# ВВЕДЕНИЕ

Данная курсовая работа на тему: «Обслуживание процессором ЭВМ очереди готовых заданий» ориентирована на изучение концепций построения операционных систем и методов управления ресурсами процессора. Данной курсовой работой, закреплены знания по следующий дисциплинам: «Программирование», «Объектно-ориентированное программирование».

Целью курсовой работы является изучение основных методов, используемых при управлении ресурсами в различных операционных системах.

Задачей курсовой работы является получение, как теоретических знаний, так и практических навыков, достаточных для проектирования и программирования системного программного обеспечения современных компьютеров, ознакомление с проблемами моделирования и анализа эффективности функционирования реальных вычислительных систем.

**Разработка приложения**

Для генерации процессов создадим классы ExTask, TimerTaskProcess. TimerTaskProcess вызывает класс GeneratingOfProcess, который в свою очередь получает имя процесса, и данный процесс попадает в ArrayList в классе Processes

Следующим классом будет ClassRealizationWork, который также наследует от класса TimerTask и запускает в классе Processes метод Work. Метод Work ищет наиболее подходящий процесс для “втиснения” в память. В данном методе определяется будет ли подходить процесс для выполнения, в противном случае процесс попадает в список отклоненных процессов.

При присвоении процессу состояния «Выполненный», в блоке памяти создаётся блок и процесс начинает работу, после того как процесс закончил работу, он удаляется из блока памяти. В классе MemoryScheduler есть метод добавления блока памяти и метод удаления блока памяти. Также есть классы ConfirmedQueue, где хранятся выполненные процессы и RejectedQueue, где хранятся не выполненные процессы, которые не прошли провеку. Создадим класс Utils, который содержит 3 константные переменные типа int: maxMemorySize, maxPriority, minPriority. Дальше понадобится Enum, который будет содержать такие типы процессов: «Открытие», «Закрытие», , «Он», «Офф».

**Практическое использование**

В интерфейсе программы можно увидеть элемент TabPane который состоит из четырёх Tab, на первом находится таблица TableView в которой можно увидеть следующие данные процессов: №, Name , Tacts , Time Of Process, Status , Priority и Memory. Следующий Tab содержит таблицу TableView, в которой выводится список принятых подтвержденных процессов, там можно увидеть столбцы: №, Name , Tacts, Time Of Process , Status, Priority и Memory . В третьем элементе TabPane`а также можно увидеть таблицу TableView, но уже отклонённых процессов, там можно увидеть столбцы: №, Name , Status и Memory.

**Вывод**

В данной курсовой работе были изучены основы построения операционных систем, управления ресурсами, управление очередью заданий, планировка заданий, выбор задания для выполнения по приоритету. Для выполнения курсовой работы, нам понадобились теоретические знания с курса «Современные технологии программирования», а именно многопоточность, интерфейсы GUI и управления элементами графического интерфейса, такими как TableView, TabPane.