

# Презентация №6

Процессы и Нити: Основы Многозадачности

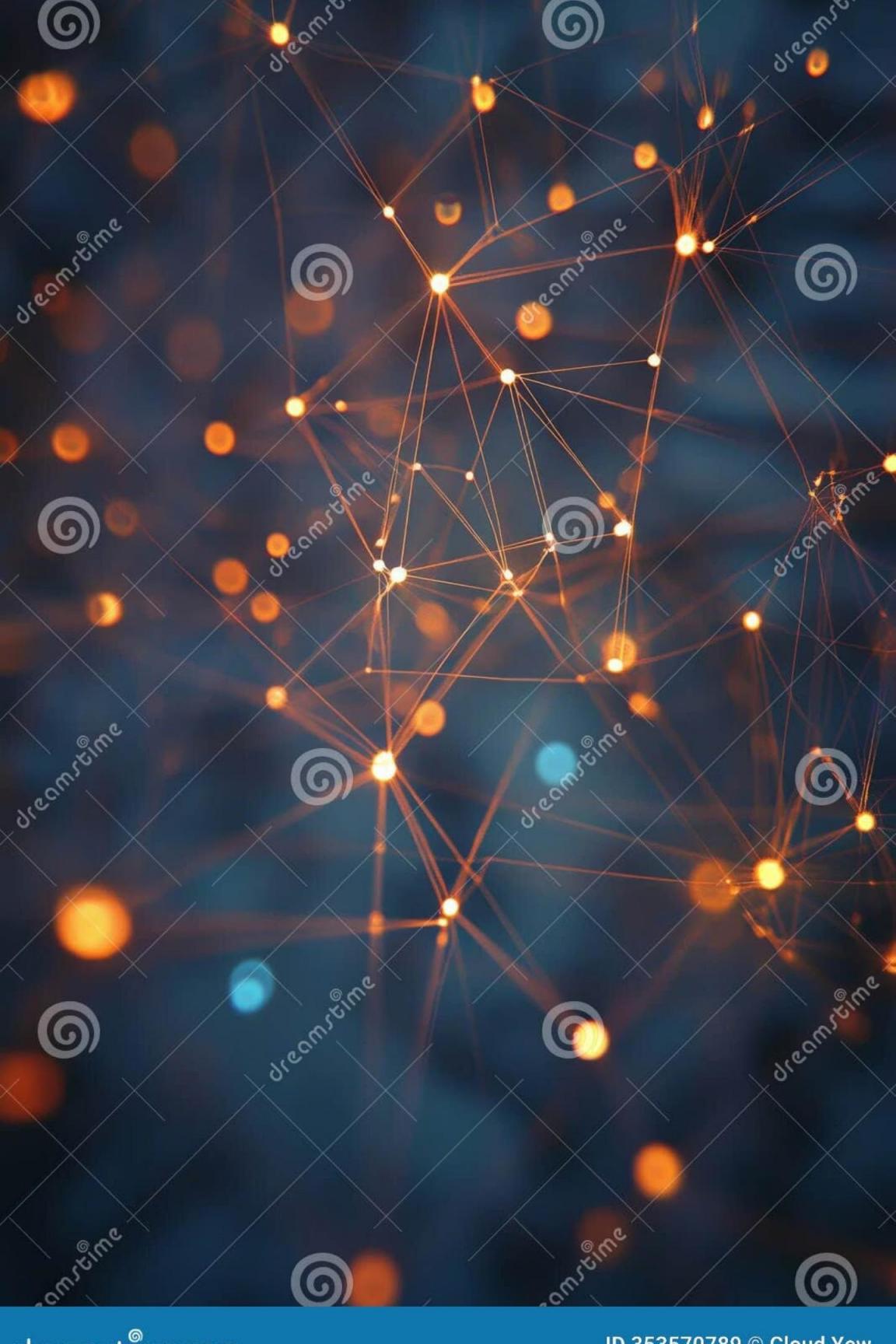
Студент: Эйвази Мани

Группа: НПИбд-03-24

Студенческий билет №: 1032245107

# Процессы и Нити: Основы Многозадачности

*Глубокое погружение в принципы работы операционных систем*



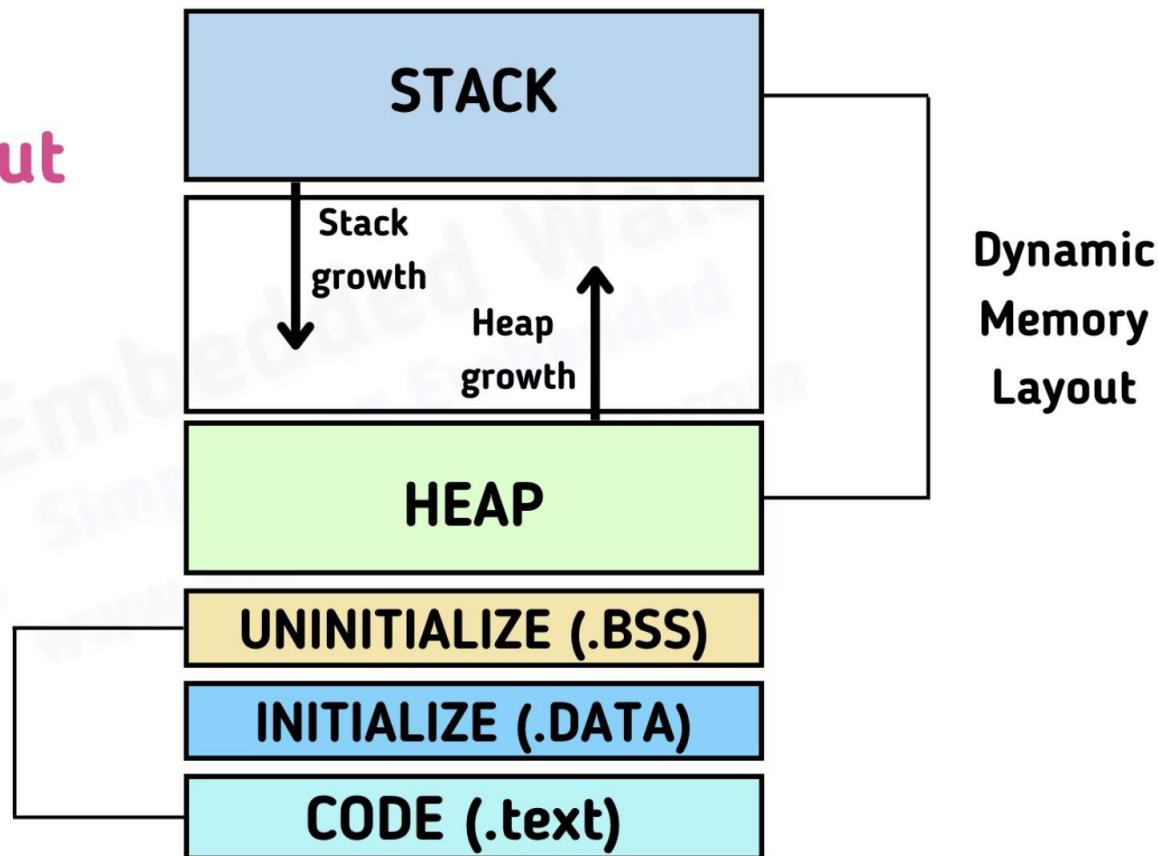
# Процесс: Контейнер Ресурсов

## Memory Layout of C

Static  
Memory  
Layout



Embedded Wala®  
Simplifying Embedded  
[www.embeddedwala.com](http://www.embeddedwala.com)



Процесс представляет собой исполняющуюся программу, действующую как автономная единица, изолированная от других процессов.

