[CSE 6010] Week01 Assignment01

Name: Lu Chen

```c
#include <stdio.h>
#include <math.h>

int get_num_repete(int i, int m, int n);

int main(void)
{
    // get inputs from user

    // x - minimum range
    printf("Enter minimum of range: ");
    int x;
    scanf("%i", &x);

    //valid x
    if (x < 0)
    {
        printf("x needs to be a positive integer\n");
        return 1;
    }

    // y - maximum range
    printf("Enter maximum of range: ");
    //valid y
    int y;
    scanf("%i", &y);

    //valid y
    if (y < x)
    {
        printf("y has to be larger than x\n");
        return 2;
    }

    // m - specific digit of interest
    printf("Specific digit of interest: ");
    int m;
    scanf("%i", &m);

    //valid m
    if (m < 0 | m > 9)
    {
        printf("m needs to be an integer between 0 - 9");
        return 3;
    }

    // n - the number of repetitions of m
    printf("The number of repetitions of the digit: ");
    int n;
    scanf("%i", &n);

    //valid n
    if (x < 0)
    {
        printf("n needs to be a positive integer\n");
        return 4;
    }

    // compare whether repete
    int count = 0;

    //for each number
    for (int i = x; i <= y; i++)
    {
        int num = get_num_repete(i, m, n);

        count += num;

        //printf("flag1: %i, %i, %i", i, num, count);
    }

    // print out result
    printf("The number of integers from %i to %i that include exactly %i %i is %i.\n", x, y, n, m, count);

    return 0;
}
```

Main function

1. Use scanf() get x, y, m, n from the user

2. Use if() check if inputs are valid, if not, the code will return error messages and print out a message to remind user input correct inputs.

3. Use a for loop to iterate every number from x to y, and pass the number to function get_num_repete(). Use ‘count’ to count the n.

4. Print out the final result.

```c
int get_num_repete(int i, int m, int n)
{
    // count how many m
    int counter = 0;

    while (i)
    {
        if ((i % 10) == m)
        {
            counter++;
        }
        i /= 10;
    }

    if (counter != n)
    {
        return 0;
    }
    else
    {
        return 1;
    }
}
```

Function get_num_repete()

Use this function to examine whether the number contain exactly n m digits.

Use a while loop divide the number to separate digit, then use ‘counter’ count the number of digit equaling to m. If the counter == n, means the number contains exactly n m, return true. Otherwise, return false.

```
CSE6010/ $ make counting
CSE6010/ $ ./counting
Enter minimum of range: 0
Enter maximum of range: 13
Specific digit of interest: 3
The number of repetitions of the digit: 1
The number of integers from 0 to 13 that include exactly 1 3 is 2.
CSE6010/ $
```

The answer is correct.
The two number is 3 and 13.

```
CSE6010/ $ ./counting
Enter minimum of range: 10
Enter maximum of range: 100
Specific digit of interest: 3
The number of repetitions of the digit: 2
The number of integers from 10 to 100 that include exactly 2 3 is 1.
```

The answer is correct.
The two number is 33.

I feel like the ‘exactly n m’ condition is tricky and didn’t solve it at first. Then I realize I could use two individual counter to get that.

Also, if I only use one main() function, the structure of code would not be very clear, so I defined another function instead to check repetitive digits in each number.