Ignore the bounds

Luis is seeing his son playing, he ask him gently in what the game consists, the boy replies "Its like this, you have a big number, a bound and a "mod" (the remainder of a division between a number and "mod"), the goal of the game is to discover the maximum sub-number you can create following this rules:"

- The sequence must not decrease lower from the first digit taken.
- The sequence chosen must *not* reach the bound. By example, if the first digit is 'd' and a bound 'k', the range you can take is between [d, min(d+k, 9)]
- You can start from *any* digit of a number.
- For any given start point, you should look for the sub-numbers making the maximum sum applied to the mod operation.

Luis, astonished by the explanation, request his son to give him and example, the boy then continues: "Suppose a number like this: 56789, a bound of 2, and a mod 10. you start with 5, being the bound of 2, this mean you can take up to the digit 7 (this means that you can always collect as many fives, sixes and sevens following the rules explained). The sub-number formed is of "5+6+7", following the rules, we will have all others sub-numbers: "6+7+8" "7+8+9" "8+9" and "9", after applying the operation, you will find that the maximum remainder of the sub-number's sum will be of "9", made by the sub-number "9".

Luis is a former programmer now and he does not have the same ability he had years ago, please, help him in his task following the game previously defined.

Input

The first line of input will contain an integer T denoting the T test cases, then, T cases will follow. Each of the following cases will contain a line with an integer L, a big number N in the range $[10^{(L-1)}, (10^L)-1]$, an integer K denoting the bound and the M that will be the mod of the whole operation.

Output

Output the string "Scenario #i: " where i is the test case you are analyzing followed by the maximum sum of the sub-sequence that can be formed.

Example

INPUT	OUTPUT
4	Scenario #1: 8
7 1235678 2 10	Scenario #2: 2
7 1235678 2 3	Scenario #3: 16
3 679 2 20	Scenario #4: 9
4 3457 2 10	

Constraints

1<=L<=100000

10^(L-1)<=N<(10^L)-1

1<=K<=8

1<=M<=45