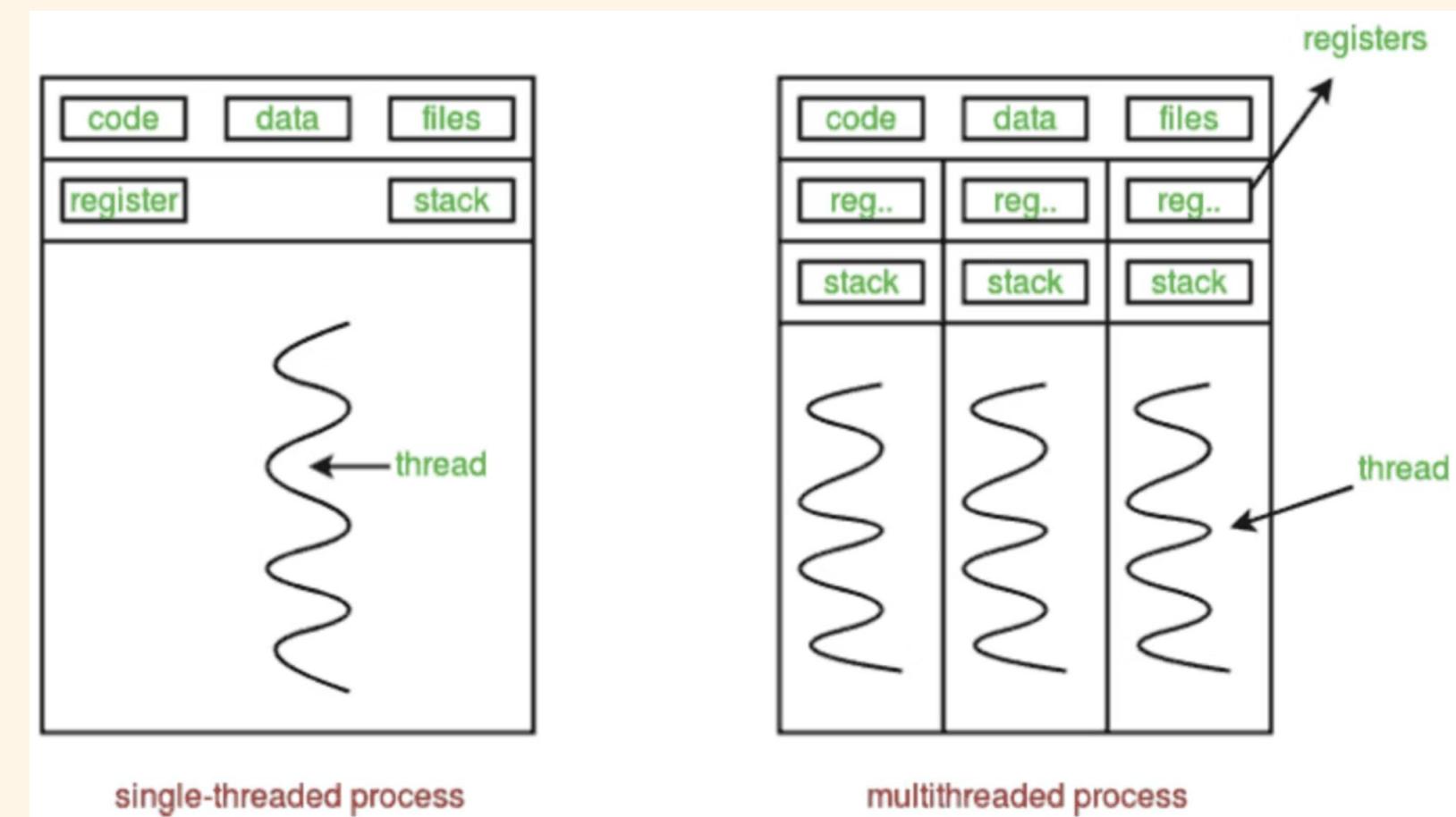
Основные компоненты процесса:

- **Адресное пространство:** Выделенная область памяти, где хранятся код, данные, стек и куча процесса.
- **Дескрипторы ресурсов:** Ссылки на открытые файлы, сетевые соединения, устройства ввода/вывода и другие системные ресурсы.
- **Контекст безопасности:** Определяет права доступа процесса к системным ресурсам.

# Нить (Поток): Единица Выполнения

Нить – это наименьшая последовательность программных инструкций, которая может быть независимо выполнена планировщиком операционной системы. Все нити в рамках одного процесса совместно используют его ресурсы, но каждая нить имеет собственный контекст выполнения.

