Birla Institute of Technology & Science, Pilani
Second Semester 2017-2018, DSA (CS F211)
Lab Assignment #2

Total number of problems = 20

1. Write a program that outputs all possibilities to put + or - or nothing between the numbers 1,2,…,9 (in this order) such that the result is 100. For an example 1 + 2 + 3 - 4 + 5 + 6 + 78 + 9 = 100.

2. Write a user interface driven C program to validate credit card numbers.
   Note: Credit card numbers are validated using following method (called Luhn test)
   The Luhn Formula:

   a. Drop the last digit from the number. The last digit is called **check digit**.
   b. Reverse the numbers
   c. Multiply the digits in odd positions by 2 (if result is greater than 9, sum the digits)
   d. Add all the numbers together
   e. The check digit (the last number of the card) is the amount that you would need to add to get a multiple of 10 (Modulo 10)

   For example: 30130708434187
   1. 3 0 1 3 0 7 0 8 4 3 4 1 8
   2. **8** 1 **4** 3 **4** 8 **0** 7 **0** 3 **1** 0 **3**
   3. **16** 1 **8** 3 **8** 8 **0** 7 **0** 3 **2** 0 **6** (**7** 1 8 3 8 8 0 7 0 3 2 0 6)
   4. 53
   5. 53+7 = 60 (credit card number is valid

3. Most of the people don't like comments. A C program is inspired by such thinking and ask you a favor to remove all comments from it. Your task is that given a C program you need to remove all comments.
   Input is a string with spaces (representing program)

   Sample Input :
```
#include<stdio.h>
/* Author : XYZ
*  Date : 21/1/2016
*/
int main()
{
        int a; // variable a
        return 0;
}
```

   Sample Output:
```
#include<stdio.h>
int main()
{
        int a;
        return 0;
}
```

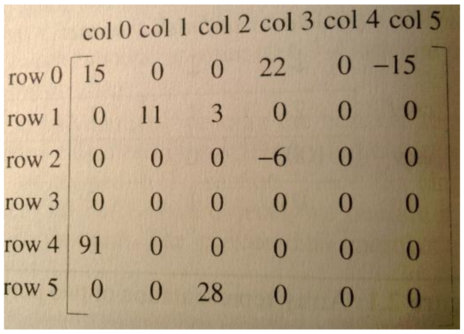
4. Write a C program to achieve the following. When the user inputs any number, show that in words.
   Note: User input range 0 – 9999

   Input:
   Enter a number <0 – 9999>.
   5435
   Output:
   Five Thousand Four Hundred and Thirty Five

5. Array based representation of sparse matrix and its transpose.
   As you know there is no precise definition of when a matrix is sparse and when it is not, you will make the following assumptions about a sparse matrix. The matrix you consider for this exercise is of size 6x6. If it has less than 10 non-zero elements then you call it a sparse matrix.

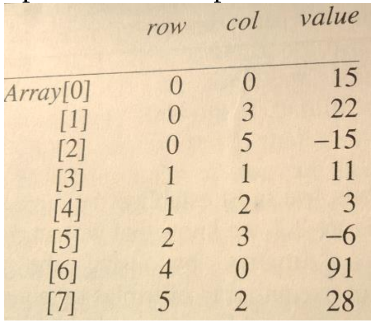
   Write a user interface driven C program to read a 6x6 matrix, create sparse representation if required, transpose the sparse representation.

   For example:



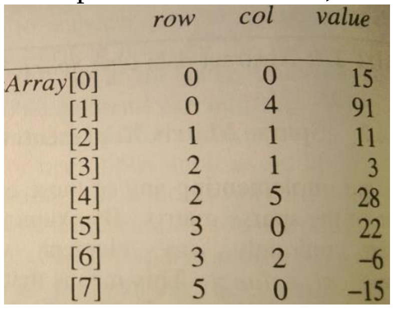
	col 0	col 1	col 2	col 3	col 4	col 5
row 0	15	0	0	22	0	−15
row 1	0	11	3	0	0	0
row 2	0	0	0	−6	0	0
row 3	0	0	0	0	0	0
row 4	91	0	0	0	0	0
row 5	0	0	28	0	0	0

