CALIFORNIA STATE SCIENCE FAIR
2006 PROJECT SUMMARY

Name(s)

Frederick C. Dopfel

Project Number

S1205

Project Title

The Effect of a Low Precision Computational Environment on Comparative Algorithm Speed for Calculating the Value of Pi

Abstract

Objectives/Goals

The objective is to determine the performance of advanced algorithms for calculating the value of pi (such as the Borwein Quartic Convergence (BQC) algorithm) in a low-precision environment typical of modern personal computing applications. I suspect that such advanced algorithms designed for supercomputers at high precision may underperform simpler algorithms in a low-precision environment.

Methods/Materials

The algorithms that I chose to calculate pi were: Borwein Quartic Convergence algorithm (an advanced algorithm that is used in supercomputers at high precision); the arctangent formula; Viete's formula; Wallis's formula; and Monte Carlo approximation. I used C++ to code these algorithms and to constrain the precision to 14 digits. The program output included number of algorithm loops completed, the elapsed computation time and the estimated value of pi (error). Before running the programs, I stopped all background programs. The experiment was replicated numerous times to control for any aberrations in the performance of the computer. The computer itself was designed and home-built by me, and runs with a 3.0 GHz processor and 1.5 GB of RAM.

Results

The significant conclusion is that BQC, the fastest known algorithm at high precision, underperforms in a typical lower-precision computational environment, and "choked" at 12-digit accuracy, never converging to pi at 14-digit accuracy. The Viete algorithm, invented in the 16th century, outperformed BQC and other algorithms at 12-digits or more accuracy. Wallis' formula and the arctangent formula were very close in efficiency and the Monte Carlo was the slowest algorithm, as expected.

Conclusions/Discussion

BQC relies on repeated high-precision calculations of fourth roots and powers while the other algorithms tested rely more on multiplication and addition of simple fractions. Although BQC takes very few loops to converge, each loop is very costly in computational time and resources (82x more time per loop than the Viete algorithm). For these reasons, the BQC algorithm, while known to be the fastest in a high-precision computing environment, underperformed relative to simpler algorithms in a low-precision computing environment. Algorithms and software designed for supercomputers may not always perform well in typical computing environments. Software designers need to account for the computing environment when choosing the algorithms to code for complex tasks.

Summary Statement

My project tests how complex algorithms (such as those for calculating the value of pi) designed for high-precision computing environments perform in comparison to simpler algorithms in a low-precision computing environment.

Help Received

My school provided a free copy of Microsoft Visual Studio.