Zoho 2nd Round Coding questions

1.Given a text and a wildcard pattern, implement wildcard pattern matching algorithm that finds if wildcard pattern is matched with text. The matching should cover the entire text (not partial text).

The wildcard pattern can include the characters ‘?’ and ‘*’
‘?’ – matches any single character
‘*’ – Matches any sequence of characters (including the empty sequence)

2. Given an input string and a dictionary of words, find out if the input string can be segmented into a space-separated sequence of dictionary words. See following examples for more details.

```
Consider the following dictionary
{ i, like, sam, sung, samsung, mobile, ice,
  cream, icecream, man, go, mango}

Input:  ilike
Output: Yes
The string can be segmented as "i like".

Input:  ilikesamsung
Output: Yes
The string can be segmented as "i like samsung"
or "i like sam sung".<>
```

3.Print the following pattern

```
   1
  3 2
 6 5 4
10 9 8 7
10 9 8 7
 6 5 4
  3 2
```

```
   1
```

4.Given an array as input, The condition is if the number is repeated you must add the number and put the next index value to 0. If the number is 0 print it at the last.

```
Eg: arr[] = { 0, 2, 2, 2, 0, 6, 6, 0, 8}

Output: 4 2 12 8 0 0 0 0 0 .
```

5. Print the given input string in ‘X’ format.

Note: The string length will be of odd length.

6.Two sorted arrays will be given. Create an array consisting of the elements of two arrays with duplicate elements removed in sorted order.

Note: Use only one loop. No sorting.

7. Two strings of equal length will be given. Print all the adjacent pairs which are not equal.

Input: asdfghij and adsfgijh

Output: sd-ds, hij-ijh

8. Find the frequency of all numbers in an array.

Note: use dynamic memory allocation.

For example, if the input is {1, 2, 45, 67, 1, 88}, do not calculate the frequency of all elements from 1 to 88.

9.From the input sentence given, find the strings which are not palindrome and print it.

Input: he knows malayalam

Output: he knows

10. Given two Strings s1 and s2, remove all the characters from s1 which is present in s2.
Input: s1=”expErIence”, s2=”En”
output: s1=”exprIece”

11. Find the next greater element for each element in given array.
input: array[]={6, 3, 9, 10, 8, 2, 1, 15, 7};
output: {7, 5, 10, 15, 9, 3, 2, _, 8} If we are solving this question using sorting, we need to use any O(nlogn) sorting algorithm

12. Print all distinct permutations of a given string with duplicate characters.

https://www.geeksforgeeks.org/distinct-permutations-string-set-2

13. Given a number, find the next smallest palindrome

https://www.geeksforgeeks.org/given-a-number-find-next-smallest-palindrome-larger-than-this-number

14.Given an array with repeated numbers, Find the top three repeated numbers.
input: array[]={3, 4, 2, 3, 16, 3, 15, 16, 15, 15, 16, 2, 3}
output: 3, 16, 15

15. 1.Given two dimensional matrix of integer and print the rectangle can be formed using given indices and also find the sum of the elements in the rectangle
Input: mat[M][N] = {{1, 2, 3, 4, 6}, {5, 3, 8, 1, 2}, {4, 6, 7, 5, 5}, {2, 4, 8, 9, 4} };
index = (2, 0) and (3, 4)
Output:
Rectangle
4 6 7 5 5
2 4 8 9 4
sum 54

16. Find the result subtraction, multiplication, division of two integers using + operator.
Input: 6 and 4
output:
addition 6+4 = 10, subtraction 6+(-4) = 2, multiplication = 24, division = 1

Input : -8 and -4
Output:
addition -8+(-4) = -12, subtraction (-8)+(-(-4)) = -4, multiplication = 32, division = 2

17. Given a sentence of string, in that remove the palindrome words and print the remaining.
Input:
He did a good deed
Output:
He good

Input:
Hari speaks malayalam
Output:
Hari speaks

18. Given two dates, find total number of days between them.

Input: dt1 = {10, 2, 2014} dt2 = {10, 3, 2015}
Output: 393

dt1 represents "10-Feb-2014" and dt2 represents "10-Mar-2015" The difference is 365 + 28

Input: dt1 = {10, 2, 2000} dt2 = {10, 3, 2000}
Output: 29

Note that 2000 is a leap year

Input: dt1 = {10, 2, 2000} dt2 = {10, 2, 2000}
Output: 0

Both dates are same

Input: dt1 = {1, 2, 2000}; dt2 = {1, 2, 2004};
Output: 1461

Number of days is 365*4 + 1

19. Let 1 represent 'A', 2 represents 'B', etc. Given a digit sequence, count the number of possible decodings of the given digit sequence.
Examples:

Input: digits[] = "121"
Output: 3 // The possible decodings are "ABA", "AU", "LA"

Input: digits[] = “1234” Output: 3

// The possible decodings are “ABCD”, “LCD”, “AWD”

https://www.geeksforgeeks.org/convert-sentence-equivalent-mobile-numeric-keypad-sequence/

20. Print longest sequence between same character

Ex I/p abcccccbba

O/p 8 (from a to a)

I/p aaaaaaaa

O/p 6

21.sort the array odd numbers in ascending and even numbers in descending.

I/p 5 8 11 6 2 1 7

O/p 1 5 7 11 8 6 2

22. It’s about anagram.i/p was array of strings .and a word was given to find whether it has anagram in given array.

I/p catch, got, tiger, mat, eat, Pat, tap, tea

Word: ate

O/p eat, tea

23. array of numbers were given to find a number which has same sum of numbers in it’s either side.

I/p 1, 2, 3, 7, 6

O/p 7(has 1+ 2+3 in left 6 in right)

24. Separate 0s and 1s in single array traversal

25. Finding the largest palindromic sub word in a given word

26. Finding height of a binary tree

27. prime number – print n prime numbers

28. prime factor – sort the array based on the minimum factor they have.