Only 8 out of 36 possible elements are nonzero. Hence, sparse!
Sparse matrix representation: <row, column, value> tuple



	row	*col*	*value*
Array[0]	0	0	15
[1]	0	3	22
[2]	0	5	−15
[3]	1	1	11
[4]	1	2	3
[5]	2	3	−6
[6]	4	0	91
[7]	5	2	28

Transposed matrix: <row, column, value> tuple



	row	*col*	*value*
Array[0]	0	0	15
[1]	0	4	91
[2]	1	1	11
[3]	2	1	3
[4]	2	5	28
[5]	3	0	22
[6]	3	2	−6
[7]	5	0	−15

6. Point of sale (POS) simulation using structures.

   Items in a departmental store can be modeled as a structure. Hence, you can write a C program using the structure to simulate the process of a departmental store.
   For example, the structure that defines one item in a departmental store is as follows:

```
struct Item
{
        char itemName[20];
        int itemCode;
        float price;
        int QtyInStock;
        int IsHighDemand;
        int SoldToday;
}Item;
```

   Implement the following operations for the POS application.
   1. Add a new item
   2. Update the price of an item
   3. Update the stock
   4. Show the price list
   5. Sell an item
   6. Exit

7. Write a user interface driven C program to implement a "railfence cipher". A transposition cipher is one in which plaintext symbols are rearranged (i.e., transposed or permuted) to produce ciphertext (encrypted text). The method of transposition may be either mathematical or typographical in nature. One such type of cipher is "railfence cipher".
   In railfence cipher, the encryption key is a positive integer.

   Example: Suppose we want to encrypt the message "NOTHING IS AS IT SEEMS" with the encryption key (depth of rail fence) equal to 2.
   Arrange the plaintext characters in an array with 2 rows (the key determines the number of rows) as shown below. Your program must support any key length encryption.
   N T I G S S T E M
   O H N I A I S E S (ignore space characters)
   The ciphertext is produced by transcribing the first row followed by the second row.
   Ciphertext: NTIGSSTEMOHNIAISES. So, you need to read plain text as well as key.

8. You all know the fundamental theorem of arithmetic also called as unique factorization theorem. So you are given a number N, do the prime factorization so that $N = p_1^{x1} * p_2^{x2} * ... * p_k^{xk}$. You have to find the sum of the exponents i.e. (x1+x2+...+ xk).
   [Constraints : $2 \leq N \leq 10^8$. Time limit : 3s]
   Sample Input :
   12
   Sample Output :
   3

9. Suppose you are inhabitant of a planet where 1, 7, and 9 are lucky digits. A lucky number for you is a number that contains only your lucky digits in it. For ex: 1, 79, 911,9917 etc., are lucky, whereas 5, 172, 93, 170 are not. Given a integer N, count the number of lucky numbers in the range 1 to N. [Constraints: 1 ≤ N ≤ 10^12. Time limit : 3s]

   Sample Input:
   71

   Sample Output:
   7
   Explanation: The lucky numbers that are not more than 71 are 1, 7, 9, 11, 17, 19, 71

10. You are caught by math's geeks and they are not leaving you until you to solve a problem. You need to save your life by solving the problem. They give you integer N and K and ask you to find the smallest and the largest positive integers each containing exactly N digits and having exactly K distinct digits from 0 to 9. for ex: 474545 has only 3 distinct digits 4, 5 and 7. Obviously leading 0's are not allowed, for ex: 0145 is not allowed. You will get the input in only one line, containing N K (1 ≤ N ≤18 and 1 ≤ K ≤ 10). Print the smallest and largest integer.[Input is given such that the answer will always exist]

   Sample Input :
   3 3
   Sample Output :
   102 987

   Sample Input :
   3 2
   Sample Output :
   100 998

11. Assume that an array A with n elements was sorted in an ascending order, but two of its elements swapped their positions by a mistake while maintaining the array. Write a code to identify the swapped pair of elements and their positions in the asymptotically best possible time. [Assume that all given elements are distinct integers.]
   Input: An integer n followed by n distinct integers in an "almost" ascending order, i.e., after two of its elements have swapped their positions in the initially ascending array
   Output: The pair of elements which swapped their positions, followed by their positions in the array

12. You have to read a sorted array and an element x. You have to find out how many times x is present in the array in **O(log n)** time complexity.
   Input Format:
   n x
   [n number's separated by spaces]

   Output Format:
   Number of times element x is present.

