Problems

1. Get Student GPA

Write a program that calculates the total GPA for a student. The program takes from the user the number of courses he wishes to enter (maximum: 10), the course grades (characters) and the hours corresponding to each grade.
The program should calculate the sum of hours of all the courses using function A.
The program should then convert the character grades into integer grades in the main function as follows:

Grade 'A ' would be 4.0
Grade 'B ' would be 3.0
Grade 'C ' would be 2.0
Grade 'F ' would be 0.0

Finally, the program should calculate the total GPA as follows.
Example:

Integer Grades Array:

4.0	2.0	3.0	4.0	0.0

Hours Array:

3.0	2.0	2.0	2.0	3.0

TotalGPA= (4.0*3.0 + 2.0*2.0 + 3.0*2.0 + 4.0*2.0 + 0.0*3.0)/ 12 (total hours) = 2.5

Function A:
```
int SumofHours(int hours[10],numofGrades)
    //returns the sum of the hours array.
```

Function B:
```
float calculateGPA(int integerGrades[10],int hours[10],
                   int sumofhours,int numofGrades)
```

Input/output should be done in the main function.

Sample Run:

```
Please Enter the number of courses: 5
Please enter the course grade : A
Please enter the credit hours for the course : 3
Please enter the course grade : C
Please enter the credit hours for the course : 2
Please enter the course grade : B
Please enter the credit hours for the course :2
Please enter the course grade : A
Please enter the credit hours for the course: 2
Please enter the course grade : F
Please enter the credit hours for the course : 3

The total GPA is 2.5
```

2. Value Removal in 1D

Given a 1D array and a value, you are required to remove all instances of that value, then save and output the array afterwards.
Note: do not allocate extra space for another array; you must do this within the same array.

Sample Input:

```
4 2 7 10 9 7 6 10 -8 7 9
7
```

Output:

```
4 2 10 9 6 10 -8 9
```

3. Cumulative 2D array

Write a program to take a 2D array of m*n size from user, then outputs in another array the cumulative sum.
Note: m, n will not exceed 10

Sample Run:

```
Enter size of array:
3 2
Enter the array:
1 5
3 4
1 2
The cumulative sum array is:
1 6
9 13
14 16
```

4. Sorting Columns

Write a program to ask the user to input the elements of a 5×5 array. Sort each column into increasing order, and print out the array with sorted columns. You can use either selection or insertion sort.

Sample Run:

```
Enter 5×5 array:
1 5 3 4 2
2 1 4 5 3
4 2 5 3 1
3 4 1 2 5
5 3 2 1 4

Array with sorted columns:
1 1 1 1 1
2 2 2 2 2
3 3 3 3 3
4 4 4 4 4
5 5 5 5 5
```

5. Average of 2D

Write a program that takes from the user a nxm integer array, and then outputs the average and the integers that are larger than or equal to the average of the numbers.
The program uses the following function to calculate the average:

```
float calculateAverage(int arr[][10],int row,int col)
```

Sample Run:

```
please enter the number of rows : 2
Please enter the number of columns : 2
Please Enter the numbers : 4 5 1 3
The Average is 3.25
The numbers above or equal average are :4  5
```

6. Perfect Number

Write C++ function that determines if a given integer is a perfect number. The program should run continuously until the user enters 0.

Hint: A perfect number is a positive integer that is equal to the sum of its positive divisors excluding the number itself.

Sample Input:

```
Enter Number to check for Perfect Number: 6
6 is a perfect number
Enter Number to check for Perfect Number: 36
36 is not a perfect number.
Enter Number to check for Perfect Number: 0
```

7. Students

Write C++ program that records the data of a number of students, including grades for each subject, and calculate the overall grade for each student and average grade for each subject. Hint:

- Assume each subject grade is out 100.
- Assume each student has 6 subjects.
- Subject Data:
    1. Subject name.
    2. Subject Grade.
- Student Data:
    1. ID
    2. Student Name;
    3. Subjects (6 Subjects)
    4. Overall Grade

