

Weekly Programming Challenge 3

Counting Bit-Sum-prime numbers

Let us first give the definition for *Bit-sum-prime*.

Definition : A positive integer is said to be Bit-sum-prime if the sum of all the bits in its binary representation is a prime number. Since bit can take value 1 or 0, so equivalent definition for Bit-sum-prime is a number whose binary presentation has prime number of 1's. For example, 6 is Bit-sum-prime since in its binary representation **110**, there are two 1's and 2 is prime number (also note that 2 is the smallest prime number).

So one can observe that 23 is not Bit-sum-prime, but 10 is Bit-sum-prime.

A mathematical function $\mathcal{P}(a, b)$ is defined for any two positive integers a, b with $a \leq b$ as follows.

$\mathcal{P}(a, b)$ = the number of integers between a and b inclusive which are Bit-sum-prime.

Examples are :

$$\mathcal{P}(4, 10) = 5$$

$$\mathcal{P}(137, 31415926535897) = 7753256197126$$

Write a JAVA program which computes $\mathcal{P}(a, b)$ for any two variables a and b of type **long**. Your program must run in at most a few seconds for any **long** values assigned to a and b .

Guidelines for submissions of solution:

1. The most important criteria for judging the solution will be **time**. Your program should not take more than a few seconds for every valid input. If your program takes minutes/hours for some inputs, then it is effectively a very poor program and does not deserve any credit.
2. The program **must** be well commented, **well** indented
3. You are recommended to use structured programming for this problem. This approach will really be very helpful. You should use right identifiers for methods which you use.
4. You should also explain the key idea underlying the solution/algorithm for the problem. It **must** not be more than a page.
5. Please **make sure** that your program **works** for all valid inputs. You should at least verify for the sample inputs mentioned above. That is, for $a=137$, and $b=31415926535897$, you program **must** output **7753256197126**.

Any submission which does not meet the above guidelines will be rejected without any notification. Please be **scrupulously honest** in your efforts to solve this programming challenge. This challenge does not require more than what has been taught in this course till now. Any other idea required for solving this challenge is well within your reach (you will realize it when I discuss its simple and elegant solution in some extra class). So please have faith in your capability and enjoy working on this challenge.