

Тема 4.4 Общие сведения о программном обеспечении станции

Основные понятия и определения

Основной компонент ПО – программа – упорядоченная в соответствии с некоторым алгоритмом последовательность команд (инструкций) компьютера для решения задачи пользователя.

Чаще всего образ программы хранится в виде исполняемого модуля (отдельного файла или группы файлов).

Основные понятия и определения

Пользователь – лицо, заинтересованное в решении некоторой задачи средствами вычислительной техники.

По отношению к программному обеспечению компьютерные пользователи делятся на следующие группы:

- системные программисты, занимающиеся разработкой, эксплуатацией и сопровождением системного программного обеспечения;
- прикладные программисты. Выполняют разработку и отладку программ решения задач из различных прикладных сфер деятельности пользователей;
- конечные пользователи. Используют прикладное программное обеспечение для решения задач в своей повседневной деятельности. Различаются по уровню своей подготовки в части знания и использования компьютерной техники;
- администраторы. Как правило, это высококвалифицированные компьютерные специалисты, отвечающие за работу вычислительной сети, баз данных, корпоративной информационной системы в целом, безопасность и защиту данных. Могут иметь определенную специализацию: управление сетевым каталогом, политикой учетных записей, политикой аудита и т.п.

Основные понятия и определения

Задача (*problem, task*) – проблема, подлежащая решению в интересах пользователя.

Термин "задача" в программировании означает единицу работы вычислительной системы, требующую выделения вычислительных ресурсов (процессорного времени, оперативной и внешней памяти, файлов и т.п.).

Приложение (*application*) – программная реализация решения задачи на компьютере. Приложение может состоять из одной или нескольких взаимосвязанных и взаимодействующих программ.

Простые программы

-решает одну четко поставленную задачу в хорошо известных ограничениях, к тому же, не очень существенную для какой-либо практической или исследовательской деятельности;

-неважно, насколько быстро она работает;

-ущерб от неправильной работы программы – практически нулевой (за исключением возможности обрушения ею системы, в которой выполняются и другие, более важные задачи);

-не требуется дополнять программу новыми возможностями, практически никому не нужно разрабатывать ее новые версии или исправлять найденные ошибки;

Т.о не очень нужно прилагать к программе подробную и понятную документацию – для человека, который ею заинтересуется, не составит большого труда понять, как ею пользоваться, просто по исходному коду.

Сложные программы

Программные системы, программные комплексы, программные продукты, отличаются от небольших не столько по размерам сколько наличием дополнительных факторов. Эти факторы связаны с их востребованностью и готовностью пользователей платить деньги, как за приобретение самой программы, так и за ее сопровождение и даже за специальное обучение работе с ней.

Свойства сложных программ

- программа решает одну или несколько связанных задач, зачастую сначала не имеющих четкой постановки, настолько важных для каких-либо лиц или организаций, что те приобретают значимые выгоды от ее использования;
- программа должна быть удобной в использовании. Она должна включать достаточно полную и понятную пользователям документацию, возможно, также специальную документацию для администраторов, а также набор документов для обучения работе с программой;
- низкая производительность программы на реальных данных приводит к значимым потерям для пользователей;
- неправильная работа программы наносит ощутимый ущерб пользователям и другим организациям и лицам, даже если сбои происходят не слишком часто;
- для выполнения своих задач программа должна взаимодействовать с другими программами и программно-аппаратными системами и обеспечивать работу на разных платформах;
- пользователи, работающие с программой, приобретают дополнительные выгоды от того, что программа развивается, в нее вносятся новые функции и устраняются ошибки. Необходимо наличие проектной документации позволяющей развивать ее возможно вовсе не тем разработчикам которые ее создавали без

Этапы создания программы

- *Постановка задачи (problem definition)* – это точная формулировка требований предъявляемых к работе программы, с описанием *входной* и *выходной* информации, и, возможно, описание подходов к решению задачи.
- *Алгоритм* – точный набор инструкций, описывающих порядок действий исполнителя (компьютера), от допустимых исходных данных для достижения результата решения задачи за конечное время.