Sample Input:
13 7
4 6 7 7 7 7 10 14 14 14 15 16 20

Sample Output:
4

13. You are given three sorted arrays of integers (in ascending order). Write a program to find a triplet (one element from each array) such that distance is minimum.

    If a[i], b[j], and c[k] are three elements, then distance is calculated as triangular distance.
    Distance = diff(a[i]-b[j]) + diff(b[j]-c[k]) + diff(c[k]-a[i])
    Solve it using O(n) complexity algorithm.

14. A Soundex code is the phonetic code of the word. If Soundex codes for two words are the same, that means they are pronounced same way. The enhanced Soundex code is created by following these steps:
    a. First, perform the following replacements:
        i. DG with G
        ii. GH with H ...WHEN... not at start of word
        iii. GN with N (not 'ng')
        iv. KN with N
        v. PH with F
        vi. MP with M ...WHEN... it is followed by S, Z, or T
        vii. PS with S ...WHEN... it starts a word
        viii. PF with F ...WHEN... it starts a word
        ix. MB with M
        x. TCH with CH
    b. Capitalize all letters in the word and drop all punctuation marks.
    c. Pad the word with rightmost blanks as needed during each procedure step.
    d. Retain the first letter of the word. Perform the following operations from the second letter onwards...
    e. Change all occurrence of the following letters to '0' (zero): 'A', E', 'I', 'O', 'U', 'H', 'W', 'Y'.
    f. Change letters from the following sets into the digit given:
        i. 1 = 'B', 'F', 'P', 'V'
        ii. 2 = 'C', 'G', 'J', 'K', 'Q', 'S', 'X', 'Z'
        iii. 3 = 'D','T'
        iv. 4 = 'L'
        v. 5 = 'M','N'
        vi. 6 = 'R'
    g. Remove one of the letters from all pairs of digits which occur beside each other from the string that resulted after step 6.
    h. Remove all zeros from the string that results from step 7.
    i. Pad the string that resulted from step 8. with trailing zeros and return only the first four positions, which will be of the form <uppercase letter> <digit> <digit> <digit>.

    For more details refer:
    http://creativyst.com/Doc/Articles/SoundEx1/SoundEx1.htm#SoundExConverter

    Write a C program to find the enhanced Soundex code of any given word. Then it prints those lines of a file that contain a word that sounds like a given word.

Input:
test.txt (input a file name with multiple lines)
ball (input word)

Output:
He addressed every class in a terrifying bawl.
Your father will bawl you out when he sees this mess.

Explanation:
Soundex code of both ball and bawl is B400

15. Random numbers are needed for many different purposes in engineering and computer science, i.e., to run simulations, to generate random passwords, etc. Around 1946 John von Neumann came up with the "**middle square method**" for generating random numbers.
    For example: if the starting number is 1234, the next seed will be middle digits of the square of 1234 which is 5227 (because square of (1234) = 01522756)
    Note: for even digit number: if the square of the initial or any seed in the process has odd no. of digits then zeros are padded to the left so that the next seed can be extracted. The same procedure has to be followed for odd number extraction.

    Input:
    Enter initial seed.
    222
    Enter no. of random numbers that are required.
    5

    Output:
    5 random numbers with the initial seed 222 are:
    928
    611
    733
    372
    383

16. Write a program to read a number n and print the numbers from 1 to n^2 pattern in a spiral order. You have to use a square matrix. And it should be dynamically allocated using **malloc**().

    for ex. n = 4 , output is :

