Л1. Основные составляющие компьютера

**1. Процессор**-это мозг компьютера. Но в отличие от человеческого мозга, у него нет памяти. Процессор занимается обработкой и управлением данных.

Процессор состоит чаще всего состоит из 2х основных частей:

1. **ALU-**арифметическое-логическое устройство. Выполняет арифметические функции (сложение, умножение чисел/битов и т.д.)
2. **Control Unit-**управляющий автомат

Процессор имеет внутреннюю память(регистры).   
**Регистры.** Работа с регистрами (ячейками памяти с адресами на процессоре) намного быстрее чем работа с основной памятью. Регистров намного меньше чем ячеек в основной памяти. Некоторые регистры доступны пользователям. Некоторые регистры используются для управления и статусов (чтобы процессор мог ответить, как прошла предыдущая операция).   
**Регистр флагов**. Регистр, в котором хранится бит и это способ процессора сказать нам, что какая-то операция завершилась. Используется для хранения статуса выполнения. Только для чтения.  
**Регистр счетчик команд.** Содержит адрес команды (текущей или следующей)

**2. Память** **(RAM, ОЗУ)**- хранит данные. Непостоянная, т.е. при отключении питания, все данные удаляются. Память представляет собой набор ячеек с адресами. Ячейки чаще всего содержат данные или инструкции (данные, которые можно исполнить, набор команд)

**3. I/O модули**(**input/output)** - модули для устройства ввода/вывода. Занимается перемещением данных между компьютером и различными устройствами, например жестким диском, сетевой картой, терминалом, мышкой.

**4. Системная шина (System Bus)-**объединяет процессор, память и I/O модули. Благодаря ей они могут обмениваться информацией.

**5. Программа-**это набор инструкций в памяти. Процессор считывает инструкции в регистр инструкции и исполняет их.  
**Инструкция-**простейшая операция, которую процессор может проводить над различными компонентами:

* Процессор-память
* Процессор – модули I/O
* Управление данными
* Обработка данных

Программа разбивается на простейшие инструкции, которые выполняет процессор.

**6. Прерывание-**прерывание нормального порядка исполнения инструкций. Необходимо для улучшения использования ЦП (центрального процессора) и реализации многозадачности.

Л2. Эволюция ОС. Фундаментальные проблемы. Слои абстракции

Основные слои абстракции:

1. Пользовательские приложения -> Прекрасный интерфейс -> Операционная система -> Ужасный интерфейс -> Железо

Л3. Представление и обработка процессов. Структуры данных. Очереди.

ОС должна:

* Управлять запуском нескольких процессов
* Предоставлять необходимые ресурсы процессу и защищать ресурсы процесса от остальных
* Позволять ресурсам обмениваться информацией
* Позволять процессам синхронизироваться

Процесс состоит из:

* Идентификатор (уникальное имя, по которому можно обратится к этому процессу)
* Состояние процесса (на каком этапе сейчас процесс)
* Приоритет
* Счетчик команд-Program counter (показывает на текущий процесс и на следующий процесс)
* Указатели на память
* Контекст (окружение процесса)
* Информация о статусе ввода/вывода
* Другая информация

**Контрольный блок процесса (Process Control Block)-**структура данных, используемая ОС для представления процесса.

Создание процесса:

1. ОС создает структуру данных для процесса
2. Процессы обычно создаются самой ОС, но часто процессы могут создавать новые процессы (process spawning), т.е. родительский процесс-> дочерний процесс (child process)

Уничтожение процесса:

Необходимо чтобы было понятно, когда процесс завершен:

1.1 Сигнал HALT

1.2 Действие пользователя

1.3 Ошибка

1.4 Завершение родительского процесса

**Множественная очередь**-очередь, готовых к запуску процессов и различные очереди, которые ждут реализации различных событий (ответа пользователя или жесткого диска).

**Приостановленные процессы (suspended process)**. Возможно такое, что все процессы ждут ввода/вывода. Их можно перенести на диск, чтобы освободить память для более актуальных процессов.

ОС для управления процессами использует различные таблицы. **Таблица памяти (**отвечает за основную и вторичную память**):**

* Предоставление основной памяти процессам
* Предоставление вторичной памяти процессам
* Атрибуты зашиты для доступа к общей памяти
* Информация, необходимая для работы виртуальной памяти

**Таблица ввода/вывода (**информация об устройствах ввода/вывода)**.** Системе необходимо знать:

* Доступно ли устройство
* Статус операции ввода/вывода
* Адрес в памяти, который используется для ввода или вывода

**Файловая таблица.** Информация о:

* Существовании файлов
* Расположение во вторичной памяти
* Текущий статус

**Таблица процессов.** Для эффективного управления системе необходимо знать:

* Идентификатор процесса
* Текущее состояние процесса
* Расположение в памяти
* И т.д.

С точки зрения процессов существует 2 режимы, в которых работает компьютер:

1. Пользовательский режим (в этом режиме процесс не может навредить, т.е. он происходит под надзором какой-то сущности операционной системы. У процесса нет абсолютного доступа к железу, т.е. он проходит через какой-то барьер, любое его действие может быть отклонено)
2. Системный режим (у процесса есть полный доступ ко всему, что есть в компьютере, на этом режиме работают самые важные процессы ОС)

Когда создается процесс, ему присевается уникальный идентификатор. Любому процессу нужна память (простое предоставление памяти, или если памяти не хватает, то ОС запустит работу по освобождению памяти, например, spawning). Далее происходит инициализация контрольного блока (структура данных, которая предоставляет процесс в памяти). Далее происходит создание или изменение существующих структур (если процесс является дочерним).

Переключение процесса:

1. Сохранить состояние процесса (регистры)
2. Обновить контрольный блок процесса, который запущен в данный момент
3. Перенести контрольный блок процесса в соответствующую очередь –ready, blocked: ready/suspended
4. Выбрать другой процесс для выполнения
5. Обновить контрольный блок процесса, на который необходимо переключиться
6. Обновить структуры данных управления памятью
7. Восстановить состояние регистров процесса

Ядро (тоже процесс):

* Запущено вне процессов
* Находится в отдельной области памяти
* Функционирует в привилегированном режиме
* Понятие процесса относится только к пользовательским программам

Windows поддерживает самые различные базовые процессоры и архитектуры систем, для чего предусмотрен **уровень аппаратных абстракций** (**Hardware Abstraction Layer, HAL**), который обеспечивает переносимость программ на системы с другой архитектурой процессора.

**Принципы, лежащие в основе Windows**

Многие системные ресурсы Windows представляются в виде объектов ядра (kernel objects), для идентификации и обращения к которым используются дескрипторы (handles). По смыслу эти дескрипторы аналогичны дескрипторам (descriptors) файлов и идентификаторам (ID) процессов в UNIX