

Tasksheet 6:

Task 1. Computing an Approximation of π .

As indicated in class, we have come up with at least two ways to represent the number π . There are infinitely many ways to set up a calculation that results in the value, π . Pick a way to represent π that was not used in class and write a code that will approximate the value of π . Develop a method that uses one of the integration methods developed in class (e.g, the Trapezoid rule or Simpson's rule) to produce the approximation.

Note that this task does not require any programming or testing with a code.

Task 2. Now you get to write code.

Write a serial code - no parallelization - that will produce the approximation of π defined in Task 1 of this tasksheet. Use the optimization flags in the compiler (e.g,-O1 in C) and report the performance results from this option. You are welcome to use the code provided for the approach developed in class. The code written by your instructor in C. The modifications should be straight forward.

Task 3. Adding OpenMP to the mix.

Repeat the work done in class to parallelize the code from Task 2 using OpenMP. In a first attempt use only omp-directives to parallelize the code used in the approximation. You are welcome to implement code using the You Tube video tutorial found at [OpenMP](#).

Task 4. Another Example for Consideration - Approximation of e .

Find a formula that will produce the value of the number, e , similar to the approach used in our work in approximating π . Come up with an algorithm for the approximation. Build a code that will produce the approximation using your algorithm. Discuss the results you obtain.

Task 5. Linear Algebra Algorithms

For the rest of the semester, we will be working with algorithms from linear algebra. Write routines/ functions that implement the following operations on vectors and matrices

- vector addition
- vector subtraction
- scalar multiplication of a number and a vector
- the length or norm of a vector (there should be at least three of these)
- the dot product of two vectors
- the cross-product of two vectors
- the triple product of three vectors
- the action of a matrix, $A \in \mathbb{R}^{m \times n}$, and a vector of appropriate length.
- the sum of two matrices
- the difference of two matrices
- the produce of two matrices of appropriate size

Make sure you provide evidence that your codes work. You can create software manual pages, one for each routine, and link to these from your write up of this assignment.