**Лабораторная работа**

***Создание многопоточного приложения***

Цель работы: получить практические навыки по использованию потоков *для организации высоконагруженных вычислений*.

**Ход работы:**

1. Ознакомиться с принципами многопоточности.

2. В среде Visual Studio разработать консольное приложение с++, выполняющим действия в соответствии с индивидуальным вариантом.

***Вариант 1.*** Реализовать **два варианта** вычисления функции F(X, N):  
1) непараллельный (последовательный код);  
2) параллельный с использованием потоков из библиотеки С++;

Формула для функции F:

F(X, N) = ΣNI=1 1/(ΣIJ=1 (J + (X + J)1/3)/(2IJ − 1))

Cделать замеры времени выполнения вычислительного кода для трёх вариантов для разного количества ядер (от 1 до 8), по результатам замеров построить графическую зависимость прироста производительности от количества ядер, объяснить полученные результаты.

В виде текстовых сообщений в консоль отобразить ход вычислительного процесса:  
а) в процентах (с шагом 5-10 процентов);  
б) оставшееся время выполнения.

С помощью средств логгирования реализовать запись в файл результатов при каждом запуске приложения.

***Вариант 2.*** Реализовать **два варианта** вычисления функции F(X, N):  
1) непараллельный (последовательный код);  
2) параллельный с использованием потоков из библиотеки С++;

Формула для функции F:

F(X, N) = ΣNI=1 1/(ΣI+NJ=1 (J + (X + J)1/5)/(2IJ − 1))

Cделать замеры времени выполнения вычислительного кода для трёх вариантов для разного количества ядер (от 1 до 8), по результатам замеров построить графическую зависимость прироста производительности от количества ядер, объяснить полученные результаты.

В виде текстовых сообщений в консоль отобразить ход вычислительного процесса:  
а) в процентах (с шагом 5-10 процентов);  
б) оставшееся время выполнения.

С помощью средств логгирования реализовать запись в файл результатов при каждом запуске приложения.

***Вариант 3.*** Реализовать **два варианта** вычисления функции F(X, N):  
1) непараллельный (последовательный код);  
2) параллельный с использованием потоков из библиотеки С++;

Формула для функции F:

F(X, N) = ΣNI=1 1/(ΣNJ=I (J + (X + J)1/7)/(2IJ − 1))

Cделать замеры времени выполнения вычислительного кода для трёх вариантов для разного количества ядер (от 1 до 8), по результатам замеров построить графическую зависимость прироста производительности от количества ядер, объяснить полученные результаты.

В виде текстовых сообщений в консоль отобразить ход вычислительного процесса:  
а) в процентах (с шагом 5-10 процентов);  
б) оставшееся время выполнения.

С помощью средств логгирования реализовать запись в файл результатов при каждом запуске приложения.

***Вариант 4.*** Реализовать **два варианта** вычисления функции F(X, N):  
1) непараллельный (последовательный код);  
2) параллельный с использованием потоков из библиотеки С++;

Формула для функции F:

F(X, N) = ΣNI=1 1/(Σ2NJ=I (J + (X + J)1/4)/(2IJ − 1))

Cделать замеры времени выполнения вычислительного кода для трёх вариантов для разного количества ядер (от 1 до 8), по результатам замеров построить графическую зависимость прироста производительности от количества ядер, объяснить полученные результаты.

В виде текстовых сообщений в консоль отобразить ход вычислительного процесса:  
а) в процентах (с шагом 5-10 процентов);  
б) оставшееся время выполнения.

С помощью средств логгирования реализовать запись в файл результатов при каждом запуске приложения.

***Вариант 5.*** Реализовать **два варианта** вычисления функции F(X, N):  
1) непараллельный (последовательный код);  
2) параллельный с использованием потоков из библиотеки С++;

Формула для функции F:

F(X, N) = ΣNI=1 1/(Σ2N+IJ=I (J + (X + J)1/8)/(2IJ − 1))

Cделать замеры времени выполнения вычислительного кода для трёх вариантов для разного количества ядер (от 1 до 8), по результатам замеров построить графическую зависимость прироста производительности от количества ядер, объяснить полученные результаты.

В виде текстовых сообщений в консоль отобразить ход вычислительного процесса:  
а) в процентах (с шагом 5-10 процентов);  
б) оставшееся время выполнения.

С помощью средств логгирования реализовать запись в файл результатов при каждом запуске приложения.

***Вариант 6.*** Реализовать **два варианта** вычисления функции F(X, N):  
1) непараллельный (последовательный код);  
2) параллельный с использованием потоков из библиотеки С++;

Формула для функции F:

F(X, N) = ΣNI=1 1/(ΣN+IJ=I (J + (X + J)1/5)/(2IJ − 1))

Cделать замеры времени выполнения вычислительного кода для трёх вариантов для разного количества ядер (от 1 до 8), по результатам замеров построить графическую зависимость прироста производительности от количества ядер, объяснить полученные результаты.

В виде текстовых сообщений в консоль отобразить ход вычислительного процесса:  
а) в процентах (с шагом 5-10 процентов);  
б) оставшееся время выполнения.

С помощью средств логгирования реализовать запись в файл результатов при каждом запуске приложения.