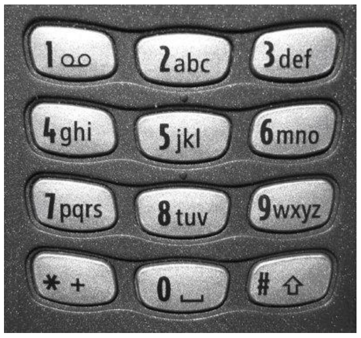
```
  1  2   3  4
 12 13  14  5
 11 16  15  6
 10  9   8  7
```

17. A submatrix of a matrix A is defined as a matrix obtained by deleting any collection of rows and/or columns from A. You have to read a square matrix **a** of size n x n. Find the maximum sum of a submatrix which is also a square matrix of size m X m where $1 \le m \le n$ & $1 \le n \le 100$ & $-100 < a[i][j] < 100$.
    Input Format:
    n (n row's having n numbers separated by spaces.)

Sample Input:
3
1 2 3
4 -5 6
7 -8 9
Output:
26

18. Feature phones have ten digits on the keypad. In Alphabet mode, if you have to type characters from 'a'-'z' you have to press keys in a certain sequence to get the desired character or number. The letters are mapped onto the digits as shown in figure below. For instance, to type the character 'n', the user has to press "66" and to type two characters in sequence from the same key, the user must pause before pressing the key a second time. The space character " " indicates a pause. For example, "6 6" indicates "mm" whereas "66" indicates "n". Numbers can be typed by pressing the key three/four times without a pause. So, to type number 6, the user needs to press "6666". Number 1 can be entered by pressing "11".

    Number 0 can be entered by pressing "00"



    Now your job is to read a line of string which have only lowercase characters 'a'–'z' and space and output the sequence of key presses(which can have pauses also).

    Sample Input-1:
    hello
    Sample Output-1:
    4433555 555666

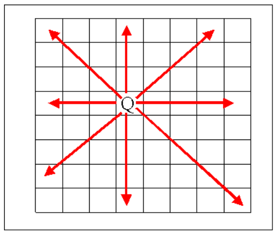
    Sample Input-2:
    dsa csf211 is cool
    Sample Output-2:
    37777202227777333222211 11044477770222666 666555

19. Given a string as the key pressed in the feature phone same as in the above figure, print the original text.

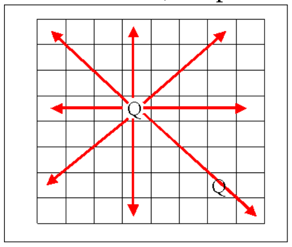
    Sample Input:
    37777202227777333222211 11044477770222666 666555

    Sample Output:
    dsa csf211 is cool

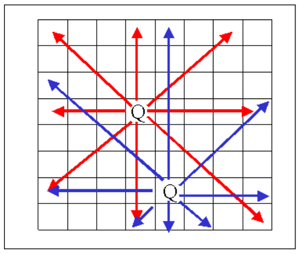
20. You are given a 8 X 8 chessboard on whicch a number of queens ranging from 2 to 8 are placed. In chess, a queen may move any number of spaces straight up, down, left, right, or along any of the 4 diagonals shown in Fig. 1. Any two queen can attack each other's position. Queens are said to be in attacking or not safe position when no two queens are in the same row, column, or diagonal. Fig. 2 depicts that two queens are attacking each other (same diagonal). Fig 3 represents two queens in non-attacking or safe position. You will be given a 2D integer array of 8 X 8 size containing 0s and 1s where 1 represent the presence of queen and 0 represent open spaces. You need to check if all queen's are placed in non-attacking position i.e. safe position. If safe, output YES otherwise NO.



(Fig.1)



(Fig. 2)



(Fig. 3)

Input Format : 8 line of input, each line contains eight numbers separated by spaces.
Output Format :YES or NO

Sample Input-1

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
```

Sample Output-1

```
NO
```

Explanation

There exist pairs of queen which are attacking each other one pair is {Queen placed at cell (3,0) and Queen placed at (6,0) } both are in same column. and other pair is {Queen placed at cell (1,6) and Queen placed at (4, 3) } both are in same diagonal

[PTO]

Sample Input-2

```
0 0 0 1 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0
0 0 0 0 0 0 1 0
0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0
```

Sample Output-2

```
YES
```

Explanation

All the queens are non-attacking hence YES