ECE 152 Programming for Engineers
Laboratory 7

NOTE:
a) Please submit only C++ source files (*.cpp) to Blackboard.
b) Please put your name, project description, and date on the top of your file as a comment.
c) PLEASE WORK ALONE. If cheating is found, you will get ZERO.

1. (Lastname_Lab7_p1.cpp)
Write a program to input ten integer numbers into an array named `fmax` and determine both the maximum value entered and the index number for the maximum. Your program should contain only one loop. After displaying the numbers of the array, display these two messages:

        The maximum value is : __
        This is element number __ in the list of numbers
Have your program display the correct values in place of the underlines in the messages.

2. (Lastname_Lab7_p2.cpp)
        Write a program to input or initialize the following integer numbers into an array called `grades`: 89,95,72,83,99,54,86,75,92,73,79,75, 82,73,87. Calculate the average of the numbers and use the average to determine the deviation of each value from the average. Store each deviation in an array named `deviation`. Each deviation is obtained as the element value less the average of all the data. Have your program display each deviation alongside its corresponding element from the `grades` array.
        Calculate the variance of the data. The variance is obtained by squaring each individual deviation and dividing the sum of the squared deviations by the number of deviations.

3. (Lastname_Lab7_p3.cpp)
        Write a program that declares three one dimensional arrays named `price`, `quantity`, and `amount`. Each array should be declared in `main( )` and should be capable of holding ten double-precision numbers. The numbers that should be stored in price are 10.64, 14.89,15.21,74.21,23.8,61.26,92.37, 12.73, 2.99, 58.98. The numbers should be stored in `quantity` are 4, 8,17,2,94,61,20,78,55,41. You program should pass these three arrays to a function named `extend()`, which should calculate the elements in the amount array as the product of the corresponding elements in the `price` and `quantity` arrays (for example, `amount[1] = price[1] * quantity[1]`). After `extend()` have put values into the amount array, the array should be display from within `main()`.

4. (Lastname_Lab7_p4.cpp)
    a. Write a function named `findmax()` that finds and displays the maximum values in a two dimensional array of integers. The array should be declared as a 10 row

by 20 column array of integers in `main()` and populated with random numbers between 0 and 100.

  b. Modify the function written above so that it also displays the row and column numbers of the element with the maximum value.

5. (Lastname_Lab7_p5.cpp)

Use a one dimensional array to solve the following problem. Read in 20 numbers, each of which is between 10 and 100, inclusive. As each number is read, print it only if it is not a duplicate of a number already read. Provide for the "worst case" in which all 20 numbers are different. Use the smallest possible array to solve this problem.

6. (Optional. Lastname_Lab7_p6.cpp)

Write a program that simulates the rolling of two dice. The program should use `rand` to roll the first die, and should use `rand` again to roll the second die. The sum of the two values should then be calculated. Note: Since each die can show an integer value from 1 to 6, then the sum of the two values will vary from 2 to 12 with 7 being the most frequent sum and 2 and 12 being the least frequent sums. The following figure shows the 36 possible combinations of the two dice. Your program should roll the two dice 36,000 times. Use a one dimensional array to tally the numbers of times each possible sum appears. Print the results in a tabular format. Also, determine if the totals are reasonable, i.e., there are six ways to roll a 7, so approximately one sixth of all the rolls should be 7.

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12