**Лекция № 8. Процессы и потоки в Windows**

**Цель занятия:** Сформировать знания студентов о процессах и потоках в операционной системе Windows, а так же об управлении памятью в ОС Windows.

Процесс является динамическим объектом, описывающим выполнение программы. Процессу выделяются системные ресурсы: закрытое адресное пространство, семафоры, коммуникационные порты, файлы и т.д. Процесс характеризуется текущим состоянием (выполнение, ожидание, готовность и т.д.).

Задача ОС состоит в том, чтобы организовать их поддержку, которая подразумевает, что каждый процесс получит все необходимые ему ресурсы.

Независимые процессы не должны влиять друг на друга, а процессы, которым необходимо обмениваться информацией, должны иметь возможность сделать это путем межпроцессного взаимодействия.

В Windows процесс – это объект, создаваемый и уничтожаемый менеджером объектов. Процесс содержит потоки, учетную информацию и ссылки на ресурсы, которые обобществляются всеми потоками процесса.

Процесс в Windows состоит из следующих компонентов:

- структура данных, содержащая всю информацию о процессе, в том числе список открытых дескрипторов различных системных ресурсов, уникальный идентификатор процесса, различную статистическую информацию и т.д.;

- адресное пространство - диапазон адресов виртуальной памяти, которым может пользоваться процесс;

- исполняемая программа и данные, проецируемые на виртуальное адресное пространство процесса.

Поток — некая сущность внутри процесса, получающая процессорное время для выполнения. Без потока программа процесса не может выполняться. Поток включает следующие наиболее важные элементы:

- содержимое набора регистров процессора, отражающих состояние процессора;

- два стека, один из которых используется потоком при выполнении в режиме ядра, а другой — в пользовательском режиме;

- закрытую область памяти, называемую локальной памятью потока и используемую подсистемами, библиотеками исполняющих систем и DLL;

- уникальный идентификатор потока;

- иногда потоки обладают своим контекстом защиты, который обычно используется многопоточными серверными приложениями, подменяющими контекст защиты обслуживаемых клиентов.

Выбор текущего потока из нескольких активных потоков, пытающихся получить доступ к процессору, называется планированием.

Выбранный для выполнения поток работает в течение некоего периода, называемого квантом, по истечении которого поток вытесняется. Предполагается, что поток не знает, в какой момент он будет вытеснен. Поток также может быть вытеснен даже, если его квант еще не истек.

Процедура планирования обычно связана с весьма затратной процедурой диспетчеризации - переключением процессора на новый поток, поэтому планировщик должен заботиться об эффективном использовании процессора. Принадлежность потоков к процессу при планировании не учитывается, то есть единицей планирования в ОС Windows является именно поток.

В ОС Windows запуск процедуры планирования вызывается одним из следующих событий. *События, связанные с освобождением процессора*:

1. Завершение потока
2. Переход потока в состояние готовности в связи с тем, что его квант времени истек
3. Переход потока в состояние ожидания.
4. События, в результате которых пополняется или может пополниться очередь потоков в состоянии готовности.
5. Поток вышел из состояния ожидания
6. Поток только что создан
7. Деятельность текущего потока может иметь следствием вывод другого потока из состояния ожидания

*Создание процессов*

Создание Win32 процесса осуществляется вызовом одной из таких функций, как CreateProcess, CreateProcessAsUser и CreateProcessWithLogonW и происходит в несколько этапов:

- Открывается файл образа (EXE), который будет выполняться в процессе. Если исполняемый файл не является Win32 приложением, то ищется образ поддержки для запуска этой программы. Например, если исполняется файл с расширением .bat, запускается cmd.exe и т.п.

- Создается объект Win32 «процесс».

- Создается первичная нить (стек, контекст и объект «нить»).

- Подсистема Win32 уведомляется о создании нового процесса и нити.

- Начинается выполнение первичной нити.

- В контексте нового процесса и потока инициализируется адресное пространство и начинается выполнение программы.

*Завершение процессов*

Процесс завершается если:

- Входная функция первичной нити возвратила управление.

- Одна из нитей процесса вызвала функцию ExitProcess.

- Нить другого процесса вызвала функцию TerminateProcess.

Когда процесс завершается, все User- и GDI\_объекты, созданные процессом, уничтожаются, объекты ядра закрываются (если их не использует другой процесс), адресное пространство процесса уничтожается

*Менеджер памяти* - часть операционной системы, отвечающая за управление памятью.

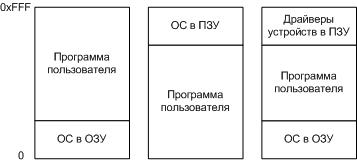
Основные методы распределения памяти:

1. Методы без использования внешней памяти

Однозадачная система без подкачки на диск

Память разделяется только между программой и операционной системой.

Схемы разделения памяти:



Третий вариант используется в MS-DOS. Та часть, которая находится в ПЗУ, часто называется BIOS.

2 Распределение памяти с фиксированными разделами.

Память просто разделяется на несколько разделов (возможно, не равных). Процессы могут быть разными, поэтому каждому разделу необходим разный размер памяти.

Системы могут иметь:

* общую очередь ко всем разделам
* к каждому разделу отдельную очередь Недостаток системы многих очередей очевиден, когда большой раздел может быть свободным, а к маленькому выстроилась очередь.

**Контрольные вопросы:**

1. *Что такое процессы и потоки в Windows?*
2. *Как происходит управление памятью в Windows?*
3. *События, связанные с освобождением процессора*
4. *Как происходит создание процессов?*
5. *Как происходит завершение процессов?*
6. *Что такое менеджер памяти*