29. adding a digit to all the digits of a number eg digit=4, number = 2875, o/p= 612119

30. Form the largest possible number using the array of numbers.

31. given a set of numbers, and a digit in each iteration, if the digit exists in any of the numbers, remove its occurrences and ask for the next digit till the list becomes empty.

32. Check if a number ‘a’ is present in another number ‘b.

```
33. Find the extra element and its index

Input : [ 10, 20, 30, 12, 5 ]

    [ 10, 5, 30, 20 ]

Output : 12 is the extra element in array 1 at index 4


Input : [ -1, 0, 3, 2 ]

    [ 3, 4, 0, -1, 2 ]

Output : 4 is the extra element in array 3 at index 5
```

34. Find the least prime number that can be added with first array element that makes them divisible by second array elements at respective index (check for prime numbers under 1000, if exist return -1 as answer) & (Consider 1 as prime number)

```
Input : [ 20, 7 ]

    [ 11, 5 ]

Output : [ 1, 3 ]


Explanation :

(20 + ?) % 11

( 7 + ?) % 5
```

35. Sort the array elements in descending order according to their frequency of occurrence

```
Input : [ 2 2 3 4 5 12 2 3 3 3 12 ]

Output : 3 3 3 3 2 2 2 12 12 4 5
```

Explanation : 3 occurred 4 times, 2 occurred 3 times, 12 occurred 2 times, 4 occurred 1 time, 5 occurred 1 time

```
Input : [ 0 -1 2 1 0 ]

Output : 0 0 -1 1 2
```

sort single occurrence elements in ascending order
36. Print true if second string is a substring of first string, else print false.

```
Note : * symbol can replace n number of characters

Input : Spoon  Sp*n  Output : TRUE

    Zoho     *o*o  Output : TRUE

    Man       n*     Output : FALSE

    Subline  line   Output : TRUE
```

37. Print second frequently occurring number in given series

Example :

```
Input: 1 1 2 3 1 2 4

Output: 2
```

Explanation: 1 occurs 3 times, 2 occurs 2 times, 3 occurs 1 time and 4 occurs 1 time. Hence second frequently occurring number in given series is 2

38. Print only numbers which is present in Fibonacci series (0 1 1 2 3 5 8 ……..)

Example:

```
Input: 2 10 4 8

Output: 2 8

Input: 1 10 6 8 13 21

Output: 1 8 13 21
```

39. Print pattern like this

```
Example:

Input: 1

Output: 0
```

```
Input: 2

Output:

0 0

0 1

1 0

1 1


Input: 3

Output:

0 0 0

0 0 1

0 1 0

0 1 1

1 0 0

1 0 1

1 1 0

1 1 1
```

40. NxN matrix will be provided. 0->block, 1->Not a block
Always starting point is (0,0), Ending point is (N-1,N-1).
You have to go from starting point to ending point. One valid solution is enough.
Example:

```
  Input:

  N=4

  1 1 0 0

  1 0 0 1

  1 1 1 1

  0 0 0 1

  Output:

  _ 1 0 0

  _ 0 0 1

  _ _ _ _
  0 0 0 _
```

41. Insert 0 after consecutive (K times) of 1 is found. Example:

```
Input:
Number of bits: 12
Bits: 1 0 1 1 0 1 1 0 1 1 1 1
Consecutive K: 2

Output:
1 0 1 1 0 0 1 1 0 0 1 1 0 1 1 0
```

42. Evaluate given expression which has factorials and exponential terms.

43. To implement snake and ladder game for given two-dimensional array having position of snakes and ladders

44. To calculate strength of the password string using some predefined rules given in the question

45. Given four points, We have to say whether it is square or rectangle or any other shape

46. Insert an element at a particular index in an array.

47. Given a large number convert it to the base 7.

48. Given an IP address validate it based on the given conditions.

49. Sort parts of an array separately using peak values.

50. Given an input array, find the number of occurrences of a particular number without looping (use hashing)
Diamond pattern printing based on some conditions
Given an array of characters print the characters that have ‘n’ number of occurrences. If a character appears consecutively it is counted as 1 occurrence
Eg: **a** b **a a** b c c d e d
Here a has only 2 occurrences

51. Efficiently merging two sorted arrays with O(1) extra space

https://www.geeksforgeeks.org/efficiently-merging-two-sorted-arrays-with-o1-extra-space/

52. Find the maximum of three numbers?

53. Print the total number of odd and even digits in the given number.

```
Ex.  Input  :  1234567

   Output  :  ODD 4
         EVEN 3
```

54 . Find the second maximum among the given numbers.

```
Ex.  INPUT  :

   Size of Array    :  8
   Enter the elements  :  2 5 1 6 2 6 7 10

   OUTPUT  :
   7

Ex.  INPUT  :

   Size of Array    :  4
   Enter the elements  :  4 1 2 2

   OUTPUT  :

   2

Ex.  INPUT  :

   Size of Array    :  1
   Enter the elements  :  1

   OUTPUT  :

   No second maximum
```

55. Print the following pattern

```
  Ex.  INPUT  :  5

    OUTPUT  :

            1
```

```
         1 1
        1 2 1
       1 3 3 1
      1 4 6 4 1

Ex.  INPUT  :  7

  OUTPUT  :

          1
         1 1
        1 2 1
       1 3 3 1
      1 4 6 4 1
     1 5 10 10 5 1
    1 6 15 20 15 6 1
```

56. Given a two dimensional array which consists of only 0’s and 1’s. Print the matrix without duplication.

```
Ex.  INPUT  :

  Enter Row Size    :  4
  Enter column size  :  3
  Enter the matrix  :
  1 0 1
  1 1 0
  1 1 1
  1 0 1

  OUTPUT  :

  Unique Matrix  :
  1 0 1
  1 1 0
  1 1 1
```