- **Стек:** Каждая нить имеет свой собственный стек для хранения локальных переменных и информации о вызовах функций.
- **Регистры:** Содержимое регистров центрального процессора, отражающее текущее состояние выполнения нити.
- **Счетчик команд:** Указатель на следующую инструкцию, которая будет выполнена нитью.



# Процессы vs. Нити: Ключевые Отличия



## Изоляция

*Процессы:* Полностью изолированы, имеют раздельное адресное пространство.

*Нити:* Разделяют адресное пространство и ресурсы одного процесса.



## Накладные Расходы

*Процессы:* Высокие на создание и переключение контекста.

*Нити:* Низкие на создание и переключение, т.к. ресурсы уже выделены процессу.

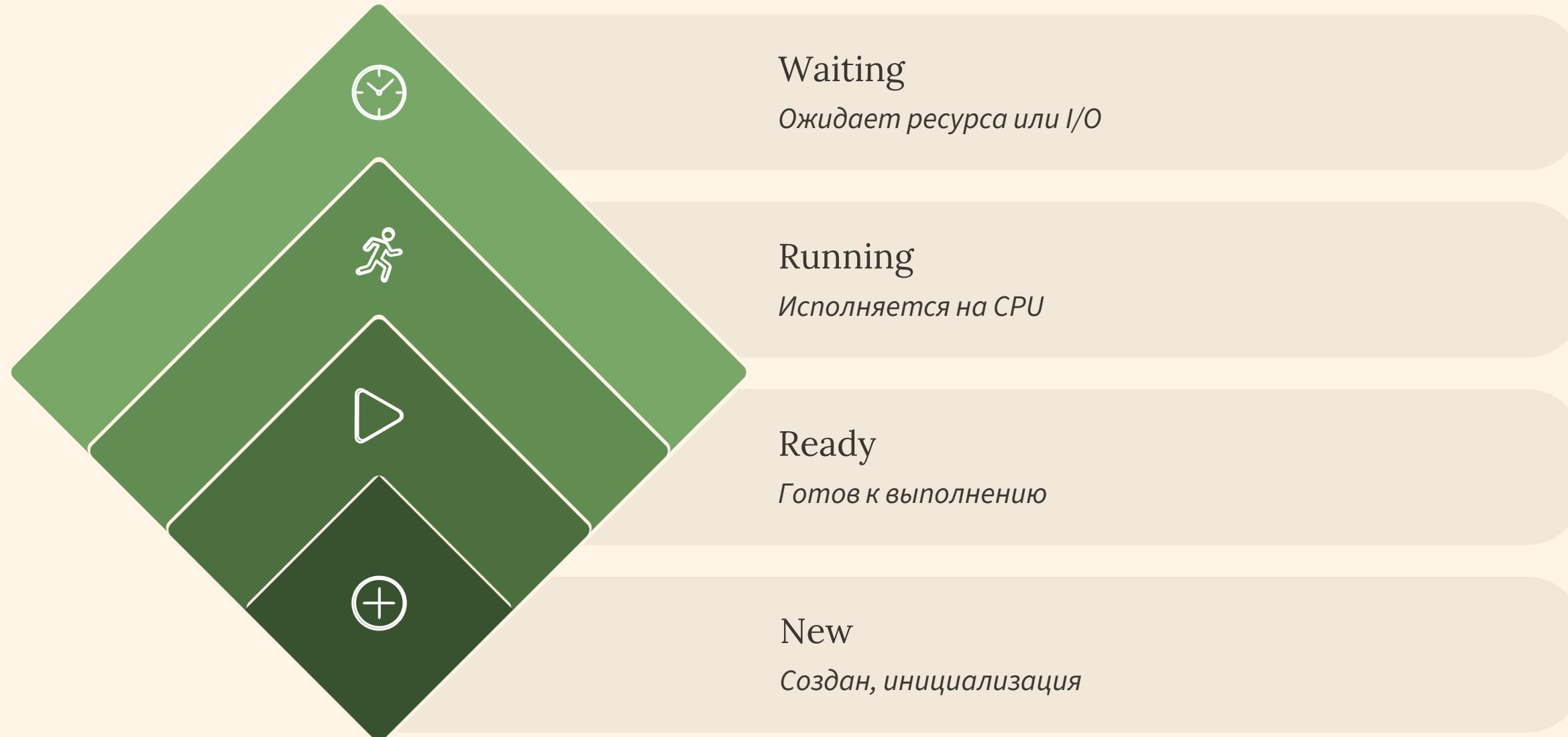


## Взаимодействие

*Процессы:* Сложное межпроцессное взаимодействие (IPC).

