

Exercise 2: Amicable numbers

[50 marks]

Problem Statement

In number theory, a pair of **amicable numbers** consists of two **different** numbers such that **the sum of the factors** of each is **equal to the other number**.

For example, 220 and 284 is a pair of amicable numbers:

- The factors of 220 are 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 and 110, of which the sum is 284.
- The factors of 284 are 1, 2, 4, 71 and 142, of which the sum is 220.

(Note: As shown in the example, we consider only the **proper** factors, which do **not** include the number itself.)

Given a range [lower, upper], we count a pair of amicable numbers as a *full pair* if both numbers in the pair are within the range. In contrast, we count a pair of amicable numbers as a *half pair* if only one of the numbers in the pair is within the range.

For example, if the range given is [100, 300], 220 and 284 is counted as a full pair because both numbers are within the range. In contrast, if the range given is [250, 300], the same pair is counted as a half pair, since only 284 is within the range.

In this exercise, you are to write a program to compute the number of full pairs and half pairs of amicable numbers in a given range.

For example, given the range [100, 300], your program should output that there is 1 full pair and 0 half pair of amicable numbers. As for the range [250, 300], your program should output that there is 0 full pair and 1 half pair of amicable numbers.

Your program should read in two integers, which represent the lower bound and upper bound of the range (both inclusive), compute the numbers of full pairs and half pairs of amicable numbers in the given range, and print these two numbers in the output messages.

You may assume that the input is valid (*i.e.*, the integers are all positive and the lower bounds are no bigger than the upper bounds).

Write on the skeleton file **amicable.c** given to you. You need to implement the following two functions:

- **void countPairs(int lower, int upper, int results[])**

This function takes in two integers `lower` and `upper`. It stores the number of full pairs of amicable numbers in the given range [lower, upper] in `results[0]`, and the number of half pairs in `results[1]`.

- **int sumFactors(int number)**

This function takes in one integer `number` and computes the sum of factors for `number`.

You may define additional functions as needed; however, **you are not allowed to change the main function**. Read the comments in the skeleton code for additional instructions.

Sample Runs

Four sample runs are shown below with user input highlighted in **bold**.

```
Enter range: 100 300  
Number of full pairs: 1  
Number of half pairs: 0
```

One pair of amicable numbers is counted: 220 and 284 (full).

```
Enter range: 250 300  
Number of full pairs: 0  
Number of half pairs: 1
```

One pair of amicable numbers is counted: 220 and 284 (half).

```
Enter range: 1 100  
Number of full pairs: 0  
Number of half pairs: 0
```

No integer within this range is part of a pair of amicable numbers.

```
Enter range: 200 1200  
Number of full pairs: 1  
Number of half pairs: 1
```

Two pairs of amicable numbers are counted: 220 and 284 (full), and 1184 and 1210 (half).