57. Given an array of positive numbers. Print the numbers which have longest continuous range.

```
Ex.  INPUT  :


  Enter array size  :  8
  Enter arryay elements  :  1 3 10 7 9 2 4 6


  OUTPUT  :


  1 2 3 4

Ex.  INPUT  :


  Enter array size  :  8
  Enter arryay elements  :  1 3 9 7 8 2 4 6


  OUTPUT  :


  1 2 3 4
  6 7 8 9
```

58. Given two arrays. Find its union.

```
Input  :


Enter size of first array  :  6
Enter the elements    :  1 2 3 4 5 3
Enter size of second array  :  4
Enter the elements    :  1 2 7 5


OUTPUT  :
```

```
1 2 3 4 5 7
```

59. Given an array of numbers. Print the numbers without duplication.

```
INPUT  :

Enter the array size  :  4
Enter the elements  :  1 1 2 4

OUTPUT  :

1 2 4
```

60. Given an array of numbers and a number k. Print the maximum possible k digit number which can be formed using given numbers.

```
INPUT  :

Enter the array size  :  4
Enter the elements  :  1 4 973 97
Enter number of digits  :  3

OUTPUT  :

974

INPUT  :

Enter the array size  :  6
Enter the elements  :  1 4 89 73 9 7
Enter number of digits  :  5

OUTPUT  :
```

```
98973
```

61. Given an array of numbers and a window of size k. Print the maximum of numbers inside the window for each step as the window moves from the beginning of the array.

```
INPUT  :


Enter the array size  :  8

Enter the elements  :  1,3,5,2,1,8,6,9

Enter the window size  :  3


OUTPUT  :


5 5 5 8 8 9
```

62. Find the minimum number of times required to represent a number as sum of squares.

```
12 = 1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 +

    1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2

12 = 2^2 + 2^2 + 2^2

12 = 3^2 + 1^2 + 1^2


Input: 12

Output: min: 3
```

63. Search a string in a given 2D matrix. And print its possible path.
allowed movements are right left up and down.

64. In a given pascal triangle find the possible triangles.

65. in a matrix find the number of rectangles filled with 1s.

```
Input:   0 1 1 0

         1 1 1 0

         0 0 1 1

         0 0 1 1


Output: 2.
```

66. There are n items each with a value and weight. A sack is filled with the weights. In other words there is an array with of length n having the values of the items arr[0…n-1] and another array with weight arr[0…n-1].
if a sack is to be filled with weight W find the minimum possible value subset.

67. Write a program to determine whether a given number can be expressed as sum of two prime numbers or not.

For example 34 can be expressed as sum of two prime numbers but 23 cannot be.

68. Take a 2 or 3 digit input number, reverse it and add it to the original number until the obtained number is a palindrome or 5 iterations are completed.

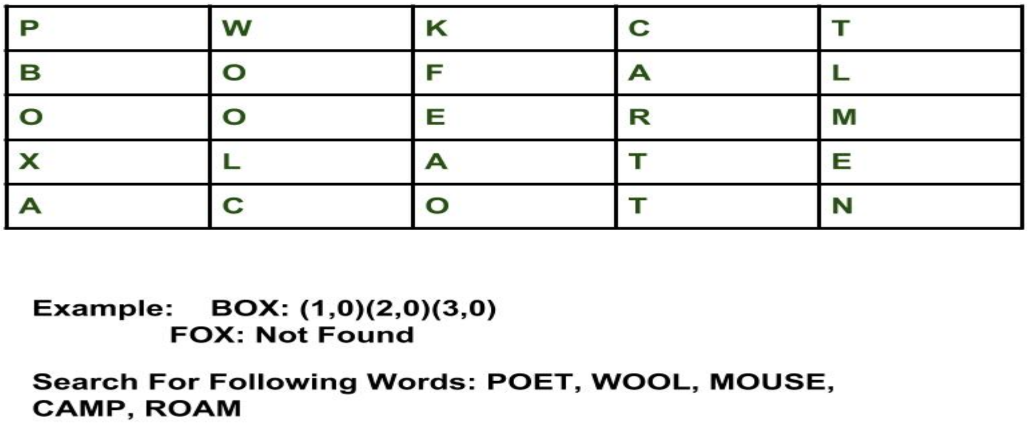
Input : n = 32
Output : 55
23 + 32 = 55 which is a palindrome.

Input : 39
Output : 363

69. Given a string, reverse only vowels in it; leaving rest of the string as it is.

Input : abcdef
Output : ebcdaf

70. Write a program to check if the given words are present in matrix given below. The words can be left to right, top to bottom and the diagonals (in top to bottom direction)



P	W	K	C	T
B	O	F	A	L
O	O	E	R	M
X	L	A	T	E
A	C	O	T	N

Example: BOX: (1,0)(2,0)(3,0)
FOX: Not Found

Search For Following Words: POET, WOOL, MOUSE, CAMP, ROAM

71. Write a program to form lines using given set of words. The line formation should follow below rules.

i) Total characters in a single line excluding the space between the words and the favorite character should not exceed the given number.

ii) Favorite character is case insensitive.

iii) Words should not be broken up. Complete words alone should be used in a single line. A word should be used in one line only.

```
Input : Max char per line = 10

        Favorite character = 'o'

        Words : Zoho, Eating, Watching, Pogo

                Loving, Mango

Output : Watching Zoho

         Eating Mango

         Loving Pogo.
```

72. Adding 2 numbers

GIven 2 huge numbers as seperate digits, store them in array and process them and calculate the sum of 2 numbers and store the result in an array and print the sum.