*Нити:* Простое взаимодействие через общую память.

# Жизненный Цикл Процесса/Нити

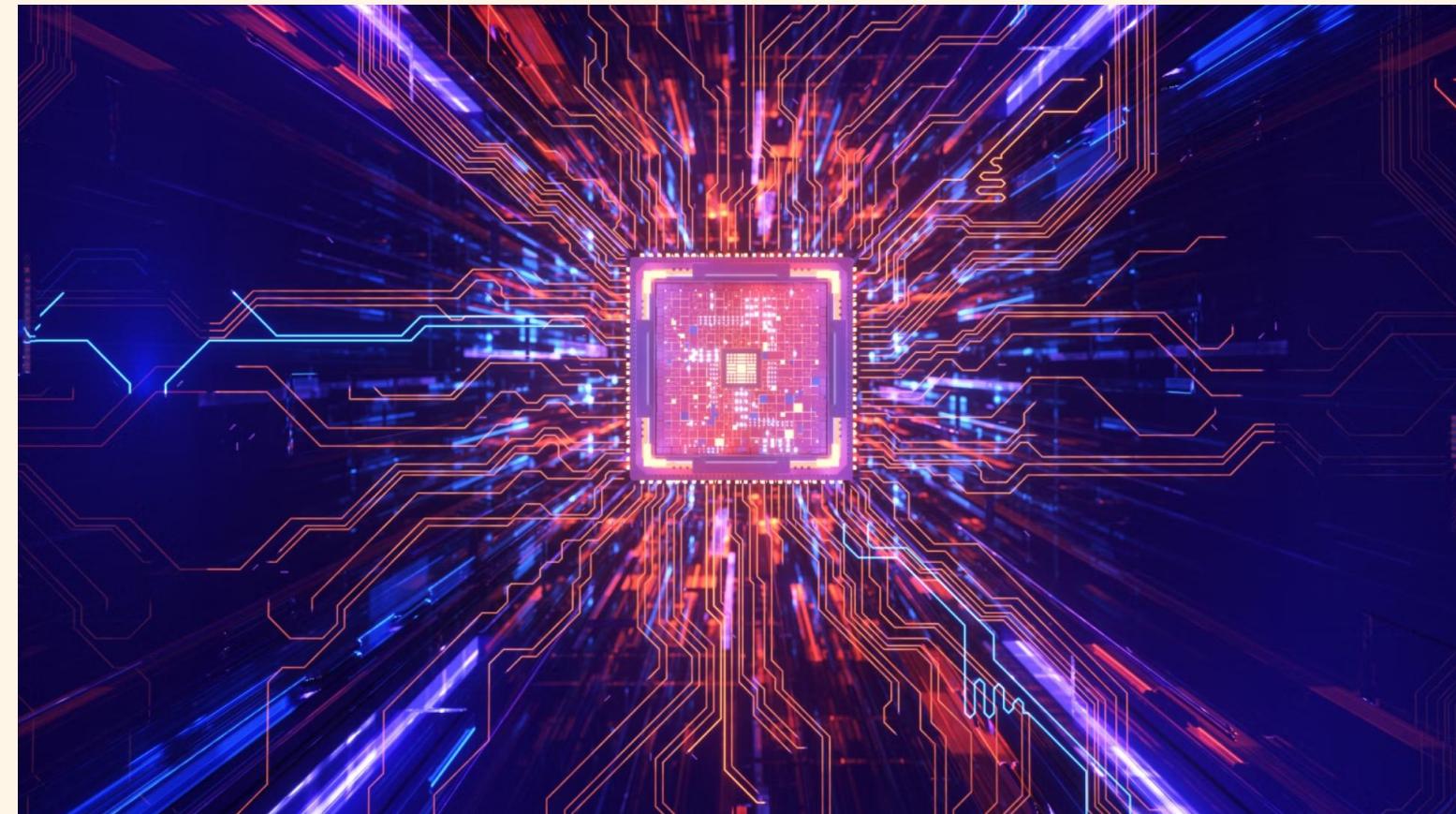


Каждый процесс или нить проходит через определенные состояния от момента создания до завершения. Понимание этих состояний помогает диагностировать проблемы производительности и зависания приложений.

