

ETEC3702 – Concurrency

Lab 7 – Multi-processing to Improve Performance of CPU-Bound Processes

Due date: 10 March 2020 by the end of class.

One method of computing a value for PI is the Leibniz series, which is composed of an infinite series of fractions with alternating signs.

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots = \frac{\pi}{4}$$

The more terms that are added, the closer the approximation is to the actual value of pi/4.

Program 1:

Implement a function called **pi_est_sequential(number)** that computes and returns an estimated value for PI by expanding the series up to the term with **number** in the denominator.

Write a main program that tests and times this function by calling it with a number of 1000000000.

Have the program display the execution time of the function and the resulting estimate for PI.

Program 2:

Write a new function called **pi_est_2proc(number)** that uses multi-processing and computes the solution by dividing the problem up into two different processes: the positive terms and the negative terms (or two separate ranges). These two processes should send their respective estimate values back and, when summed and multiplied by 4, these will be the resulting estimate of PI that your function should return.

Write a main program that tests and times this function by calling it with a number of 1000000000.

Have the program display the execution time of the function and the resulting estimate for PI.

Program 3:

Write a new function called **pi_est_4proc(number)** that uses multi-processing and computes the solution by dividing the problem up across 4 processes. Capture the results of each of these processes and sum them. Return this sum multiplied by 4 as the estimate result.

Write a main program that tests and times this function by calling it with a number of 1000000000.

Have the program display the execution time of the function and the resulting estimate for PI.

Questions:

Q1: What was the speedup computed of the 2 process version over the sequential version?

Q2: What was the speedup computed of the 4 process version over the sequential version?

Q3: What was the speedup computed of the 4 process version over the 2 process version?

Q4: How many cores does your system have? What is the theoretical maximum speedup attainable on your machine?