CALIFORNIA STATE SCIENCE FAIR
2008 PROJECT SUMMARY

Name(s)
Casey L. Fu

Project Number
S1305

Project Title
The Efficiency of Prime-Testing Algorithms

Abstract

Objectives/Goals
The purpose of this project is to determine the most efficient prime-testing algorithm from a widely used algorithm (control) and four algorithms I developed. The algorithms include:
1. dividing by odd numbers less than or equal to the square root of the number being tested (control)
2. applying divisibility rules
3. dividing by odd numbers less than or equal to the square root of the number tested that are not multiples of 5
4. dividing by odd numbers less than or equal to the square root of the number tested that are not multiples of 3
5. dividing by odd numbers less than or equal to the square root of the number tested that are not multiples of 3 or 5

Methods/Materials
A Java program that applies all of the five algorithms was written. The program takes an entered number and then determines and displays the next five prime numbers along with the time taken to do the calculations for each algorithm. Five numbers, randomly selected with a unit#s digit of 1, 3, 7, or 9 for each number length of twelve to nineteen digits, were entered in the program to be run by all five algorithms.

Results
Algorithm #5, which skips divisors that are multiples of 3 or 5, was the fastest and about 1.9 times as fast as the control. The second fastest was algorithm #4 (skipping multiples of 3), 1.5 times as fast as the control, followed by algorithm #3 (skipping multiples of 5), 1.3 times as fast as the control, and algorithm #2 (applying divisibility rules), 1.1 times as fast as the control.

Conclusions/Discussion
I hypothesized that the fastest algorithm was #2 (applying divisibility rules). My hypothesis was not supported because #2 turned out to be the second slowest. This is because it requires more calculations in each dividing step. The most efficient algorithm was #5, which indicates that by skipping divisors that are multiples of 3 or 5, the time for prime testing can be saved by about 47%. For further investigation, I will test a new algorithm that skips multiples of 3, 5, and 7 to see how much it improves the prime-testing efficiency.

Summary Statement
The goal of this project was to determine the most efficient prime-testing algorithm from a widely used algorithm and four that I created, and the most efficient algorithm was algorithm #5 (skipping divisors that are multiples of 3 or 5).

Help Received
Math teachers Mrs. Herrington and Mrs. Brown gave advice on how to improve my report. Mother taught some Java concepts.