# Планирование Выполнения: Ключ к Эффективности

*Операционные системы используют планировщик для эффективного распределения процессорного времени между процессами и нитями.*

- *Упреждающая многозадачность:* ОС прерывает выполнение процесса/нити и передает управление другому, основываясь на временных квантах или приоритетах.
- *Приоритеты:* Определяют, сколько процессорного времени получает процесс или нить. Более высокие приоритеты означают больший доступ к CPU.
- *Кванты времени:* Максимальный промежуток времени, в течение которого процесс или нить может непрерывно выполняться на CPU.



# Многопоточность: Преимущества и Вызовы



## Преимущества

- Улучшенная отзывчивость приложений.
- Эффективное использование многоядерных процессоров.
- Упрощенное совместное использование данных внутри процесса.



## Проблемы

- **Гонки данных:** Неожиданные результаты из-за несогласованного доступа к общим данным.
- **Взаимоблокировки:** Ситуации, когда нити блокируют друг друга, ожидая ресурсы, занятые другими.
- Увеличенная сложность отладки.

# Механизмы Синхронизации Нитей

Для предотвращения гонок данных и взаимоблокировок используются специальные механизмы синхронизации.



## Мьютексы

*Взаимное исключение: гарантируют, что только одна нить может владеть ресурсом одновременно.*



## Семафоры

*Управляют доступом к ограниченному числу ресурсов, используя счетчик разрешений.*



## Критические Секции

*Обеспечивают эксклюзивный доступ к участкам кода, аналогично мьютексам, но легче и быстрее для использования внутри процесса.*



## События

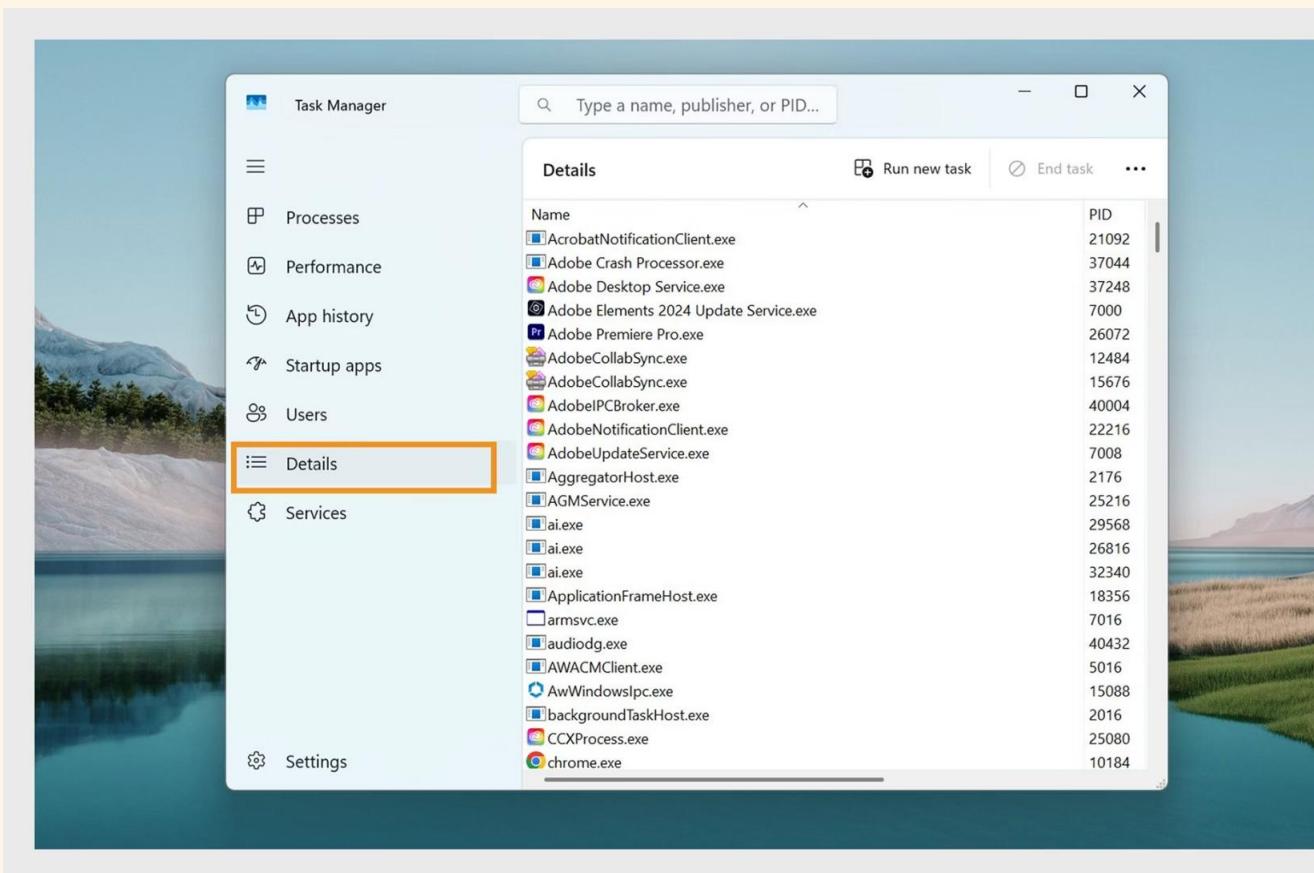
*Механизмы, позволяющие нитям сигнализировать друг другу о наступлении определенного условия.*

