Q1. Digit DP #1

Digit DP can be used to solve the following types of problems:

NP hard problems like Travelling salesman.

Any kind of DP problems except SOS dp problems.

Problems that require calculation of how many numbers between two values satisfying a particular property.

Digit DP is not really a DP technique and the term is a misnomer.

Q2. Digit DP #2

A digit DP algorithm can be applied to which of the following number systems :

Decimal Number system

Hexadecimal Number system

Binary Number system

All of the above.

Q3. Digit DP #3

Find the count of numbers X with less than N digits such that the X is divisible by 3. Algorithm:

```
Initialize Count to 0.
Consider X such that X has less than N digits.
If X is divisible by 3 increment Count.
output Count.
```

What is the time complexity of this brute force algorithm.

O(N^2logN)

O(N)

O(10^N)

O(3^N)

Q4. Digit DP #4

Find the count of numbers X with less than N digits such that the X is divisible by 3.

We know that a number is divisible by 3 if the sum of digits of the number mod 3 is 0.

We define:

DP[i][0] as count of numbers with less than equal to i digits and sum of digits mod 3 is 0.

DP[i][1] as count of numbers with less than equal to i digits and sum of digits mod 3 is 1.

DP[i][2] as count of numbers with less than equal to i digits and sum of digits mod 3 is 2.

Choose the correct recurrence:

DP[i][0] = 4*DP[i-1][0] + 3*DP[i-1][1] + 3*DP[i-1][2]

DP[i][1] = 4*DP[i-1][1] + 3*DP[i-1][0] + 3*DP[i-1][2]

DP[i][2] = 4*DP[i-1][2] + 3*DP[i-1][0] + 3*DP[i-1][1]

ALL OF THE ABOVE

Q5. Digit DP #5

What is the time and space complexity respectively of the above Digit DP algorithm.

O(N*logN) and O(N)

O(N*N) and O(N*N)

O(N) and O(N)

O(logN) and O(N)

Q6. Digit DP #6

P1 : count of numbers with atmost 3 non-zero digits in range L to R.

`P2 : Number with the maximum product of digits in range L to R.`

Technique : First find the Answer from 1 to R that is F[R] then find Answer from 1 to L - 1 that is F[L-1].

The solution to orignal problem is F[R] - F[L-1].

The technique can be applied to both P1 and P2.

Can be applied only to P1.

Can be applied only to P2.

The technique can be applied to neither of the problems.

Q7. Digit DP #7

You need to find the count of numbers in range L to R which have their sum of digits equal to S.

Let F[X] represent count of numbers in range 1 to X. It will suffice to find out F[R] - F[L-1].

Say X has D digits then any number less than or equal to X will have D digits(with possible leading zeros). We wish to apply digit DP to solve this problem.

Which of the following is a way discussed in the course to avoid counting numbers that are greater than X.

Use two different DP tables to help obey the upper limit.

Use a brute force and iterate from 1 to X so as to avoid greater numbers.

Use a boolean variable tight in the DP state which helps us to obey the upper limit.

None of the above.

Q8. Digit DP #8

You need to find the count of numbers in range L to R which have their sum of digits equal to S.

Let F[X] represent count of numbers in range 1 to X. It will suffice to find out F[R] - F[L-1].

Say X has D digits then any number less than or equal to X will have D digits(with possible leading zeros). We wish to apply digit DP to solve this problem.

Define **DP(pos, S, tight) :** count of numbers with initial (pos - 1) digits set and a sum of S is expected from the remaining digits. Here tight is 1 if initial (pos - 1) digits are the same as initial (pos - 1) digits of X, 0 otherwise.

NOTE : p is the digit of X at the pos position.

Choose the most appropriate option:

If tight is 0, then we may choose any number from 0 to 9 at the position pos else we may only use numbers from 0 to p.

if we choose the digit i at the position pos and i < p or tight = 0 then the problem reduces to DP(pos + 1, S - i, 0).

if we choose the digit i at the position pos and i = p and tight = 1 then the problem reduces to DP(pos + 1, S - i, 1).

All of the above.

Q9. Digit DP #9

For the digit DP algorithm discussed for the previous problem what is the time and space complexity respectively given that the in range [L, R], given R has D digits and we need numbers whose sum of digits is S.

O(DS) and O(D)

O(D) and O(DS)

O(DS) and O(DS)

O(S) and O(D)

Q10. Digit DP #10

n reference to coding video on Maximum Product digit DP: which of the following do we maintain:

A 3D dp for position, lower tight and upper tight.

A 4D dp for position, lower tight, upper tight and leading zero indicator

A 4D dp for position, lower tight, upper tight and middle tight.

A 3D dp for product of digit, lower tight and upper tight.

Q11. Digit DP #11

In reference to the video on Magic Numbers digit DP.

Choose which of the following is a 5-Magic Number:

1525365

55

853545

855545

Q12. Digit DP #12

In reference to the video on Magic Numbers digit DP.

We need to find all d-Magic numbers that are a multiple of m in range [a, b] under the modulo p = 10^9 + 7.

We reduce the problem to finding F[b] - F[a-1], where F[x] is the count of d-Magic numbers in the range [1, x] modulo p.

Which of the following is correct :

answer is (F[b] - F[a-1] + p) % p.

This problem cannot be reduced using this technique.

answer is (F[b] - F[a-1]) % p.

answer is F[b] - F[a-1].