

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

ОТЧЕТ
О ВЫПОЛНЕНИИ КУРСОВОГО ПРОЕКТА
ПО ДИСЦИПЛИНЕ «ОПЕРАЦИОННЫЕ СИСТЕМЫ»

Выполнил(а) студент

группы М8О-208Б-23

Романов Вадим Михайлович

Проверили и приняли:

Живалев Е.А.

Катаев Ю. И.

Москва, 2024

Тема: «Разработка планировщика DAG джобов»

Цель работы: Целью курсового проекта являлось приобретение практических навыков в:

- разработке планировщиков задач на основе направленных ациклических графов (DAG);
- работе с конфигурационными файлами (JSON);
- реализации многопоточности и примитивов синхронизации.

Вариант: 39. Ini/Barrier.

Задачи проекта:

1. Реализовать проверку DAG на корректность, включая отсутствие циклов и наличие одной связной компоненты.
2. Организовать выполнение задач (джобов) DAG с учётом их зависимостей.
3. Реализовать параллельное выполнение джобов с ограничением максимального количества одновременно выполняемых задач.
4. Реализовать поддержку синхронизации через барьеры с возможностью настройки через конфигурационный файл.

Описание решения: Программное решение разработано на языке C++ с использованием библиотеки pthread для работы с потоками. Основные компоненты системы:

1. Проверка корректности DAG:

- На этапе загрузки конфигурационного файла происходит построение графа джобов.
- Реализована проверка графа на наличие циклов с использованием алгоритма обхода в глубину.

- Проверяется наличие стартовых и завершающих джобов, а также единственность связной компоненты.

2. Организация выполнения DAG:

- Для каждой джобы хранится информация о её зависимостях и семафоре (если он указан).
- Выполнение начинается с джобов, не имеющих зависимостей (стартовые).
- Задачи DAG запускаются в отдельном потоке, и их выполнение координируется с использованием мьютексов и условных переменных.

3. Параллельное выполнение джобов:

- Максимальное количество одновременно выполняемых задач задаётся параметром, передаваемым программе.
- Реализована очередь готовых к выполнению задач, которая координируется через условные переменные.

4. Поддержка барьеров:

А её-то у меня и нет

Использование программы:

1. Конфигурационный файл в формате Ini задаёт описание джобов и их зависимостей. Пример:

```
[job1]
```

```
dependencies = job2
```

```
[job2]
```

```
dependencies = job3
```

[job3]

dependencies = job1

2. Программа принимает на вход имя конфигурационного файла и максимальное число параллельных задач.
3. После проверки DAG запускается выполнение джобов с учётом их зависимостей и ограничений.

Заключение:

В результате работы программы были достигнуты все поставленные цели. Реализован планировщик DAG с поддержкой параллельного выполнения. Программа успешно проверяет корректность графа и выполняет задачи с учётом заданных ограничений. Получены навыки работы с многопоточностью и обработкой графовых структур.

Вывод:

Данная работа далась намного легче предыдущей, хоть у меня и не получилось добавить в программу барьеры. Никакие способы отладки мне в этом не помогли, что очень печально. Однако программа работает и сдачу прошла. На этом мои мытарства с операционными системами завершены.