void sortArray(int *array, const int size); 3pts
6. The element in the sorted array should be arranged in the ascending order. Inside **sortArray** function, you will call the **swapElements** function as mentioned below to sort array elements. So, you also need to complete the **swapElement** function.
void swapElements(int *a, int *b); 2 pts
7. Lastly, you need to free the allocated memory. 2pts

Note:

As you may need to use the math library in this assignment (**#include <math.h>**), you have to explicitly link to it. It is not automatically linked to along with the rest of the standard C library. You would accomplish this by putting **-lm** at the end of the command. For example:

```
gcc -o foo foo.c -lm
```

Below is a screenshot on how your output will look like once you implement all the steps:

```

syasmin@cscd-linux01:~/HW2$ ./SortModifyArrayBasic
This is the basic part of the program that asks the user to type the number of integers, i.e., 'n'. Next, allocate memory
for 'n' integers, read the values of 'n' integers into the allocated memory using scanf; next, find the mean, median a
nd average of 'n' integers. Lastly, the allocated memory needs to be freed.

Read using scanf how many integers you would like to type:
5
Please type 'n' integers:
6
4
3
2
1
Displaying the numbers:
6 4 3 2 1
Mean of the numbers is: 3.200000
Median of the numbers is: 3.000000
Standard deviation of the numbers is: 1.923538
syasmin@cscd-linux01:~/HW2$

```

BONUS PART: Sort and Modify a Dynamically Allocated Array

5 pts

Specification: Modify the basic part so that instead of reading a fixed number of integers, the user can keep on typing as many integers as they would like to. The program will keep on displaying the integers typed so far as well as keep on finding the cumulative mean, median and average of the integers until ctrl+d is pressed from the keyboard. The following output was generated for a particular run of the program.

```

syasmin@cscd-linux01:~/HW2$ ./SortModifyArrayBonus
Bonus version of the program. This program can read as many integers as the user types and keep on displaying the cumula
tive mean, median and average of teh numbers typed so far. Lastly, the allocated memory needs to be freed.
Please type as many integers as you would like to:
Typing the number:
6
Displaying the numbers typed so far:
6
Mean of the numbers is: 6.000000
Median of the numbers is: 6.000000
Standard deviation of the numbers is: 0.000000
Typing the number:
4
Displaying the numbers typed so far:
6 4
Mean of the numbers is: 5.000000
Median of the numbers is: 5.000000
Standard deviation of the numbers is: 1.414214
Typing the number:
3
Displaying the numbers typed so far:
6 4 3
Mean of the numbers is: 4.333333
Median of the numbers is: 4.000000
Standard deviation of the numbers is: 1.527525
Typing the number:
2
Displaying the numbers typed so far:
6 4 3 2
Mean of the numbers is: 3.750000
Median of the numbers is: 3.500000
Standard deviation of the numbers is: 1.707825
Typing the number:
1
Displaying the numbers typed so far:
6 4 3 2 1
Mean of the numbers is: 3.200000
Median of the numbers is: 3.000000
Standard deviation of the numbers is: 1.923538
Typing the number:
syasmin@cscd-linux01:~/HW2$

```

Submission (BASIC PART and BONUS PART):

A zip file containing:

BASIC PART:

- Your C code named **SortModifyArrayBasic.c**
- Output capture named **SortModifyArrayBasicHW2out.pdf** – containing at least **3** different runs.

BONUS PART (optional):

- Submit the complete C file as “**SortModifyArrayBonus.c**” via canvas.
- Output capture named **SortModifyArrayBonusHW2out.pdf** – containing at least **3** different runs.

Name your zip file with your last name first letter of your first name **HW2.zip** (ex: **yasminsHW2.zip**)

Submission deadline is: 11:59 pm, Friday, November 6. No late submission will be considered.