**Операционные системы**

**Лекция 1**

**Отчётность:**

1) Конспекты (к зачёту)

2) По каждой части материала лекционного осуществить обработку (формулировка вопросов 10-15) / словарь

3) Найти код каждой лекции, адаптировать его под ту систему программированию, которую будете использовать, составить блок-схему

4) Копия экрана запуска программы

5) Практику посвятим основы работы в командной строки ОС Linux (40-50 заданий) / для вывода списка перечня команд набрать help

Практикум по командной строке реализуется в VM, а отчёт в редакторе текста.

Операционная система — это комплекс взаимосвязанных программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем.

Операционная система — это комплекс программ, которые обеспечивают возможность рационального использования оборудования компьютера наиболее комфортным для пользователя образом

В логической структуре типовой вычислительной системы ОС занимает положение между устройствами с их микро архитектурой, машинным языком и возможностями реализации микропрограмм (драйверов) с одной стороны, а с другой потребностями и функциями прикладных программ

Разработчикам ПО операционная система позволяет абстригироваться от деталей реализации и функционирования технических устройств, предоставляя пользователям необходимый набор функций, например, интерфейсы программирования приложений

В большинстве вычислительных систем ОС является основной, наиболее важной (а взачастую единственной) частью системного ПО.

С 90-х годов 20-го века наиболее распространенными ОС являются представители семейств Windows, Unix, Linux.

**Структура вычислительной системы и место ОС в ней:**

В англо язычных странах под вычислитеной системой чаще всего понимают оборудование (hardware) или технической обеспечение. По структуре в вычислительную систему включают процессор, память, монитор, дисковое устройство, объединённые единой шиной или магистралью, которая обеспечивает взаимодействие. С другой стороны под ВС понимают систему, состоящую из элементов прикладного и системного программного обеспечения. К прикладному ПО относят практические приложения, которые позволяют реализовать те, или иные виды профессиональной деятельности (жкономические процессы, коммуникативные, социальные, образовательные и т. д.). К системному ПО относят программы, которые используются для разработки или оптимизации работы прикладных программ. Развитие ПО за последние 30-40 лет привело к тому, что постепенно стирается грань между системным и прикладным ПО, поэтому в настоящее время данное деление является условным и зависит лишь от того пользователя, который её осуществляет. Тем не менее, несмотря на нечёткие грани структуру ПО и место ОС можно представить в виде следующей схемы:

Пользователь

Прикладные программы - ПО

Системные программы, кроме ОС (утилиты) — ПО / системное ПО

Операционная система - ПО / системное ПО

Техническое обеспечение

**История создания и развития вычислительных систем**

Предшественником ОС являются служебные программы ***загрузчики и мониторы,*** а также библиотеки, часто используемых подпрограмм, которые начали разрабатываться с появлением универсальных компьютеров первого поколения в конце 1940-х годов. Служебные программы минимизировали в физические манипуляции оператора с оборудованием, а библиотеки позволяли избегать многократного программирования одних и тех же действий (операции ввода-вывода данных и вычисление математических функций).

В 1950-60-х годах сформировались и были реализованы основные идеи, определяющие современную функциональность ОС:

1) Пакетный режим работы

2) Разделение времени

3) Многозадачность

4) Разделение полномочий

5) Работа в реальном масштабе времени

6) Файловая структура и файловые системы

**Пакетный режим.** Необходимость оптимального использования дорогостоящих вычислитеных ресурсов привела разработчиков к необходимости концепции пакетного режима исполнения программ, который предполагает наличие ***очереди программ на исполнение,*** при которой ОС может обеспечивать загрузку программы с внешних носителей данных в оперативную память, ***не дожидаясь завершения исполнения предыдущей программы,*** что позволило избегать простоя процессора.

**Разделение времени и многозадачность.** Реализация пакетного режима в своих поздних реализациях требовала разделения процессорного времени между выполнением нескольких программ.

Необходимость в разделении времени, многозадачности, мулитипрограммировании проявилась ещё острее при внедрении относительно новых на то время устройств ввода-вывода так называемых телетайпов, которые впоследствие были заменены мониторами на электронно-лучевых трубках. Так как скорость ввода данных с терминала (клавиатуры) намного ниже, чем скорость обработки данных процессоров, то использование компьютеров в монопольном режиме (одним оператором) приводило к простою дорогостоящего оборудования.

Разделение времени позволило создать многопользовательские системы, в которых один ЦП и блок оперативной памяти соединялся с множеством теорминалов. При этом часть задач исполнялось в режиме диалога (например, ввод, редактирование данных оператора), а часть задач реализовывалось в пакетном режиме (например, задачи, связанные с обработкой больших массивов информации).

**Разделение полномочий.** Распространение многопользовательских систем потребовало решения задач разделения полномочий, позволяющих избегать возможности изменения исполняемой программы или данных одной программы в памяти другой программой или пользователем (намеренно или по ошибке), а также изменение самой системы со стороны прикладных программ.

Реализация разделения полномочий в ОС была поддержана прежде всего разработчиками процессоров, которые на тот момент создали архитектуры с двумя режимами работы процессоров: реальный режим (режим, в котором исполняемой программе доступно всё адресное пространство компьютера) и защищённый режим (режим, в котором доступность адресного пространство ограничено диапазоном, выделенным данным приложением)

**Масштаб реального времени.** Внедрение универсальных компьютеров для управления производственными процессорами потребовало разработки принципа «Масштаба реального времени» - это процессы синхронизации исполнения программ с внешними физическими процессами. Включение функции масштаба реального времени позволило создавать решение ***одновременно обслуживающие производственные процессы и решающее другие системные задачи*** в различных режимах.

**Файловые системы и структуры.** Разработка файловых систем и структур была связана с постепенной заменой носителей последовательного доступа к данным (перфокарты, перфоленты, магнитные карты), они были заменены накопителями произвольного доступа. Файловая система — это способ хранения данных на внещних запоминающих устройствах.

Основные функции:

1) Исполнение запросов программ (ввод-вывод данных), запуск и остановка других программ, выделение / освобождение дополнительной памяти и пр.

2) Загрузка программ в оперативную память и их выполнение

3) Стандартизованный доступ к периферийным устройствам (устройства ввода-вывода)

4) Управление оперативной памятью (разделение памяти между процессами, организация виртуальной памяти)

5) Управление доступом к данным на энерго независимых носителях (например, жёсткие диски, оптические диски, флэш-память), организованных в той или иной файловой системе

6) Обеспечение пользовательского интерфейса к файлам и операциями над ними

7) Сохранение или мониторинг информации об ошибках работы системы

Кроме основных существуют и дополнительные, к части которых относятся:

1) Параллельное и псевдопараллельное выполнение задач (многозадачность)

2) Эффективное разделение ресурсов вычислительных систем между процессами

3) Разграничение доступа различных процессов к ресурсам

4) Организация надёжных вычислений (невозможности одного вычислительного процессора намеренно или по ошибке влять на вычисление в другом процессоре). Данная функция основана на разграничении доступа к ресурсам.