

# СИСТЕМНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ



# Системное программное обеспечение -

это комплекс программ, которые обеспечивают эффективное управление компонентами вычислительной системы, такими как процессор, оперативная память, каналы ввода-вывода, сетевое оборудование, выступая как «межслойный интерфейс» с одной стороны которого аппаратура, а с другой приложения пользователя. В отличие от прикладного программного обеспечения, системное не решает конкретные прикладные задачи, а лишь обеспечивает работу других программ и управляет аппаратными ресурсами вычислительной системы.

# Предназначение системных программ.

1. Организация выполнения прикладных программ.
2. Обеспечение надежной и эффективной работы самого компьютера и вычислительной сети.
3. Автоматизация разработки и создания новых программ.
4. Выполнение вспомогательных технологических процессов.
5. Проведение диагностики аппаратуры компьютера и вычислительных сетей.

# Системное ПО

```
graph TD; A[Системное ПО] --> B(Операционные системы); A --> C(Системы управления файлами); A --> D(Утилиты (служебные программы)); A --> E(Системы программирования); A --> F(Интерфейсные оболочки);
```

Операционные  
системы

Системы  
управления  
файлами

Утилиты  
(служебные  
программы)

Системы  
программирования

Интерфейсные  
оболочки

# Операционная система -

совокупность программных средств, обеспечивающая управление аппаратной частью компьютера и прикладными программами, а также их взаимодействием между собой и пользователем. Операционная система, с одной стороны, выступает как интерфейс между аппаратурой компьютера и пользователем с его задачами, с другой стороны, предназначена для эффективного использования ресурсов вычислительной системы и организации вычислений.

# Функции операционной системы.

1. Прием от пользователя заданий и команд, сформулированных на соответствующем языке.
2. Загрузка в оперативную память подлежащих исполнению программ.
3. Обеспечение работы СУФ и СУБД.
4. Организация и управление всеми операциями ввода/вывода.
5. Идентификация всех программ и данных.
6. Обеспечение режима мультипрограммирования, т.е. выполнения двух или более программ на одном процессоре.
7. Распределение и организация виртуальной памяти.

# Основные операционные системы.

- **Mac OS** – операционная система корпорации Apple.
- **OS/2** — операционная система фирмы IBM.
- **Windows** - операционная система корпорации Microsoft.
- **Linux** - общее название Unix-подобных операционных систем на основе одноимённого ядра и собранных для него библиотек и системных программ, разработанных в рамках проекта GNU.

# Системы управления файлами

предназначены для более удобного доступа к данным, организованным в виде файла. Вместо низкоуровневого доступа к данным с указанием конкретных физических адресов система управления файлами позволяет использовать логический доступ с указанием имени файла. Для работы с файлами, организованными в некоторую файловую систему, для каждой ОС должна быть разработана соответствующая система управления файлами.



# Основные файловые системы.

- **FAT** - наиболее простая из файловых систем. Диск, отформатированный в файловой системе FAT, делится на кластеры, размер которых зависит от размера тома. Основой системы FAT является таблица размещения файлов, которая помещена в самом начале тома.
- **HPFS** - в этой файловой системе поддерживается структура каталогов FAT и добавлена сортировка файлов по именам. Файл состоит из «данных» и специальных атрибутов, помимо их так же присутствуют сведения о создании и внесении изменений, а также дата и время доступа.

## УТИЛИТЫ -

программы, служащие для выполнения вспомогательных операций обработки данных или обслуживания компьютеров и расширяющие стандартные возможности оборудования и операционных систем, выполняющие узкий круг специфических задач. Утилиты предоставляют доступ к возможностям (параметрам и установкам), недоступным без их применения, либо делают процесс изменения некоторых параметров проще, т. е. автоматизируют его.

# Функции утилит.

1. Мониторинг показателей датчиков и производительности оборудования, мониторинг температур процессора.
2. Управление параметрами оборудования.
3. Тонкая настройка параметров системы.
4. Расширение возможностей компьютера.
5. Контроль показателей — проверка ссылочной целостности; правильности записи данных.

## Типы утилит.

- Дисковые утилиты: дефрагментаторы, программы проверки диска (**Scandisk**), очистки диска (**CCleaner**), разметки диска (**fdisk**).
- Менеджеры процессов (**AnVir Task Manager**).
- Утилиты работы с реестром (**Reg Organizer**).
- Утилиты мониторинга оборудования (**SpeedFan**).
- Средства обеспечения компьютерной безопасности (**резервное копирование, антивирусное ПО**).
- Архиваторы (**WinRar**).
- Программы-драйверы.

# Системы программирования -

программное обеспечение компьютера, предназначенное для разработки, отладки и исполнения программ, записанных на определенном языке программирования. Системами программирования, ориентированные на создание Windows-приложений, являются: **Borland Delphi, Microsoft Visual Basic, Borland C++ .**

# Компоненты систем программирования.

1. Транслятор (преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд). Реализуются в виде компиляторов или интерпретаторов.
2. Библиотека подпрограмм и функций.
3. Интегрированная среда разработки.
4. Отладочные программы (помогают находить и устранять ошибки в программе).
5. «Дружественная» к пользователю диалоговая среда.

# Интерфейсные оболочки -

используются для удобства взаимодействия пользователя с операционной системой. Их основное назначение — либо расширить возможности по управлению ОС, либо изменить встроенные в систему возможности.

# Требования к системному ПО.

- Прозрачность работы.
- Гарантированная надежность выполнения.
- Максимальная скорость выполнения.
- Минимальные затраты на хранение машинных кодов.
- Поддержка средств связи с прикладными программами.

Эффективность системных программ зависит от надежности исполняемого кода. Требование эффективности системных программ вызывает необходимость использования специальных языков: машинно-ориентированных типа языка **Assembler** и высокого уровня типа **C** или **C++**.



**СПАСИБО ЗА ВНИМАНИЕ**