Input:
Number of digits:12
9 2 8 1 3 5 6 7 3 1 1 6
Number of digits:9
7 8 4 6 2 1 9 9 7
Output :
9 2 8 9 2 0 2 9 5 1 1 3

73. Given sorted array check if two numbers sum in it is a given

value
Input
Array = {1 3 4 8 10 }

N = 7
output
true

74. Compiuting value of sin (x)
Input x = 30 n = 10
output = 0.5

Hint : The equation sin(x) = x – x^3 / 3! + x^5 / 5! – ….

75. Write function to find multipication of 2 numbers using + operator You must use minimu possible iterations.

Input: 3 , 4
Output 12

76. Given array find maximum sum of contiguous sub array

{-2 -3 4 -1 -2 1 5 -3}

output 7 elements [4 -1 -2 1 5]

77. Given unsorted array find all combination of the element for a given sum. Order should be maintained.

Input :
8 3 4 7 9

N=7
Output
{3 4 } {7}

78. Given an odd length word which should be printed from the middle of the word. The output should be in the following pattern.
Example:

```
Input: PROGRAM

Output:

              G

            GR

          GRA

       GRAM

     GRAMP

   GRAMPR

GRAMPRO
```

79. It is a program to implement Least Recently Used (LRU) concept. Given a key, if it is already existed then it should be marked as recently used otherwise a value should be stored which is given as input and marked as recently used. The capacity is to store only 10 key, value pairs. If the table is full and given a new key; the key, value pair which is not recently used should be deleted which gives feasibility to store the new key, value pair.

80. Given a few pairs of names in the order child, father. The input is a person name and level number. The output should be the number of children in that particular level for the person given.
Example:
Input:
[
{Ram, Syam},
{Akil, Syam},
{Nikil, Ram},
{Subhash, Ram},
{Karthik, Akil}
];

Syam 2

Output: 3 (Syam has Ram and Akil in level 1 and in level 2 he have Nikil, Subhash, Karthik. So the answer is 3).

81. Given an array of positive integers. The output should be the number of occurrences of each number.
Example:
Input: {2, 3, 2, 6, 1, 6, 2}
Output:
1–1
2–3
3–1
6–2

82. sort numbers based on digits starting from most significant numbers. Eg: input-100 1 11 21 2 3. Output-1 100 11 2 21 3

83. Given an array, find the minimum of all the greater numbers for each element in the array.

```
Sample:

Array : {2, 3, 7, 1, 8, 5, 11}


Output:

{2>3, 3>5, 7>8, 1>2, 8>11, 5>7, 11>}
```

84. Find the largest sum contiguous subarray which should not have negative numbers. We have to print the sum and the corresponding array elements which brought up the sum.

```
Sample:

Array : {2, 7, 5, 1, 3, 2, 9, 7}


Output:

     Sum : 14

     Elements : 3, 2, 9
```

85. Given a string, we have to reverse the string without changing the position of punctuations and spaces.

```
Sample:   house no : 123@ cbe
Output:    ebc32 1o :  nes@ uoh
```

86. Given a 2D grid of characters, you have to search for all the words in a dictionary by moving only along two directions, either right or down. Print the word if it occurs.

```
Sample :
  a   z  o   l
  n   x  h   o
  v   y   i   v
  o   r   s  e
 Dictionary = {van, zoho, love, are, is}

 Output:
    zoho
    love
    Is
```

87. Given a string, change the order of words in the string (last string should come first). Should use RECURSION

```
Sample:   one two three
Output :  three two one
```

88. Given a number, convert it into corresponding alphabet.

```
Input Output
      1     A
     26     Z
     27     AA
     676     ZZZ
```

89. Given a Roman numeral, find its corresponding decimal value.
https://www.geeksforgeeks.org/converting-roman-numerals-decimal-lying-1-3999/

90. Write a program to print all permutations of a given string. Note here you need to take all combinations as well, say for the input ABC the output should be as follows:

```
Input: ABC
Output:
A
B C
AB AC BA BC CA CB
ABC ACB BCA BAC CBA CAB
```

91. Write a program to rotate an n*n matrix 90,180,270,360 degree.
https://www.geeksforgeeks.org/inplace-rotate-square-matrix-by-90-degrees/ is the solution for rotating a matrix 90 degree. For rotating the matrix 180,270,360 degree, u need to call the same method 2,3,4 times based on the input.

92. https://www.geeksforgeeks.org/reverse-words-in-a-given-string/

93. Write a program to convert a number into a mono-digit number.

Conditions:
a) You are allowed to add and subtract the consecutive digits (starting from left).
b) You are allowed to do only one operation on a digit.
c) You cannot perform any operation on a resultant digit of the previous operation.
d) Your code should also find if a given number cannot be converted to a mono digit number.

```
      Input      Output
    72581          7(2+5)81
                   77(8-1)
                   777
    3962           cannot create a mono digit number
```

94. You're given an array. Print the elements of the array which are greater than its previous elements in the array.
Input : 2, -3, -4, 5, 9, 7, 8 **Output:** 2 5 9 You should solve this question in O(n) time.

95.

