Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

Реферат

По дисциплине «Основы алгоритмизации и программирования»

На тему «Процессы и потоки»

Выполнила:

Студент(ка) 1 курса 7 группы

Подшиваленко Диана Игоревна

Проверил:

Белодед Николай Иванович

2023, Минск

**Содержание**

1. Предисловие … 3
2. Понятие “процесс” … 4
3. Параметры и характеристика процесса … 5
4. Понятие “поток” … 6
5. Параметры и характеристика потока … 7
6. Главные отличия процесса от потока … 8
7. Аналогия представления понятий процесса и потока на примере аквариума … 9
8. Выводы … 10

**Предисловие**

При работе с операционными системами в программировании возникают понятия «процесс» и «поток», которые могут вызывать некоторую путаницу. В данном реферате рассмотрим, что такое процесс и что такое поток, а также основные отличия между ними.

**1.Понятие “процесс”**

**Процесс** — это экземпляр программы, запущенный на компьютере, который выполняется в операционной системе. Каждый процесс имеет свой уникальный идентификатор и набор ресурсов, таких как память, открытые файлы и сетевые сокеты, процессорное время. Процесс имеет собственное пространство адресов, в котором он может загружать и выполнять свой код. Процессы создаются и уничтожаются операционной системой в ответ на различные действия пользователя или других программ.

Процесс в операционной системе – это основная единица выполнения программы. Он представляет собой некоторую последовательность команд, которая выполняется в определенном порядке. Процессы обладают определенными принципами работы, которые определяют их поведение и взаимодействие с другими процессами:

**1.** **Изоляция:** каждый процесс работает в своем собственном виртуальном адресном пространстве, что позволяет изолировать данные и ресурсы от других процессов. Таким образом, процессы не имеют прямого доступа к памяти или файлам других процессов.

**2.** **Планирование:** операционная система отвечает за планирование процессов и определение, какой процесс будет выполняться в данный момент времени. Для этого применяются различные алгоритмы планирования, которые учитывают приоритеты процессов и другие параметры.

**3.** **Синхронизация:** данный принцип позволяет организовать взаимодействие между процессами. Он включает в себя механизмы синхронизации, которые позволяют процессам синхронизировать свою работу и обмениваться данными.

**4.** **Обмен данными:** процессы могут обмениваться данными с помощью механизма меж процессного взаимодействия (IPC). Это позволяет процессам совместно работать над решением задачи, передавать данные друг другу и синхронизировать свою работу.

**5.** **Переключение контекста:** операционная система может выполнять переключение контекста между процессами в любой момент времени. Это означает, что процессы могут быть приостановлены и возобновлены в различных точках программы без потери состояния и данных.

Данные принципы работы процесса позволяют операционной системе эффективно управлять ресурсами и обеспечивать надежное и безопасное выполнение программных приложений.

**2.Параметры и характеристика процесса**

Процесс обладает следующими параметрами и характеристиками, которые определяют его поведение и взаимодействие с операционной системой и другими процессами:

**1.** **Идентификатор процесса (PID):** каждый процесс в операционной системе имеет уникальный идентификатор, который позволяет операционной системе точно определить процесс.

**2.** **Состояние процесса:** каждый процесс может находиться в определенном состоянии: «запущен», «приостановлен», «выполняется» или «завершен». Состояние процесса может изменяться в зависимости от действий операционной системы и других процессов.

**3.** **Приоритет:** обычно операционная система назначает каждому процессу определенный приоритет, который определяет его важность и порядок выполнения. Процессы с более высоким приоритетом выполняются раньше, чем процессы с более низким приоритетом.

**4.** **Родительский процесс:** каждый процесс может иметь родительский процесс, из которого он был создан. Родительский процесс может контролировать дочерний процесс, изменять его параметры и получать информацию о его состоянии.

**5.** **Ресурсы:** каждый процесс может использовать определенные ресурсы системы, такие как процессорное время, память, файловая система и сетевые ресурсы. Ресурсы процесса ограничены и могут быть распределены операционной системой.

**6.** **Средства ввода/вывода:** каждый процесс может использовать различные средства ввода/вывода, такие как клавиатура, мышь, дисплей, принтер и сетевые устройства. Процесс может получать данные от пользователя и отображать результаты своей работы.

Понимание этих параметров позволяет эффективно управлять процессами и ресурсами системы.

**3.****Понятие “поток”**

**Поток** — это последовательность инструкций, выполняемая в рамках процесса. Каждый процесс может содержать несколько потоков, которые работают параллельно или последовательно. Каждый поток имеет свой собственный стек вызовов, указатель инструкции и набор регистров, но использует общую память и ресурсы процесса.

