

МГТУ им. БАУМАНА

ЛАБОРАТОРНАЯ РАБОТА №1 ч.2.

По курсу: "ОПЕРАЦИОННЫЕ СИСТЕМЫ"

**Функции обработчика системного
таймера. Пересчет динамических
приоритетов**

Работу выполнил: Мокеев Даниил, ИУ7-56

Преподаватель: Рязанова Н.Ю.

Москва, 2020

1 | Функции обработчика прерываний от системного таймера

1.1 Unix

1.1.1 По тикку

- Инкремент счетчика использования процессора текущим процессом;
- инкремент часов и других таймеров системы;
- декремент счетчика времени, оставшегося до отправления на выполнение отложенных вызовов и отправка отложенных вызовов на выполнение, при достижении нулевого значения счетчика;
- декремент кванта.

1.1.2 По главному тикку

- Добавление в очередь отложенных вызовов функций планировщик;
- пробуждение системных процессов swapper и pagedaemon;
- декремент счетчиков времени, оставшегося до отправления сигналов тревоги:
 - SIGALRM — сигнал будильника реального времени, который отправляется по истечении заданного промежутка реального времени;
 - SIGPROF — сигнал будильника профиля процесса, который измеряет время работы процесса;
 - SIGVTALRM — сигнал будильника виртуального времени, который измеряет время работы процесса в режиме задачи.

1.1.3 По кванту

- При превышении текущим процессом выделенного кванта, отправка сигнала SIGXCPU этому процессу.

1.2 Windows

1.2.1 По тикку

- Инкремент счетчика системного времени;
- декремент счетчиков отложенных задач;
- декремент остатка кванта текущего потока;
- активация обработчика ловушки профилирования ядра.

1.2.2 По главному тикку

- Инициализация диспетчера настройки баланса путем освобождения объекта «событие», на котором он ожидает.

1.2.3 По кванту

- Инициализация диспетчеризации потоков путем добавления соответствующего объекта DPC в очередь.

2 | Пересчет динамических приоритетов

2.1 Unix

В Unix планировщик предоставляет процессор каждому процессу системы на небольшой период времени, после чего производит переключение на следующий процесс. Этот период называется квантом времени.

Переключение контекста - на самом низком уровне планировщик заставляет процессор производить переключения от одного процесса к другому.

Классическое ядро UNIX является строго невытесняемым. Это означает, что если процесс выполняется в режиме ядра, то ядро не заставит этот процесс уступить процессорное время какому-либо более приоритетному процессу. Выполняющийся процесс может освободить процессор в случае своего блокирования в ожидании ресурса, иначе он может быть вытеснен при переходе в режим задачи. Такая реализация ядра позволяет решить множество проблем синхронизации, связанных с доступом нескольких процессов к одним и тем же структурам данных ядра.