You're given an even number n. If n=4, you have to print the following pattern :

```
4444

4334

4334

4444
```

If n=6, then the pattern should be like this :

```
666666

655556

654456

654456

655556

666666
```

96. You're given a number n. If write all the numbers from 1 to n in a paper, we have to find the number of characters written on the paper.For example if n=13, the output should be 18 if n = 101, the output should be 195

97. A number is called as binary-decimal if all the digits in the number should be either '1' or '0'. Any number can be written as a sum of binary-decimals. Our task is to find the minimum number of binary-decimals to represent a number.**Input :** 32**Output :** 10 11 11
Input : 120
Output : 10 110

98. You're given a string as an input. You have to reverse the string by keeping the punctuation and spaces. You have to modify the source string itself with creating an another string.
Input :A man, in the boat says : I see 1-2-3 in the sky

Output :
y kse, ht ni3 21ee slsy : a sta o-b-e ht nin amA

99. Given two numbers a and b both < 200 we have to find the square numbers which lie between a and b(inclusive)

```
eg) i/p a = 20;b = 100;
      o/p 25,36,49,64,81,100
```

Alternately sort an unsorted array..

```
eg) i/p {5,2,8,7,4,3,9}
      o/p {9,2,8,3,7,4,5}
```

100. Given an array and a threshold value find the o/p

```
eg) i/p {5,8,10,13,6,2};threshold = 3;
      o/p  count = 17
      explanation:
Number   parts                counts
5                    {3,2}                2
8                      {3,3,2}             3
10                    {3,3,3,1}          4
13                    {3,3,3,3,1}       5
6                      {3,3}                 2
2                      {2}                    1
```

101. Given two binary numbers add the two numbers in binary form without converting them to decimal value.

```
eg) a = 1010 b = 11001
      o/p  100011
  b.The two numbers were given in base n
         eg) a = 123  b = 13  n = 4
               o/p  202
```

102. Write a program to print the below pattern

```
for n = 6
         1        7        12       16       19       21
         2        8        13       17       20
3        9        14       18
4        10       15
5        11
6
```

103. Given bigger NxN matrix and a smaller MxM matrix print TRUE if the smaller matrix can be found in the bigger matrix else print FALSE

104. Given two matrices a and b both of size NxN find if matrix a can be transformed to matrix b by rotating it 90deg , 180deg , 270deg if so print TRUE else print FALSE

105. In addition to the above question you have to check if matrix a can be transformed by mirroring vertically or horizontally to matrix b.

106. Given an array of integers, rearrange the array in such a way that the first element is first maximum and second element is first minimum.

```
    Eg.) Input  : {1, 2, 3, 4, 5, 6, 7}
         Output : {7, 1, 6, 2, 5, 3, 4}
```

107. Remove unbalanced parentheses in a given expression.

```
    Eg.) Input  : ((abc)((de))
         Output : ((abc)(de))

         Input  : (((ab)
         Output : (ab)
```

108. Form a number system with only 3 and 4. Find the nth number of the number system. Eg.) The numbers are: 3, 4, 33, 34, 43, 44, 333, 334, 343, 344, 433, 434, 443, 444, 3333, 3334, 3343, 3344, 3433, 3434, 3443, 3444 ….

109. Check whether a given mathematical expression is valid.

```
    Eg.) Input  : (a+b)(a*b)
         Output : Valid

         Input  : (ab)(ab+)
         Output : Invalid

         Input  : ((a+b)
         Output : Invalid
```

110. Using Recursion reverse the string such as

```
Eg 1: Input: one two three
      Output: three two one
Eg 2: Input: I love india
      Output: india love I
```

111. Given two sorted arrays, merge them such that the elements are not repeated

```
Eg 1: Input:
        Array 1: 2,4,5,6,7,9,10,13
        Array 2: 2,3,4,5,6,7,8,9,11,15
       Output:
       Merged array: 2,3,4,5,6,7,8,9,10,11,13,15
```

112. Find if a String2 is substring of String1. If it is, return the index of the first occurrence. else return -1.

```
Eg 1:Input:
        String 1: test123string
         String 2: 123
        Output: 4
Eg 2: Input:
        String 1: testing12
        String 2: 1234
        Output: -1
```

113. Write a program to print the following output for the given input. You can assume the string is of odd length

```
Eg 1: Input: 12345
       Output:
1       5
  2   4
    3
  2  4
1      5
Eg 2: Input: geeksforgeeks
         Output:
g                         s
  e                     k
    e                 e
      k             e
        s         g
          f      r
             o
          f     r
        s         g
      k             e
    e                 e
  e                      k
g                          s
```

114. Write a program to sort the elements in odd positions in descending order and elements in ascending order

```
Eg 1: Input: 13,2 4,15,12,10,5
        Output: 13,2,12,10,5,15,4
Eg 2: Input: 1,2,3,4,5,6,7,8,9
        Output: 9,2,7,4,5,6,3,8,1
```

115. Write a program to give the following output for the given input

```
Eg 1: Input: a1b10
       Output: abbbbbbbbbb
Eg: 2: Input: b3c6d15
          Output: bbbccccccddddddddddddddd
The number varies from 1 to 99.
```

ADVANCED CODING

1.

One application will be given. Time: 2h30m

Write an application for booking railway ticket reservation system. The application should have four functionalities.

1. Book
2. Cancel
3. Print booked tickets (details with summary)
4. Print available tickets (details with summary)

Conditions for booking:

There are a total of 63 berths for 63 confirmed tickets, 9 berths for 18 RAC tickets and 10 tickets in waiting-list. If the waiting-list ticket count goes above 10, print as ‘No tickets available’. The following passenger details should be obtained from the user.

1. Name
2. Age
3. Gender
4. Berth Preference

The tickets should not be allocated for children below age 5.But, their details should be stored. Lower berth should be allocated for persons whose age is above 60 and ladies with children if available. Side-lower berths should be allocated for RAC passengers.

Conditions for cancelling:

Whenever a ticket is cancelled, a ticket from RAC should be confirmed and a waiting-list ticket should move to RAC.

Conditions for printing booked tickets:

Print all the tickets that are filled along with the passenger details and at the end, print the total number of tickets that are filled.

Conditions for printing available tickets:

Print all the tickets that are unoccupied and at the end, print the total number of tickets that are unoccupied.

2.

This round is mainly based on data structure and oops concepts. No inbuilt collections are allowed. You need to implement on your own.

Needs to discuss your approach before start solving the problem.

