

Problem Set 2 Exercise #10: Collatz Conjecture

Reference: Lecture 5 notes

Learning objective: Repetition statements

Estimated completion time: 30 minutes

Problem statement:

There is simple yet still unresolved question in computing, known as the Collatz Conjecture. The conjecture revolves around the following computation:

$$f(N) = \begin{cases} N/2, & \text{if } N \text{ is even} \\ 3 * N + 1, & \text{if } N \text{ is odd} \end{cases}$$

The Collatz Conjecture states that:

If we repeatedly apply the computation above, using the result of $f(N)$ as the next N , the process will *always* reduce to the value of 1.

For example, if N is 6:

$f(6) = 3$ (6 is even, $6/2 = 3$)

$f(3) = 10$ (3 is odd, $3*3+1 = 10$)

$f(10) = 5$

$f(5) = 16$

$f(16) = 8$

$f(8) = 4$

$f(4) = 2$

$f(2) = 1$ (Reached!)

It is very fascinating to note that the value of N has no known relationship with the number of steps required to reach the value of 1. For example, $N=6$ takes 8 steps but $N=27$ takes 111 steps! The number of steps required to reach the value of 1 is known as the **cycle length**.

Write a program **cycle_length.c** that (1) reads two positive integers *low* and *high* from command line; (2) calculates the cycle length for every number in the range $[low, high]$ (both inclusive); (3) prints out the largest cycle length.

For example, sample run #1 (next page) shows that the largest cycle length among all the integers in the range $[3, 6]$ is 8 (for $N=6$).

Modular design makes your programming more manageable. You are advised to divide the given task into several sub-problems and define functions to conquer them one by one.

Notes:

This task is a continuation of Exercise #09, in which you take a natural number and compute its cycle length. Therefore you may reuse the code written in the previous exercise.

After attempting these two exercises, you should reflect and have a better understanding of modular design – how it makes logic clearer and coding incremental.

Sample run #1:

```
Enter range: 3 6
Maximum cycle length = 8
```

Sample run #2:

```
Enter range: 20 27
Maximum cycle length = 111
```