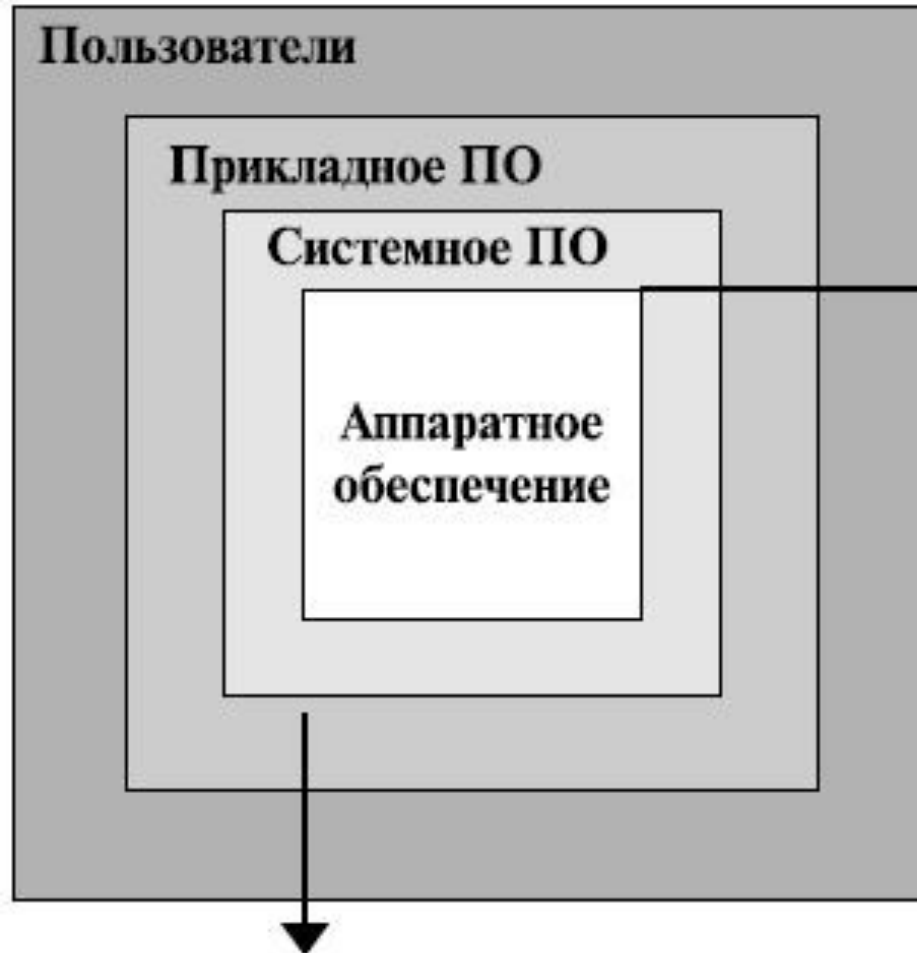
Таким образом, некоторые инструкции должны выполняться строго после завершения работы инструкций, от которых они зависят. Независимые инструкции или инструкции, ставшими независимыми из-за завершения работы инструкций, от которых они зависят, могут выполняться в произвольном порядке, параллельно или одновременно, если это позволяют используемые *процессор* и *операционная система*.

- *Программирование (programming)* – теоретическая и практическая деятельность, связанная с созданием программ.

- *Программный продукт* должен быть соответствующим образом подготовлен к эксплуатации (отлажен), иметь необходимую техническую документацию, предоставлять сервис и гарантию надежной работы программы, иметь *товарный знак* изготовителя, а также код государственной регистрации.

Классы программного обеспечения

•



Системное ПО

Операционная система

- Распределяет ресурсы компьютера
- Планирует использование ресурсов
- Контролирует работу компьютера

Языковые трансляторы

- Интерпретатор
- Компилятор

Утилиты

Обслуживание компьютера и периферийных устройств

Прикладное ПО

Языки программирования

Структура системного программного обеспечения



Структура прикладного программного обеспечения



Классы программных продуктов



Задачи системного ПО

Системное программное обеспечение (System Software) – совокупность программ и программных комплексов, предназначенная для обеспечения работы компьютера и сетей ЭВМ. Системное программное обеспечение выполняет следующие задачи:

- создание операционной среды функционирования других программ;
- обеспечение надежной и эффективной работы самого компьютера и вычислительной сети;
- проведение диагностики, локализации сбоев, ошибок и отказов и профилактики аппаратуры компьютера и вычислительных сетей;
- выполнение вспомогательных технологических процессов (копирование, архивирование, восстановление файлов программ и баз данных и т.д.).

Прикладное программное обеспечение

Прикладное программное обеспечение представляет собой комплекс взаимосвязанных программ, предназначенный для решения задач определенного класса конкретной *предметной области*. *Пакеты прикладных программ (ППП)* общего назначения служат программным инструментарием решения функциональных задач и являются самым многочисленным классом программных продуктов. В данный *класс* входят программные продукты, выполняющие обработку информации различных предметных областей.

- Установка пакетов прикладных программ на *компьютер* выполняется системными администраторами, системными программистами, а также (в некоторых случаях) квалифицированными пользователями. Непосредственную эксплуатацию программных продуктов осуществляют, как правило, *конечные* пользователи – потребители информации, во многих случаях *деятельность* которых весьма далека от компьютерной области. Данный *класс* программных продуктов может быть весьма специфичным для отдельных предметных областей.

Инструментальное ПО

Инструментарий технологии программирования представляет собой совокупность программ и программных комплексов, обеспечивающих технологию разработки, отладки и внедрения создаваемых программных продуктов

Инструментарий технологии программирования включает специализированные программные продукты, которые являются инструментальными средствами разработчика. Программные продукты данного класса поддерживают все технологические этапы процесса проектирования, программирования (кодирования), отладки и тестирования создаваемых программ. Пользователями технологии программирования являются системные и прикладные программисты.