

## ЛАБОРАТОРНАЯ РАБОТА № 10

### «Многопоточное программирование»

Цель работы: ознакомиться с организацией многопоточной обработки данных, получить основные навыки программирования с использованием потоков и работы с визуальными компонентами.

#### *Лабораторное задание*

Разработать консольное и визуальное приложение согласно варианту. Требования к программе: реализовать возможность задавать приоритет каждого из порожденных потоков; использовать символы псевдографики для визуализации потоков на экране монитора.

**1** Умножение матрицы на вектор. Обработку одной строки матрицы производить в порожденном потоке.

**2** Поиск всех простых чисел (простым называется число, которое является своим наибольшим делителем) в указанном интервале чисел, разделенном на несколько диапазонов. Обработка каждого диапазона производится в порожденном потоке. Классический алгоритм Евклида определения наибольшего общего делителя двух целых чисел ( $x$ ,  $y$ ) может применяться при следующих условиях:

- оба числа  $x$  и  $y$  неотрицательные;
- оба числа  $x$  и  $y$  отличны от нуля.

На каждом шаге алгоритма выполняются сравнения:

- если  $x == y$ , то ответ найден;
- если  $x < y$ , то  $y$  заменяется значением  $y - x$ ;
- если  $x > y$ , то  $x$  заменяется значением  $x - y$ .

**3** Винни-Пух и пчелы. Заданное количество пчел добывают мед равными порциями, задерживаясь в пути на случайное время. Винни-Пух потребляет мед порциями заданной величины за заданное время и столько же времени может прожить без питания. Работа каждой пчелы реализуется в порожденном потоке.

**4** Шарики. Координаты заданного количества шариков изменяются на случайную величину по вертикали и горизонтали. При выпадении шарика за нижнюю границу допустимой области шарик исчезает. Изменение координат каждого шарика в отдельном потоке.

**5** Противостояние нескольких команд. Каждая команда увеличивается на случайное количество бойцов и убивает случайное количество бойцов участника. Борьба каждой команды реализуется в отдельном потоке.

**6** Контрольная сумма. Для нескольких файлов (разного размера) требуется вычислить контрольную сумму (сумму кодов всех символов файла). Обработка каждого файла выполняется в отдельном потоке.

**7** Бег с препятствиями. Создается условная карта трассы в виде матрицы, ширина которой соответствует количеству бегунов, а высота фиксирована, которая содержит произвольное количество единиц (препятствий) в произвольных ячейках. Стартующие бегуны (потоки) перемещаются по трассе и при встрече с препятствием задерживаются на фиксированное время. По достижении финиша бегуны сообщают свой номер.

**8** Создать два потока. Первый ищет числа Фибоначчи (каждое последующее число равно сумме двух предыдущих чисел), второй – простые числа. Результат работы каждого потока сохраняется в отдельный файл. После остановки потока программа производит анализ файлов, выводит их на экран, а так же показывает количество найденных чисел Фибоначчи и простых чисел.

**9** Создать два потока. Первый поток производит запись в файл случайных данных, второй производит чтение данных из этого файла и вывод их на экран.

**10** Создать приложение, выполняющее вычисление значений функции  $y = 23 \cdot x^2 - 33$ , с шагом  $x=0.01$ . Первый поток выполняет расчёт функции и добавляет результаты расчёта в конец массива. Второй поток извлекает из массива значения  $x$  и  $y$  и выводит их на экран.

**11** Создать приложение, которое выполняет сортировку массива данных и отображает процесс сортировки на экране. Первый поток производит сортировку по возрастанию, второй – по убыванию. После каждого перемещения элементов производится вывод на экран текущего состояния сортировки. Каждый поток работает с отдельным экземпляром массива данных.

**12** Создать игру, где будут 2...3 барана и волк. При совпадении координат волка с бараном баран исчезает. При совпадении координат баранов появляется новый баран. Все движутся хаотически.

**13** Создать три потока, генерирующих случайным образом целые числа от 0 до 9. При нажатии на клавишу «Enter» потоки останавливаются и результат анализируется. Цель анализа – выявить наличие следующих комбинаций цифр в потоках: три одинаковых числа, два одинаковых числа, три единицы, три семерки, две единицы.

**14** Создать три потока, каждый из которых управляет перемещением псевдосимвола на экране вдоль оси X, и устроить «тараканьи бега» среди них.

**15** Поиск указанной строки в указанном файле. Обработка одной строки в порожденном потоке.