Design a system with following functionalities,

1. SET a variable
2. GET a variable
3. UNSET a variable
4. COUNT NUMBERS OF VARIABLE with given value
5. BEGIN — Begins a new transaction
6. ROLLBACK — Roll back all the commands in the open transaction
7. COMMIT — Commit the transaction
8. EXAMPLE 1:

9. SET a 20
   GET a 20
   SET b 30
   GET b 30
   SET a 10
   GET a 10
   UPDATE c 40 No variable named “c”
   SET c 30
   COUNT 30 2
   COUNT 40 null
   UNSET a
   GET a null

10. EXAMPLE 2:

11. GET a null
    SET a 30
    GET a 30

12. EXAMPLE 3:

13. SET a 30
    BEGIN
    GET a 30
    SET a 40
    GET a 40
    SET b 40
    GET b 40
    ROLLBACK

```
    GET                          b                           null
    GET a 30
14. EXAMPLE 4:
15. BEGIN
    SET                          a                             40
    SET                          b                             40
    SET                          c                             50
    COUNT                          40                           2
    BEGIN
    COUNT                         40                         null
    COMMIT
    COUNT                          40                           2
    BEGIN
    SET                          c                             10
    GET                          c                             10
    ROLLBACK
    GET c 50
```

3. Given an employee date base.

Name, Age, Designation, Department Of ten people.

and Five tasks were given such as

1. Print all employee details.

2. Searching employee details

3. Employees under the given manger name of the department

4. reporting to tree of the given employee name

4.Size of the array is given were w is wall, g ground, o ball, numbers are bricks.

I/p size 7

Number of bricks 6

Position (2, 2)(2, 3)(2, 4)(3, 2)(3, 3)(3, 4)

Ball:5(life)

```
w w w w w w w
w           w
w   1 1 1   w
w   1 1 1   w
w           w
w           w
w g g  o g g w
```

There are three commands St, lt, rt straight, left, right respectively.

If it is st the ball moves straight .if there any brick(1) along the way it hit it .then brick disappear.ball back to original position.if there is no brick .it come to initial position.

I/p st

O/p

```
w w w w w w w
w           w
w   1 1 1   w
w   1   1   w
w           w
w           w
w g g  o g g w
```

Ball count:5

I/p lt

O/p

```
w w w w w w w
w           w
w   1 1 1   w
w        1  w
w           w
w           w
w g o g  g g w
```

Ball count:4

(Lt : ball moves diagonally left there is no bricK on the way so it hit the wall reflect back horizontally there is a brick(3, 2) after hitting it it moves downwards ball position changed.hence ball count get reduced. On moving downwards is there any brick that also disappear.)

Same for rt but moves diagonally right.

This is first module.In second module each brick has value for each hit value get reduced.it disappear only when it become zero.

There are about 7 modules.

5. implement a game in C, that’s well known as ‘Tetris blocks’. It involves full of matrix operations.

6. This time all the questions were based on matrix. I got 6 programs, again you get the next program after you complete the current program. All the programs were just an addition to the previous one. Here the concepts to be strong about were logic for diagonals (all the four, top-left, top-right, bottom-left, bottom-right), finding the path in a matrix from source to destination.

7. Lift system
   There were 8 modules
   1. Display the position of Lift

```
Lift   : L1 L2 L3 L4 L5
```

```
Floor: 0   0   0   0   0
```

2. Assign Lift to the users

```
Input : 2  5

Output : L1 is assigned
```

```
Lift   : L1 L2 L3 L4 L5

Floor: 5   0   0   0   0
```

3. Assign nearest lift by comparing their current positions
Assume,

```
Lift   : L1 L2 L3 L4 L5

Floor: 5   2   7   9   0

Input : 4 10

Output :

L1 is assigned

Lift   : L1 L2 L3 L4 L5

Floor: 10   2   7   9   0
```

Explanation : L1 is near to 4 floor

4. If two lifts are nearest to the user's source floor, the assign the lift with same direction of user's requirement.
Example: if user request to move from 4 to 2 ,and if L3 is in 5th floor & L5 is in 3rd floor, then we should assign L3 because user requested for downward motion so L3 ill move down from 5th floor

5. Restrict L1 & L2 for 0-5th floor , L3 & L4 for 6-10th floor , L5 for 0-10th
Initially all lifts are at 0th floor.

6. Assign lift with least number of stops
Example:
If L3 is in 9th floor
And L5 is at 8nd floor
If user wants to move from 8 to 0
We should assign L3 because L3 ill stop at 8,7,6 and then 0 NumberOfStops = 3, but L5 ill stop at 8,7,6,5,4,3,2,1,0 and NumberOfStops = 8 so we should assign L3

7. Assign capacity (Number of people capable to travel) to all lift and assign according to the capacity

8. If any lift is under maintenance then their current position should be marked as “-1” and that lift should not be assigned at any cost.

8. The question is to implement the mail server.

9. In these rounds we were asked to implement the logic for 2 games – Minesweeper – Breakout a.k.a. Arkanoid a.k.a. Brick-Breaker (you'll find it online)

   The game was split into various stages and we were asked to implement the logic stage by stage.

