Projects

Each program should be written using one of the techniques listed in the content of task. Each program should have a rich description of the code in the form of comments.

1. Design a program to simulate the lift. Passengers come to random floors and if the lift is idle, they call for it. When the lift is in motion, the passenger has to wait on his floor, until it becomes idle and then he can invoke it. It may happen that the lift will stop on the floor because someone is getting off. In such case it can take passengers, who are going in the same direction as the lift. The lift is idle on the floor where all the passengers leaves it and no one is waiting for it. The program should be made with the use of MPI.

2. Develop a concurrent program that calculates the approximate value of the definite integral by the trapezoidal rule. The parameter of the procedure should be the number of subintervals of integration. Measure the time of execution of the program for the different number of threads. The program should be made with the use of Pthreads.

$$I = 4\int_0^1 \frac{1}{1+x^2}\,dx$$

Helpful information:

$$\int_a^b f(x)dx \approx h\left(\frac{y_0+y_n}{2}+\sum_{i=1}^{n-1} y_i\right),$$

where:

$$h=\frac{b-a}{n},$$

$$y_0 = f(a) \qquad y_n = f(b),$$

$$y_i = f(x_i) \qquad x_i = a + i \cdot h,$$

n and h -number of subintervals and width of each subinterval.

3. Write a program that calculates the value of the expression using the Mclaurin series. Compare the obtained value with the value from the math library. Measure the time of execution of the program for the different number of threads. The program should be made with the use of MPI or Pthreads.

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

4. Euler gamma constant is defined as the limit of a sequence g_n where:

$$g_n = \lim_{n \to \infty} \left(\sum_{k=1}^{n} \frac{1}{k} - ln(n) \right)$$

Write a concurrent program that calculates g_n, with the following assumptions:
- N - the number of elements of the sum is loaded from the keyboard
- P - number of processes, which will compute the sum is a constant in a program

Each process calculate his own fragment of the sum - for example, for n = 100, p = 10, the first process computes elements 1 to 10, the second from 11 to 20, etc., at the end the partial sums are added and logarithm is subtracted. Note: n may not be divisible by p. The program can be made using any familliar technique .

5. Write a parallel program that solves the following system of linear equations with the use of OpenMP:

$a_{11}x_1 \qquad\qquad\qquad = b_1$

$a_{21}x_1 + a_{22}x_2 \qquad\qquad = b_2$

....

$a_{n1}x_1 + a_{n2}x_2 + ... + a_{nn}x_n \qquad = b_n$

6. Write a program that calculates the prime numbers using the sieve of Erathosthenes. The program should be made with the use of OpenMP.

example:

We have a sequence of natural numbers:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ...

At first, we eliminate multiples of 2 by which we get:

1, 2, 3, * 5, * 7, * 9, * 11, * 13, ...

In the next step we remove multiples of 3:

1, 2, 3, * 5, * 7, *, *, 11 *, 13, ...

Further we delete the multiple of 5, 7 etc.