Problem Set 2 Exercise #23: Square-free Integer

Reference: Lecture 5 notes

Learning objective: Repetition statements; Modular Design

Estimated completion time: 50 minutes

Problem statement:

[CS1010 AY2013/14 Semester 1 PE1 Ex2 Question]

In mathematics, a *square number* is an integer that is the square of a positive integer. Examples are 9 (= 3×3), 4 (= 2×2), and 1 (= 1×1). 1 is the smallest square number.

On the other hand, a *square-free integer* is a positive integer divisible by **NO** square number, except 1. For instance,

- 10 is a square-free number
- 18 is not square-free as it is divisible by a square number 9
- 4 is also not square-free as it is divisible by the square number 4

The first 10 square-free integers are:

1, 2, 3, 5, 6, 7, 10, 11, 13, 14

Note that 1 is both a square number and a square-free integer.

Write a program **square_free.c** to read four positive integers (in that sequence): *lower1*, *upper1*, *lower2*, *upper2*, compute the number of square-free integers in two ranges [*lower1*, *upper1*] (both inclusive) and [*lower2*, *upper2*] (both inclusive), compare and report which range has more square-free integers.

You may assume that:

$1 \le lower1 \le upper1$, and $1 \le lower2 \le upper2$

No input validation is needed.

For example, in the sample run #1 below, range [1, 5] contains 4 square-free integers while range [5, 9] contains 3 square-free integers. Therefore your program should print out:

```
Range [1, 5] has more square-free numbers: 4
```

Modular design makes your coding easier. Besides the **`main()`** function, your program must contain at least another function (of your own choice) to compute some results.

Sample run #1:

```
Enter four positive integers: 1 5 5 9
Range [1, 5] has more square-free numbers: 4
```

Sample run #2:

```
Enter four positive integers: 1 8 2 10
Both ranges have the same number of square-free numbers: 6
```