# Диагностика и Мониторинг

Администраторы и инженеры используют специализированные инструменты для отслеживания и анализа активности процессов и нитей.

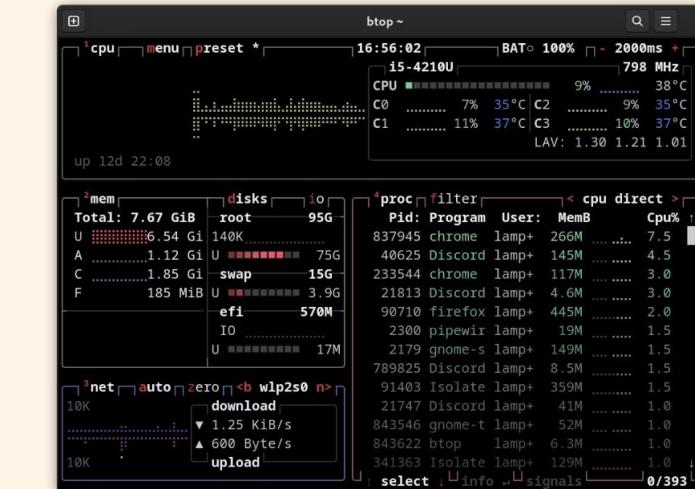
## Windows

- **Диспетчер задач:** Базовый инструмент для просмотра процессов, использования CPU, памяти и диска.
- **Process Explorer:** Расширенный инструмент от Sysinternals для детального анализа процессов и их нитей.
- **Resource Monitor:** Предоставляет графическое представление использования ресурсов.



## Linux

- **ps:** Командная утилита для отображения информации о текущих процессах.
- **top:** Интерактивный монитор процессов, показывающий использование ресурсов в реальном времени.
- **htop:** Улучшенная версия `top` с более удобным интерфейсом и дополнительными функциями.





# Итоги: Фундамент Современных Систем

Понимание процессов и нитей является краеугольным камнем для работы с современными операционными системами и разработки эффективного программного обеспечения. Они обеспечивают многозадачность, отзывчивость и оптимальное использование аппаратных ресурсов.

- **Процессы:** Изолированные среды выполнения, важные для стабильности системы.
- **Нити:** Легковесные единицы выполнения, обеспечивающие параллелизм и высокую производительность.
- **Синхронизация:** Критически важна для предотвращения ошибок в многопоточных приложениях.
- **Мониторинг:** Необходим для диагностики и оптимизации производительности.

Спасибо за внимание