Поток — это логически независимая последовательность инструкций, которые выполняются параллельно с другими потоками. Принципы работы потока определяют способы его создания, выполнения и завершения. Среди них можно выделить следующие:

**1.** **Параллельность:** потоки выполняются параллельно друг другу, что позволяет использовать ресурсы компьютера более эффективно.

**2.** **Безопасность потока:** правильно реализованный поток должен быть безопасным, то есть не должен вызывать гонок данных или других проблем, связанных с одновременным доступом к данным.

**3.** **Распределение ресурсов:** в многозадачной среде потоки могут получать доступ к общим ресурсам, таким как память или файлы, и должны правильно управлять этим доступом, чтобы избежать конфликтов.

**4.** **Синхронизация:** потоки могут синхронизироваться друг с другом, чтобы координировать свои действия и обеспечить правильную работу программы.

**4.****Параметры и характеристика потока**

Потоки имеют следующие параметры и характеристики:

**1.** **Идентификатор потока (Thread ID):** каждый поток в системе имеет свой уникальный идентификатор, который позволяет идентифицировать его в контексте процесса.

**2.** **Состояние потока:** поток может находиться в одном из нескольких состояний: готовность к выполнению «ready», выполнение «running», ожидание «waiting» и завершение «terminated».

**3.** **Приоритет потока:** каждый поток имеет свой приоритет, который определяет относительную важность потока для системы планирования задач. Потоки с более высоким приоритетом будут получать больше ресурсов и могут выполняться чаще.

**4.** **Ресурсы потока:** поток может иметь свои локальные ресурсы, такие как стек вызовов и регистры, а также разделять общие ресурсы процесса с другими потоками. Ресурсы могут быть выделены и освобождены потоком во время его выполнения.

**5.** **Контекст потока:** каждый поток сохраняет свой контекст выполнения, включая значения регистров процессора, указатель на стек и другие регистры. Это позволяет потокам переключаться между собой и продолжать выполнение с того места, где они остановились.

Параметры и характеристики потоков очень важны для планирования работы системы, оптимизации использования ресурсов и управления выполнением программы. Если правильно использовать потоки, то можно создавать эффективные и отзывчивые приложения, способные обрабатывать одновременно несколько задач.

**5.****Главные отличия процесса от потока**

Поток и процесс имеют следующие отличия:

**1.** Процесс является самостоятельным и изолированным, в то время как поток работает внутри процесса и разделяет его ресурсы.

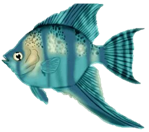
**2.** У каждого процесса есть своё уникальное адресное пространство, а каждый поток разделяет общую область памяти.

**3.** Создание и завершение процессов требуют больше времени и ресурсов в сравнении с потоками.

**4.** Потоки могут быстро обмениваться данными, так как они разделяют общую область памяти, в то время как процессы требуют использования механизмов меж процессного взаимодействия (IPC).

**5.** Потоки являются более быстрыми по сравнению с процессами, так как не требуют выделения дополнительных ресурсов.

**6.Аналогия представления понятий процесса и потока на примере аквариума**



В данном случае мы можем представить, что наша память – это вода, водоросли и коряги – это ресурсы, рыбки – это потоки. Тогда аквариум с его содержимым – это и есть процесс. В процессе должен быть хотя бы 1 поток (точно также как и в аквариуме должна быть хотя бы 1 рыбка).

**Выводы**

Процессы и потоки связаны друг с другом, но при этом имеют существенные различия.

Процессы и потоки предоставляют мощные средства для реализации параллельного выполнения и управления ресурсами. Их применение охватывает широкий спектр областей, от операционных систем до вычислений на GPU. Понимание различий и особенностей процессов и потоков может помочь разработчикам создавать более эффективные приложения.

Главные отличия процесса и потока:

**1.** Процесс является самостоятельным и изолированным, в то время как поток работает внутри процесса и разделяет его ресурсы.

**2.** У каждого процесса есть своё уникальное адресное пространство, а каждый поток разделяет общую область памяти.

**3.** Создание и завершение процессов требуют больше времени и ресурсов в сравнении с потоками.

**4.** Потоки могут быстро обмениваться данными, так как они разделяют общую область памяти, в то время как процессы требуют использования механизмов меж процессного взаимодействия (IPC).

**5.** Потоки являются более быстрыми по сравнению с процессами, так как не требуют выделения дополнительных ресурсов.