1. Cut Last Name:

Write a function named "nameSlice" to cut the last name off of a string that contains the user's first and last names.

****

****

****

2.Password Verifier:

Imagine you are developing a software package that requires users to enter their own passwords. Your software requires that users' passwords meet the following criteria:
• The password should be at least six characters long.
• The password should contain at least one uppercase and at least one lowercase letter.
• The password should have at least one digit.
Write a program that asks for a password and then verifies that it meets the stated criteria. If it doesn't, the program should display a message telling the user why.

****

****

****

3. Word Separator:

Write a program that accepts as input a sentence in which all of the words are run together(Without any space), but the first character of each word is uppercase. Convert the

sentence to a string in which the words are separated by spaces and only the first word starts with an uppercase letter. For example the string "StopAndSmellTheRoses." would be converted to "Stop and smell the roses."

****

****

****

4. Drop The Lowest Test Score:

The professor gives "four" exams during the semester in her chemistry class. At the end of the semester she drops each student's lowest test score before averaging the scores. She has asked you to write a program that will read a student's four test scores as input, and calculate the average with the lowest score dropped.

****

****

****

5. Payroll:

Write a program that uses the following arrays:

• empId: an array of seven long integers to hold employee identification numbers. The array should be initialized with the following numbers:
5658845 4520125 7895122 8777541 8451277 1302850 7580489

• hours: an array of seven integers to hold the number of hours worked by each employee

• payRate: an array of seven doubles to hold each employee's hourly pay rate
• wages: an array of seven doubles to hold each employee's gross wages

The program should relate the data in each array through the subscripts. For example, the number in element 0 of the hours array should be the number of hours worked by the employee whose identification number is stored in element 0 of the empId array.

That same employee's pay rate should be stored in element 0 of the payRate array.
The program should display each employee number and ask the user to enter that employee's hours and pay rate. It should then calculate the gross wages for that employee (hours times pay rate) and store them in the wages array. After the data has been entered for all the employees, the program should display each employee's iden- tification number and gross wages.

Input Validation: Do not accept negative values for hours or numbers less than 15.00 for pay rate.

****
****
****

6. 2D TestScores:

Write a program that creates a two-dimensional array initialized with test data. Use any data type you wish. The program should have the following functions:

• getTotal. This function should accept a two-dimensional array as its argument and return the total of all the values in the array.

• getAverage. This function should accept a two-dimensional array as its argument and return the average of all the values in the array.

• getRowTotal. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the total of the values in the specified row.

• getColumnTotal. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a column in the array. The function should return the total of the values in the specified column.

• getHighestInRow. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the highest value in the specified row of the array.

• getLowestInRow. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the lowest value in the specified row of the array.
Demonstrate each of the functions in this program.

*** Assumption: Rows=4 | Columns=5 ***