Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

Московский авиационный институт

(национальный исследовательский университет)

**ЛАБОРАТОРНАЯ РАБОТА №3**

по курсу

«Операционные системы»

Студент: Эссаулов А.А. (№31)

Группа: М80-207Б-18

Преподаватель: Миронов Евгений Сергеевич

Оценка:

Дата:

**Цель работы:**

Целью является приобретение практических навыков в:

⦁ Управление потоками в ОС

⦁ Обеспечение синхронизации между потоками

**Задание (вариант № 23):**

Коды Хэмминга. Необходимо реализовать 2 программы. Первая программа кодирует по Хэммингу нормальную последовательность байт. Вторая программа раскодирует и корректирует закодированную последовательность (корректировка осуществляется только, если было повреждение). Многопоточность необходимо реализовать только в программе 2 (дешифрование и коррекция кодов). Количество контрольных разрядов должно задаваться пользователем.

**Принцип работы:**

Программа coder.exe - кодирует последовательность бит по коду Хэммингу. Количество контрольных бит задается пользователем. В случае, если длина последовательности бит больше чем покрывают контрольные биты, то последовательность разбивается на сообщения. Результат записывается в файл.

Программа decoder.txt - декодирует последовательность бит из файла и проверяет на наличие ошибки и исправляет, если она есть. При обработке сообщения для каждого контрольного бита в сообщении запускается отдельный поток в которой передается ссылка на счётчик позиции ошибочного бита. Обработка заключается в пересчете значения контрольного бита и сравнении с уже записанным и если они не равны, то увеличиваем счетчик на значение позиции.

**Вывод:**

Потоки позволяют использовать ресурсы компьютера максимально эффективно. Распараллеливание задач, где это возможно может дать значительный прирост производительности.

В случае с Кодом Хэмминга мы получаем возможность одновременно вычислять значения контрольных битов. В случае больших сообщений закодированных большим количеством контрольных бит - можно получить существенный прирост производительности.

Ловля ошибок при работе с потоками заставляет нас учитывать какой поток сейчас находится в работе и с какими данными он работает. Так как дебаг выводами в консоль в данном случае становится неэффективным, то становятся полезными внешние инструменты работы с программой в рантайме.

Также обязательно надо учитывать специфику ОС и её типы данных. Основные проблемы возникают при приведении одного специфичного для ОС типа данных в другой.