Logic Development Workshop

Problem Statements

Team ASCI 2012-14

Some Tips:-

1. Go for modular coding
2. Increase readability of code (**Good naming conventions**).
3. Use minimum number of VARIABLE's (**reuse them**).
4. Go for dynamic programming. (**Generalization**)

Loops:

1.Sum of n first numbers.(normal, formula)

2. Write a C program to print the multiplication table of the number entered by the user. The table should get displayed in the following form:-

29*1=29
29*2=58
.
.

3.Factorial of given no.

4. Sum of digits in a number.

5. Sum of Digits until Single Digit is obtained

6.Sum of digits at even, odd place in a number.

7. Program to find x raise to y.

8.Algorithm to check if no. is prime or not.

9.Write a algorithm to print all prime no's from 1 to n.

10. If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The sum of these multiples is 23. Find the sum of all the multiples of 3 or 5 below 1000.

11. Character Frequency : ASSASSINATION->a=3, s=4

Algo 1: high space complexicity
Algo2: space n time both moderate.
Algo 3: O(nlogn)

12. Armstrong number within given range

Playing with bits:

(refer q no 20)

13. Given a positive integer n, count the total number of all set bits in binary representation of all numbers from 1 to n.
14. Check if no. is power of 2 or not.

Conditional Problems

15. Maximum of three numbers.
16. To check leap year and print it.
17. To check if number is palindrome or not.
18. To return a digit at tens place at tens place.
19. Reversing a number
20. Given a number having only one'1' and all other 0's in its binary representation, find position of the only set bit.
21. Decimal to binary (method -> traditional)
22. Find hex equivalent of a given no.
23. If 3 sides of triangle are entered, write a program to check whether the triangle is isosceles, equilateral, scalar or right angled.
24. **Take 3 points from user & check if they are on straight line or not?

Series:

25. Write a function to calculate the series

    2 , 8/3 , 32/5 , 128/7 , 512/9 , 2048/11….

26. Fibonacci series.

    0 1 1 2 3 5 8 13 21

Arrays

27.Biggest and Lowest elements of an array
28.Sum of no. at even/odd indices in an array.
29.Shifting an element one location left or right.
30.Reversal of array (in place) .
31.Move smaller elements to LEFT and larger elements to RIGHT of pivot (from array). (normal and in place)
32.Swapping of two arrays (normal and in place).
33.Merging two arrays in third alternate element at a time.
34.Deletion of duplicate elements in an array.
35.Counting number of even no's & odd no's in an array.
36.Sort the array according to its frequency.
{2,3,2,4,5,12,2,3,3,3,12} then answer should be {3,3,3,3,2,2,2,12,12,4,5}

Matrices

37.Write an algorithm to add two 2*2 matrices.
38.Print/calculate sum of diagonal elements of 2-D array i.e. matrix
39.Largest and smallest values of a matrix (in a row , in col, in matrix)
40.Write an algorithm such that if an element in an MxN matrix is 0, its entire row and column is set to 0.
41.Print matrix in upper triangular form.

1	2	3	4		1	2	3	4
5	6	7	8	➔	0	6	7	8
9	10	11	12		0	0	11	12
13	14	15	16		0	0	0	16

42.Print matrix in lower triangular form.

Patterns

43. Floyd's Triangle:-

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

44. Write an algorithm to print:

```
5
454
34543
2345432
123454321
```

45. Print the pattern :

```
ABCDEFGFEDCBA
ABCDEF   FEDCBA
ABCDE      EDCBA
ABCD         DCBA
ABC            CBA
AB               BA
A                  A
```

46. Print the following pattern.

```
1
01
010
1010
10101
```

47. Pascal's Triangle (spaces are used to understand logic)

```
        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
```

Functions

48. To find if a string is cyclic rotation of the main string. Given a substring function which returns true if str2 is a substring of str1 and false if not.

e.g. Str1 = abcdef
Str2 = efabcd

Str2 is cyclic rotation of Str1.

Recursion

49.Write an algorithm to add first 'n' nos. using recursion. (an introduction to recursion)
50.Write a method that returns all subsets of a given set.
51.Factorial(with recursion)

Strings

52.Write an algorithm to find the lucky number of a person. (From the name of the person. First find the number corresponding to each alphabet in the name. Add these numbers till we get a single digit).

53.Sort a given string.
54.Write a function to check whether two given strings are anagram of each other or not. An anagram of a string is another string that contains same characters, only the order of characters can be different. For example, "abcd" and "dabc" are anagram of each other.
(Normal method n sort n check)

55.Write a program that will read a line & delete from it all occurrences of the word 'the'.

56.Achromatic string
E.g. String: State bank of India
o/p: SboI

57.**Sorting of strings.
58.***Arrange the words in dictionary format.(same as 57)

59. Lapindromes:

a. Gaga-→yes
b. Abcde→NO
c. Rotor→yes
d. Xyzxy→yes
e. Abbaab→no
f. Ababc→no

Random questions:

1. Swap the values of two variables without third variable, write minimum 3 methods with different logic.

2. (3-d array & function)

   How Will u use 3-D array to map our college library's book arrangement system.

/**********END**********/