# Name : Shrinivas Hatyalikar

# Div: TY-B (B2)

# Roll No: 26

**Q)** Input: n and d  
  
Generate all decimal strings of length n which do not contain consecutive occurrences of digit d.  
  
Use dynamic programming.  
  
For example: if n = 3 and d = 2 then the output is   
00, 01, 02, ..., 999 except 22x, x22, where x is any digit.

**Code:**

#include <stdio.h>

#include <string.h>

#define MAX\_N 100

#define NUM\_DIGITS 10

long long dp[MAX\_N][NUM\_DIGITS][2];

// Function to generate and print valid strings

void generateStrings(int n, int d, int prev, int leadingZero, char\* current) {

    // Base case: If the string length is 0, print the current string.

    if (n == 0) {

        printf("%s, ", current);

        return;

    }

    // Iterate through all possible digits (0-9).

    for (int digit = 0; digit < NUM\_DIGITS; digit++) {

        // Skip consecutive occurrences of digit 'd'.

        if (digit == d && prev == d) continue;

        // If the current digit is 0 and we are not allowed leading zeros,

        // skip it unless it's the first digit (leadingZero).

        if (digit == 0 && !leadingZero) {

            current[strlen(current) - 1] = '\0'; // Remove the last character

            generateStrings(n - 1, d, digit, 0, current);

        } else {

            char newCurrent[MAX\_N];

            sprintf(newCurrent, "%s%d", current, digit);

            generateStrings(n - 1, d, digit, 1, newCurrent);

        }

    }

}

int main() {

    int n, d;

    printf("Enter the length of the strings (n): ");

    scanf("%d", &n);

    printf("Enter the digit to avoid consecutive occurrences (d): ");

    scanf("%d", &d);

    memset(dp, -1, sizeof(dp));

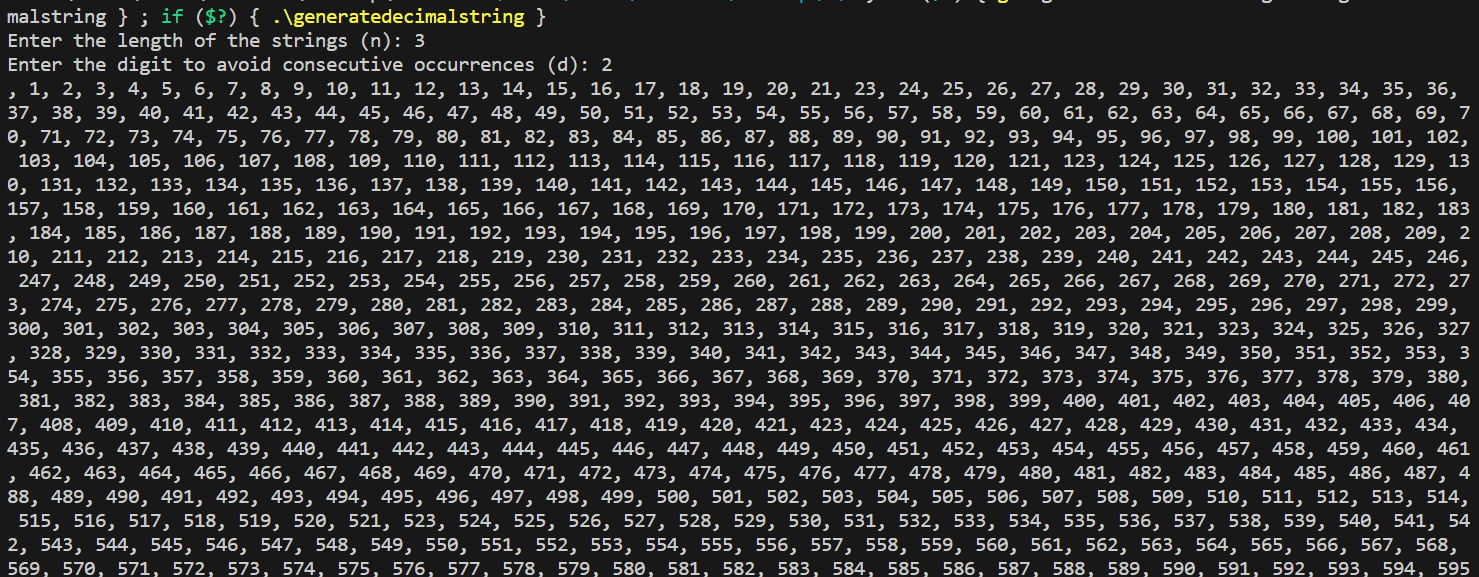
    char current[MAX\_N] = "";

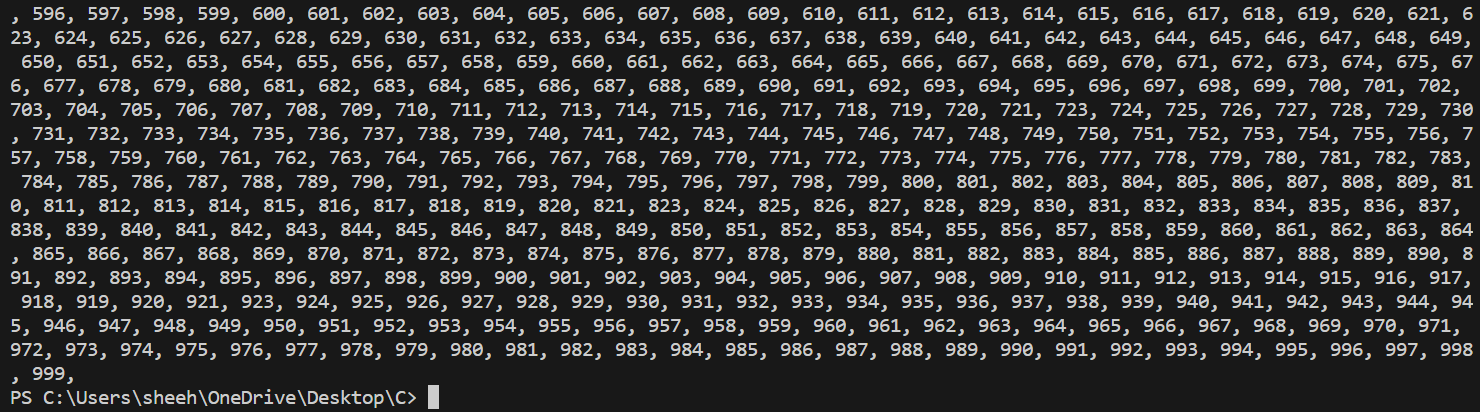
    generateStrings(n, d, -1, 0, current); // -1 for 'prev' means no previous digit.

    return 0;

}

**Output:**

****

****