

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ (НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

ОТЧЕТ
О ВЫПОЛНЕНИИ 1 ЛАБОРАТОРНОЙ РАБОТЫ
ПО ДИСЦИПЛИНЕ «ОПЕРАЦИОННЫЕ СИСТЕМЫ»

Выполнил(а) студент
группы М8О-208Б-23
Федорова Екатерина Васильевна

Проверили и приняли:
Живалев Е.А.
Катаев Ю. И.

Москва, 2024

Тема: «Первая программа»

Цели работы:

Приобретение практических навыков в:

- Управление процессами в ОС
- Обеспечение обмена данных между процессами посредством каналов

Задание (вариант №22):

Родительский процесс создает два дочерних процесса. Первой строкой пользователь в консоль родительского процесса вводит имя файла, которое будет использовано для открытия File с таким именем на запись для child1. Аналогично для второй строки и процесса child2. Родительский и дочерний процесс должны быть представлены разными программами. Родительский процесс принимает от пользователя строки произвольной длины и пересылает их в pipe1 или в pipe2 в зависимости от правила фильтрации. Процесс child1 и child2 производят работу над строками. Процессы пишут результаты своей работы в стандартный вывод.

Правило фильтрации: с вероятностью 80% строки отправляются в pipe1, иначе в pipe2.

Обработка строк: Дочерние процессы инвертируют строки.

Идея, метод, алгоритм решения задачи:

На вход функции ParentRoutine подаются пути к исполняемым файлам дочерних процессов и поток ввода. Родительский процесс считывает из него названия файлов, создаёт два дочерних процесса и передаёт им по названию. Также создаются два канала для общения с соответствующими процессами: родительский только на записи, дочерние только на чтение. Родительский процесс продолжает считывать строки из потока и псевдорандомно записывать их в каналы pipe1 или pipe2; дочерние процессы читают переданные по каналу строки, инверсируют их и записывают в выходной файл.

Вывод:

Работа позволила освоить базовые принципы многопроцессного программирования, работу с каналами межпроцессного взаимодействия и систему сборки CMake. В процессе выполнения были изучены основы C++, библиотека Google Test для модульного тестирования. Несмотря на некоторые сложности при написании тестов, поставленные цели были достигнуты, что создает хорошую основу для дальнейшего изучения многопроцессного программирования.