Sample Run:

```
Enter Number of Students you want to add:  3
Enter ID for Student Number 1:  1
Enter Name for Student Number 1:  Ahmed
Enter Subjects for Student Number 1:
Enter Subject Name for Subject Number 1: Math1
Enter Subject Grade for Subject Number 1:  88
Enter Subject Name for Subject Number 2: English
Enter Subject Grade for Subject Number 2:  80
Enter Subject Name for Subject Number 3: Linguistics
Enter Subject Grade for Subject Number 3:  85
Enter Subject Name for Subject Number 4: Humanities
Enter Subject Grade for Subject Number 4:  80
Enter Subject Name for Subject Number 5: Physics
Enter Subject Grade for Subject Number 5:  78
Enter Subject Name for Subject Number 6: Programming
Enter Subject Grade for Subject Number 6:  88
"************************************************************"
Enter ID for Student Number 1:  2
Enter Name for Student Number 1:  Mona
Enter Subjects for Student Number 1:
Enter Subject Name for Subject Number 1: Math1
Enter Subject Grade for Subject Number 1:  90
Enter Subject Name for Subject Number 2: English
Enter Subject Grade for Subject Number 2:  80
Enter Subject Name for Subject Number 3: Linguistics
Enter Subject Grade for Subject Number 3:  78
Enter Subject Name for Subject Number 4: Humanities
Enter Subject Grade for Subject Number 4:  70
Enter Subject Name for Subject Number 5: Physics
Enter Subject Grade for Subject Number 5:  80
Enter Subject Name for Subject Number 6: Programming
Enter Subject Grade for Subject Number 6:  92
"************************************************************"
Enter ID for Student Number 1:  3
Enter Name for Student Number 1:  Ali
Enter Subjects for Student Number 1:
Enter Subject Name for Subject Number 1: Math1
Enter Subject Grade for Subject Number 1:  90
Enter Subject Name for Subject Number 2: English
Enter Subject Grade for Subject Number 2:  90
Enter Subject Name for Subject Number 3: Linguistics
Enter Subject Grade for Subject Number 3:  88
Enter Subject Name for Subject Number 4: Humanities
Enter Subject Grade for Subject Number 4:  70
Enter Subject Name for Subject Number 5: Physics
Enter Subject Grade for Subject Number 5:  78
Enter Subject Name for Subject Number 6: Programming
Enter Subject Grade for Subject Number 6:  88
```

```
"*************************************************************"
ID  Name  Overall Grade
1  Ahmed   83.16
2  Mona    81.66
3  Ali     84
"*************************************************************"
Subject Name   Average Grade
Math1            89.33
English          83.33
Linguistics      83.66
Humanities       73.33
Physics          78.66
Programming      89.33
```

8. Armstrong Numbers within an Interval

This program asks user to enter two integers and displays all Armstrong numbers between the given intervals.

A positive integer is called an Armstrong number if the sum of cubes of individual digits is equal to that number itself. For example:

153 = (1 * 1 * 1) + (5 * 5 * 5) + (3 * 3 * 3). 153 is an Armstrong number.

12 is not equal to (1 * 1 * 1) + (2 * 2 * 2) = 9 !=12, so 12 is not an Armstrong number

Sample Run:

```
Enter first number: 100
Enter second number: 400
Armstrong numbers between 100 and 400 are:
153
370
371
```

9. Insert in array

Write a program that asks for an index and a number. Then it includes that number at the indicated index of an entered array and moves a position forward (from k to k+1) each element after the selected index and drop the last element.

Sample Run:

```
Enter the 1D array:
7  8  9  1  5  6   8
Enter the number:
0
Enter the index:
2
OUTPUT:
7   8   0   9   1   5   6
```

10. Crop 2D array

Write a program to Crop a 2D 5X5 array by constructing a new 2D array to retain only elements within a specified range.

Sample Run:

```
Enter the 2D array:
3  4  5  6   8
4  5  6  9   0
6   1   2   7   7
8   9   0   6   4
Enter the range for row:
1     3
Enter the range for column:
2     3
OUTPUT:
1   2   7
9   0   6
```

11. Super Ball Competition

In a Super Ball competition, the organizers keep track of each team score after each game to determine who the four finalists in the quad finals are. Each score should be represented as a structure that contains Team Name and Points collected from winning. Write a program to take a list of 16 score data, sort the list and display the final four teams (teams with the highest points).

12. Small Char Frequency

Write a program that reads a set of characters from a text file and prints a table listing the number of occurrences of the of lower-case characters only (‘a’ to ‘z’).

Sample Run:

```
Enter file name: example.txt


CHARACTER        OCCURRENCES
a                20
b                5
c                11
d                0
e                3
f                6
g                0
h                5
…                …
```