

**EE 2372 Software Design I**  
**Assignment #5: Functions Programs**  
**Due Date: Monday November 13 at 11:59 PM**

**Objectives:**

- 1) Learn to write functions that operate on arrays
- 2) Learn to manipulate strings using functions

**Description**

This programming assignment will have you write two programs.

1. The first program will be a modification of Assignment 3. Just as in Assignment 3, the user will input up to 20 numbers, and these numbers will be stored in an array. Write functions called arrmin, arrmax, and arraver that will take as input an array, and return the minimum value of the array, the maximum value of the array, and the average of the array respectively. After the user enters the values of the array, use the functions to compute the min, max, and average of the array values and print them out.
2. (Prata Chap. 11, Problem 10). Write a function that takes a string as an argument and removes the spaces from the string. Test it in a program that uses a loop to read lines until you enter an empty line. The program should apply the function to each input string and display the result.

**Deliverables:**

- 1) Submit the C source code as an attachment to our TA Pavithra Pochamreddy at ppochamredd@miners.utep.edu. Name your programs arrstats.c and noblanks.c.
- 2) The subject line must be "EE 2372 Assignment 5" (don't include the quotes).

**Scoring:**

Your grade for this assignment will be determined by three criteria. The first criterion determines if your program compiles and runs producing the correct result. The correct result must adhere to what is specified in the **Tasks** section. The second criterion is whether the program follows the interface specification outlined in the **Tasks** section. The third criterion determines if your source code is well documented. Your source code must include (at the top) your name, class section, due date, assigned date, and a small description of your program. For this assignment, each line of code should have a descriptive comment.

<b>Operation/Successful Demonstration</b>	<b>60%</b>
Was a C source file submitted? 10%	
Does the program compile on Linux? 15%	
Does the program run correctly on Linux? 35%	

<b>Adherence to Interface Specification</b>	<b>30%</b>
Does the program adhere to the interface specification in the tasks section? 30%	
<b>Comments</b>	<b>10%</b>
Is the source code well-documented? 10%	
<b>Lateness</b>	<b>10% per day (including weekends and holidays)</b>