Assignment Day3

Q1) Given two non-negative integers n1 and n2, where n1 is < n2 find all numebrs in the rnge without repete consecutive digits.

```
#include <stdio.h>
#include <stdbool.h>



bool has_repeated_consecutive_digits(int num) {
    int prev_digit = -1;
    while (num > 0) {
        int current_digit = num % 10;
        if (current_digit == prev_digit) {
            return true;
        }
        prev_digit = current_digit;
        num /= 10;
    }
    return false;
}

int main() {
    int n1, n2, count = 0;

    printf("Enter the value of n1: ");
    scanf("%d", &n1);
    printf("Enter the value of n2: ");
    scanf("%d", &n2);

    for (int i = n1; i <= n2; i++) {
        if (!has_repeated_consecutive_digits(i)) {
            count++;
        }
    }

    printf("Output: %d\n", count);
    return 0;
}
```

Q2) Write the code to generate the following pattern with the minimum lines of code.

```
#include <stdio.h>

int main() {
    int n = 4; // Number of rows
    for (int i = 1; i <= n; i++) {
        // Print leading spaces for alignment
        for (int j = 0; j < n - i; j++) {
            printf("  ");
        }
        // Print increasing characters
        for (int j = 0; j < i; j++) {
            printf("%c ", 'A' + 2 * j);
        }
        // Print decreasing characters
        for (int j = i - 2; j >= 0; j--) {
            printf("%c ", 'A' + 2 * j);
        }
        printf("\n");
    }
    return 0;
}
```

Q3) A supermarket maintains a pricing format for all its products. A value N is printed on each product. When the scanner reads the value N on the item, the product of all the digits in the value N is the price of the item. The task here is to design the software such that given the code of any item N the product (multiplication) of all the digits of value should be computed(price).

```
#include <stdio.h>

int main() {
  int n, product = 1;

  // Input the value of N
  printf("Enter the value of N: ");
  scanf("%d", &n);

  // Calculate the product of digits
  while (n > 0) {
    product *= n % 10; // Multiply the last digit
    n /= 10;       // Remove the last digit
  }

  // Output the result
  printf("Output: %d\n", product);
  return 0;
}
```

Q4) Write a program to find the nth term in this series.

Consider the following example: 0, 0, 2, 1, 4, 2, 6, 3, 8, 4, 10, 5, 12, 6, 14, 7, 16, 8

This series is a mixture of two series.

All the even terms in this series are derived from the previous term using the formula (x/2).

All the odd terms in this series form even numbers in ascending order.

The value n in a positive integer that should be read from STDIN the nth term that is calculated by the program should be written to STDOUT. Other than the value of the nth term no other characters /strings or message should be written to STDOUT.

```
#include <stdio.h>


int main() {
   int n;
   scanf("%d", &n);


   if (n % 2 == 0) {
      // Even index (e.g., 0th, 2nd, 4th, ...)
      printf("%d", (n / 2) * 2);
   } else {
      // Odd index (e.g., 1st, 3rd, 5th, ...)
      printf("%d", (n / 2));
   }


   return 0;
}
```

Q5) Write a program that does as follows. It takes a number as unput from the user. Then it takes a choice as input from the user. The the program does as follows. If choice is 1 -> check and display if the number is a prime number or not. If choice is 2 -> reverse the number aand display

```
#include <stdio.h>


// Function to check if a number is prime
int is_prime(int num) {
   if (num <= 1) return 0;  // 0 and 1 are not prime
   for (int i = 2; i * i <= num; i++) {
      if (num % i == 0) {
         return 0;  // Not prime
      }
   }
```

```
    return 1;  // Prime
}

int reverse_number(int num) {
    int reversed = 0;
    while (num != 0) {
        reversed = reversed * 10 + num % 10;
        num /= 10;
    }
    return reversed;
}

int main() {
    int number, choice;

    scanf("%d", &number);
    scanf("%d", &choice);

    if (choice == 1) {
        if (is_prime(number)) {
            printf("Prime");
        } else {
            printf("Not Prime");
        }
    } else if (choice == 2) {
        printf("%d", reverse_number(number));
    }

    return 0;
}
```