

Step 3

You may practise **Incremental Coding** (which Aaron has demonstrated in his week 4 lecture) here. The correctness of your final program depends on the correctness of **is_square_free()** function. So it is worthwhile checking that the function works before you proceed.

How do you think you can test the **is_square_free()** function?

Step 4

After you have ensured that **is_square_free()** is perfect, how do you use it to solve the task? Since you are going to do the same thing (count the number of square-free integers) for TWO ranges, you should write a function for it. Let's call it **count_square_free()**.

Complete the table about **count_square_free()** below.

<i>Return type</i>	<i>Parameters (types and names)</i>	<i>Precondition</i>

Step 5

Write out your algorithm for **count_square_free()**.

```
Algorithm:
_____ count_square_free( _____ ) {

}

}
```

Step 6

Complete your **main()** function and test your program thoroughly before you submit it to CodeCrunch.