10. Given a MxN matrix filled with '-' and you need to drop the balloons in the desired columns starting from the bottom. You need to print the matrix when a new balloon is dropped. You need to continue getting inputs until the box is full or until the user chooses to stop.

```
TEST CASE  :


Enter the matrix size(m*n)  :  3 3

Enter the column number    :  2

Enter the color of the balloon  :  R

Contents of the matrix    :

- - -
- - -
- R -
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  B
Contents of the matrix    :
- - -
- B -
- R -
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  1
Enter the color of the balloon  :  R
Contents of the matrix    :
- - -
- B -
R R -
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  R
Contents of the matrix    :
- R -
- B -
R R -
Do you wish to continue(Y/N)  :  N
Program Stopped
```

Extended version of the previous problem. Now you need to quit when a row become filled completely.

```
TEST CASE  :


Enter the matrix size(m*n)  :  3 3
Enter the column number    :  2
Enter the color of the balloon  :  R
Contents of the matrix    :
- - -
- - -
- R -

Do you wish to continue(Y/N)  :  Y

Enter the column number    :  2

Enter the color of the balloon  :  B

Contents of the matrix    :

- - -

- B -

- R -

Do you wish to continue(Y/N)  :  Y

Enter the column number    :  2

Enter the color of the balloon  :  R

Contents of the matrix    :

- R -

- B -

- R -

Column is filled completely. Program is terminated.
```

Extended version of the previous problem. Now you need to drop balloon in the first free cell from left if the specified column is filled in every row.

```
TEST CASE  :


Enter the matrix size(m*n)  :  3 3

Enter the column number    :  2

Enter the color of the balloon  :  R

Contents of the matrix    :

- - -

- - -

- R -

Do you wish to continue(Y/N)  :  Y

Enter the column number    :  2

Enter the color of the balloon  :  B

Contents of the matrix    :

- - -

- - -

B R -

Do you wish to continue(Y/N)  :  Y
```

```
  Enter the column number    :  2
  Enter the color of the balloon  :  R
  Contents of the matrix    :
  - - -
  - - -
  B R R
  Do you wish to continue(Y/N)  :  Y
  Enter the column number    :  2
  Enter the color of the balloon  :  R
  Contents of the matrix    :
  - - -
  - R -
  B R R
  Do you wish to continue(Y/N)  :  Y
  Enter the column number    :  2
  Enter the color of the balloon  :  B
  Contents of the matrix    :
  - - -
  B R -
  B R R
  Do you wish to continue(Y/N)  :  N
  Program terminated.
```

Extended version of the previous problem. If any column has three continuous balloons of same colors then we need to burst them.

```
  TEST CASE  :


  Enter the matrix size(m*n)  :  3 3
  Enter the column number    :  2
  Enter the color of the balloon  :  R
  Contents of the matrix    :
  - - -
  - - -
  - R -
  Do you wish to continue(Y/N)  :  Y
  Enter the column number    :  2
```

```
Enter the color of the balloon  :  R
Contents of the matrix    :
- - -
- - -
R R -
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  R
Contents of the matrix    :
- - -
- - -
R R R
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  R
Contents of the matrix    :
- - -
- R -
R R R
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  B
Contents of the matrix    :
- - -
R R -
R R R
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  R
Contents of the matrix    :
- - -
R R R
R R R
Do you wish to continue(Y/N)  :  Y
Enter the column number    :  2
Enter the color of the balloon  :  R

Contents of the matrix    :

- - -

R - R

R - R

Do you wish to continue(Y/N)  :  N

Program Terminated.
```

Extended version of the previous problem. Now you need to burst the three continuous colors in the same row.

11. You are given coordinates as input
(0 0) (0 1) (0 2) (0 3) (1 5) (1 4) (3 5)

1. Check if the given points lie in the same line

2. GIven a point find the points on the largest line [in terms of the points it contain] passing through that point

Input : 0 1
Output : (0 0) (0 1) (0 2) (0 3)

3. Given 2 points find the points in between them.
Input (0 0) (0 3)
Output : (0 1) (0 2)

4. Find the number of points in the line with the largest number of points in it.

5. Given a point print all the lines passing through it [ie for each line print the points in it]

12 . The task is to develop an Invoice Management. The main focus in this round is in designing part. We should mainly focus on the data base part. How we are going to design the tables to store data and how we are going manage the data plays an important role. It is better to refresh DBMS concepts before attending the ZOHO placements. Techtud videos on DBMS in youtube helped me a lot to understand basic concepts of DBMS. Particularly, in this round think in the aspect of ER models (Tables, attributes, primary and foreign key etc). After designing part they gave us two queries. Here, the coding part should be done in a very optimal way. Based on the query, from the data we stored, the result should be printed. I did in C language but better to implement in any OOPS language.

(Another version of same question)

We were asked to develop a mini-project 'Invoice Management' with the following modules :

1. Add a customer
2. Add an invoice
3. Add items to an invoice
4. List all customers
5. List all invoices
6. List all invoices of a customer
7. Display the full details of an invoice

13. The application was TOLL PAYMENT PROCESSING

They insisted us to do it in a object oriented language. First they asked the design(what are all the classes and objects & what data structure do you use).

Application description:

-
- There are 'n' number of points in a highway out of which some points collect toll.
- Each toll has its own charging scheme according to the vehicles and whether or not they are a VIP user.
- If they are VIP user, 20% discount apply.

- If the vehicle passes 3 toll gates, it has to pay in all the 3 toll gates according to the scheme of respective tolls.

There were 4 modules.

1. Given the details of vehicle type, start and destination……display the total toll paid during the journey and print the amount after applying the discount.

2. Display the details of all the tolls…..like what are all the vehicles(vehicle number) passed that respective toll and the amount each vehicle paid……and the total amount charged in that toll.

3. Display the details of all the vehicles …….like what are all the journeys did it take….the start and destination of the same……tolls it passed during that journey….amount paid in that journey…..and the total amount paid by that vehicle.

4. Assume the highway as a circular path……we have to find the short route and identify the tolls between that route and calculate the amount.

14. We were asked to design an application program for n*n tic-tac-toe game. Here, you are expected to code with proper standards and in a most optimized way. And, in this round you need to find all the edge cases and corner cases yourself. The interviewers won't help you if you miss anything. So, make sure you covered all the cases before showing output to the interviewers.

https://www.geeksforgeeks.org/implementation-of-tic-tac-toe-game/

15 . An adventurer, A monster, A trigger, A treasure, Pits these are the components.
The size and location shall be given in run time. Adventurer must reach treasure fast than monster else he dies (Hint: use absolute of distance)

16.questions based on maze.

