Prime Game

Problem C

Chin Ka Wang {rina__owo}

The 1st Buddhist Sin Tak College Computer Club Programming Contest

October 18, 2024

stpc();

Background

Problem Idea by `rina__owo, La Salle - Pui Ching Programming Challenge`
Preparation by `rina__owo, pepper1208`

Problem Restatement

Given *N* positive integers, containing at least two prime numbers.

Select two prime numbers, *a* and *b* from the given numbers.

Output the maximum possible value of *a* mod *b*.

Statistics

Points are given per subtask in this problem. There are 4 subtasks in this question.

Attempts: 29

0 points	12	+	0	=	12
Subtask 1 (16 points)	0	+	2	=	2
Subtask 2 (21 points)	0	+	1	=	1
Subtask 3 (27 points)	0	+	1	=	1
Subtask 4 (36 points)	0	+	2	=	2

First solved by **Lam, Pak Kiu Sean** at **31m 56s**

Solution

Observe that when $a < b$, $a \bmod b = a$, and $b \bmod a < a$.

Therefore if we want $a \bmod b$ to be the largest, a have to be as large as possible, and we need to select another number larger than it to be b.

Trivially, a would be the second largest prime number and b would be the largest prime number among the given numbers.

Note that there is a possibility for $a = b$ while they are both the largest. In this case, $a \bmod b$ would be 0.

Therefore we need to additionally detect whether $a = b$. We should select the second largest prime number which does not equal to b to be a.

Solution

Set up a `for loop` to iterate every number given.

For each number n_i, check whether n_i is a prime number using Primality test (Session 4).

Set up two variables `largest`, `second` to store the largest and the second largest prime number.

If $n_i >$ `largest`, put n_i into `largest`, else if $n_i >$ `second` and $n_i \neq$ `largest`, put n_i into `second`.

Output `second` mod `largest`.

Score: 100!

Non-mathematical solution

For each number n_i, check whether n_i is a prime number using Primality test (Session 4) and store all primes in an array.

Set up a two-layer `for loop` to iterate every pair of primes a and b in the array and set up a variable `largest` to store the largest answer.

Calculate a mod b and update `largest` when a mod $b >$ `largest`. b does not necessarily be larger than a in this case as the nested for loop iterates every combination of two primes. Be aware that the value of the two primes still cannot be the same.

Output `largest`.

Takeaways

1. Be familiar with code of primality test.
2. Try to think about the problem mathematically sometimes instead of doing it by brute force.