Q1: Given a matrix dimension find the shortest path between two points.

Q2: shortest path between two points and a monster is present. So we have to avoid the monster and take a shortest path or die.

Q3: print the maze step by step.

Q4: same as Q2 but trigger is present so we can take the trigger to shoot the monster anywhere in the maze.

Q5: this is the tricky part. There are many holes in between. we should avoid them and take the shortest path.

17 . text editor
Only 40 characters per line and words should be wrapped if they brake
Also perform insert delete operations

18. To form a structure which has few elements:

```
struct product {

   char productname[20];

   int product_price;
   int product_id;

}
```

Get the product name, price and id and display the product name and price in descending of the price.

For the same above structure, now add another structure which is the category. That category will have products in it.

```
Struct category

{

   char category_name[20];

   int cat_id;

}
```

According the category get the product name, product price and id, then display all the products category wise in descending order.

For the same structure which as category and product, get the category id from the user in the product structure and save to the category list. Then display them all in category wise.

A sheet full of data will be given with inventory stock list, which as different categories and different products as input with category capacity and product availability in the structure. Now we need to add a new category or new product with capacity and availability. Need to check whether the product availability is exceeding the category capacity, if yes the output rack is full or else tell how much free space is available and add the product to list.

Constraints in the above in question will be given, need to solve all the constraints

19. Design a Call taxi booking application
-There are n number of taxi’s. For simplicity, assume 4. But it should work for any number of taxi’s.
-The are 6 points(A,B,C,D,E,F)
-All the points are in a straight line, and each point is 15kms away from the adjacent points.
-It takes 60 mins to travel from one point to another
-Each taxi charges Rs.100 minimum for the first 5 kilometers and Rs.10 for the subsequent kilometers.
-For simplicity, time can be entered as absolute time. Eg: 9hrs, 15hrs etc.
-All taxi’s are initially stationed at A.
-When a customer books a Taxi, a free taxi at that point is allocated
-If no free taxi is available at that point, a free taxi at the nearest point is allocated.
-If two taxi’s are free at the same point, one with lower earning is allocated
-Note that the taxi only charges the customer from the pickup point to the drop point. Not the distance it travels from an adjacent point to pickup the customer.
-If no taxi is free at that time, booking is rejected

Design modules for

```
1)    Call taxi booking

Input 1:

Customer ID: 1

Pickup Point: A

Drop Point: B

Pickup Time: 9


Output 1:

Taxi can be allotted.

Taxi-1 is allotted


Input 2:

Customer ID: 2

Pickup Point: B

Drop Point: D

Pickup Time: 9


Output 1:

Taxi can be allotted.

Taxi-2 is allotted
```

(Note: Since Taxi-1 would have completed its journey when second booking is done, so Taxi-2 from nearest point A which is free is allocated)

```
Input 3:
Customer ID: 3
Pickup Point: B
Drop Point: C
Pickup Time: 12

Output 1:
Taxi can be allotted.
Taxi-1 is allotted
```

2) Display the Taxi details

```
Taxi No:    Total Earnings:
BookingID    CustomerID    From    To    PickupTime    DropTime    Amount

Output:
Taxi-1    Total Earnings: Rs. 400

1     1     A    B    9    10    200
3    3    B    C    12    13    200

Taxi-2 Total Earnings: Rs. 350
2    2    B    D    9    11    350
```

These were just sample inputs. It should work for any input that they give. Those who finished both the modules within 3 hours and if it worked for all the inputs they give, those candidates were given extra modules to work with.

20. A matrix game was given with 5 rules. We were asked to implement each of the rules separately.

```
  R3 | -   -   - |
  R2 | -   -   - |
  R1 | -   -   - |
       C1  C2  C3
```

Each of the 9 cells can either be empty or filled with an atom. R3, R2, R1 are the rays that originate from the left. C1, C2, C3 are the rays that originate from the bottom of the box.

Input : Position of the atoms and the rays that gets originated from the outside of the box.

```
  Eg.) 3
       3 1
       2 2
       1 3
       3
       R3 C1 C3


Output  : Print the box.
```

Rule 1:
A ray that has an atom in its path should print 'H' (Hit) If it does not have any atoms in its path, the ray should pass to the other side.

```
       C1      C3
  R3 | -   -   - | R3
  H  | -   X   - |
  R1 | -   -   - | R1
       C1  H   C3
```

Rule 2 & 3:
A ray that has an atom in its diagonal adjacent position should refract.

```
  H  | -   -   - |
  H  | X   -   - |
  R  | -   X   - |
       R   H   R

Input rays: R1, R2, C3


  H  | -   X   - |
  R2 | -   -   - | C3
     | -   -   - |
       R2      C3
```

Rule 4:
A ray that has atoms in both of the diagonal adjacent positions should reflect back.

```


Input ray: C2
   | -   -   - |
   | X   -   X |
   | -   -   - |
         R


Input ray: R2
     | -   X   - |
  R  | -   -   - |
     | -   X   - |
```

Rule 5:
The deflection of rays should happen in the order of the input rays.

```
Input Rays: R3, R2, C1, C3
   H | -   X   - |
  R2 | -   -   - | C3
     | -   -   - |
       R2      C3
```

The final task was to implement these rules for dynamic matrix size.

```
Input : no of rows, no of columns

  Eg.) 4 4 (row & column)

       2 (No of atoms)

       4 4 (Position of atom)

       2 2 (Position of atom)

       2 (No of rays)

       R4 C2 (Ray number)


  H  | -   -   -   X |

     | -   -   -   - |

     | -   X   -   - |

     | -   -   -   - |

           H
```

The final task was very confusing and it had to handle all the cases. There are chances for a ray to end at the starting position if the number of rows and columns are more than 5.