Прочитано на лекции 2

1 семестр 2012 года

Структуры данных

- Для удобства операций с данными их структурируют
- Три основных типа структур данных – *линейная, иерархическая* и *табличная*

Линейная структура (список)

- *Список* – структура данных, в которой каждый элемент определяется своим номером в массиве
- *Векторы* – списки данных, все элементы которых имеют *одинаковую длину*

Табличные и матричные структуры

- Табличные структуры отличаются от списочных тем, что элементы данных определяются от списочных тем, что элементы данных определяются *адресом ячейки,* который зависит не от одного параметра, как в списках, а из нескольких
- В *матрицах* адрес элемента определяется номером строки и номером столбца, на пересечении с которыми находится ячейка, содержащая искомый элемент
- Бывают также *многомерные таблицы*

Иерархические структуры

- *В иерархической структуре адрес каждого элемента определяется путём доступа (маршрутом), ведущим от вершины структуры к данному элементу*

Адресные данные

- Если данные хранятся в любой организованной структуре, то каждый элемент приобретает новое свойство (параметр), которое можно назвать *адресом.*
  Адреса элементов данных сами представляют собой данные нового типа, они также хранятся и обрабатываются

Упорядочение структур данных

- Основным методом упорядочения является *сортировка* (по любому избранному критерию)
- *При добавлении произвольного элемента в упорядоченную структуру списка может происходить изменение адресных данных у других элементов*

Понятие алгоритма

- Алгоритм – формальное описание способа решения задачи посредством выполнения определённой последовательности действий
- В алгоритмах можно найти линейные участки, ветвления, и циклы

Понятие о языках программирования

- Для представления алгоритма в виде, понятном компьютеру, служат *языки программирования*
- Отличие языков программирования от естественных – строгое ограничение число «слов» и строго определённые правила записи команд (*операторов*)

Языки программирования низкого уровня

- Любой из них ориентирован на конкретный тип процессора и учитывает его особенности
- *Язык ассемблера* представляет каждую команду машинного кода не в виде чисел, а с помощью условных обозначений, называемых *мнемониками*
- Преобразование машинных команд в команды ассемблера называется *транслитерацией*

Языки программирования высокого уровня

- *Языки программирования высокого уровня* ближе к естественным и к языку математических обозначений

Пять поколений языков программирования

1. Начало 50-х. Язык ассемблера: одна инструкция – одна строка
2. Рубеж 50-60-х. Появление символических переменных
3. 60-е. Появление универсальных языков высокого уровня
4. С начала 70-х. Проблемно-ориентированные языки, оперирующие конкретными понятиями узкой предметной области
5. С середины 90-х. Системы автоматического создания программ с помощью визуальных средств разработки, без знания программирования

Трансляторы

- *Транслятор* (англ. *translator* — переводчик) — это программа-переводчик. Она преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.
  2 основных типа – *компиляторы* и *интерпретаторы*

Компиляторы

- **Компилятор** (англ. *compiler* — составитель, собиратель) читает всю программу *целиком*, делает ее перевод и создает законченный вариант программы на машинном языке, который затем и выполняется.

Интерпретаторы

- **Интерпретатор** (англ. *interpreter* — истолкователь, устный переводчик) переводит и выполняет программу *строка за строкой*.
- После того, как программа откомпилирована, ни сама исходная программа, ни компилятор более не нужны. В то же время программа, обрабатываемая интерпретатором, должна заново *переводиться* на машинный язык при каждом очередном запуске программы.
- *Откомпилированные* программы работают быстрее, но *интерпретируемые* проще исправлять и изменять.

Промежуточный код

- **Промежуточный код, он же байт-код** или **байткод** (англ. *byte-code*) иногда также используется термин **псевдокод** — машинно-независимый код низкого уровня, генерируемый *транслятором* и исполняемый *интерпретатором*. Большинство инструкций байт-кода эквивалентны одной или нескольким командам ассемблера. Трансляция в байт-код занимает промежуточное положение между компиляцией в машинный код и интерпретацией.

Компьютер

- Компьютер - прибор, одновременно реализующий аппаратные и программные информационные методы
- Результат обработки представляется в виде данных, пригодных для взаимодействия с естественными информационными методами, которыми располагает человек, или с искусственными методами, которыми располагают другие устройства, подключенные к компьютеру.

Внешний подход: модульное устройство персонального компьютера

- Системный блок
- Монитор
- Клавиатура
- Мышь
  - принятая на практике минимальная конфигурация

    Она может наращиваться внешними ***периферийными устройствами*** и внутренними дополнительными устройствами, называемыми ***платами (картами) расширения***

Принтер, плоттер, сканер, фото- и видеокамеры;

Сетевой адаптер;

Модем

Устройство персонального компьютера

- Принцип синхронности состояния частей компьютера. Компьютер – *дискретное устройство*
- Основной тактовый генератор компьютера находится на большой печатной плате, называемой ***системной платой*** (порядка сотен мегагерц)
- Большинство непосредственных вычислений в компьютере выполняются внутри самой важной микросхемы, называемой ***процессором***
- Внутренняя частота процессора в десятки раз больше. Этот конструкторский принцип называют ***внутренним умножением частоты процессора***

Внутреннее устройство системного блока. Материнская плата

- *Процессор*
- *Микропроцессорный комплект(чипсет)* – набор микросхем, управляющих работой внутренних устройств компьютера и определяющих основные функциональные возможности материнской платы
- *Шины* – наборы проводников, по которым происходит обмен сигналами между внутренними устройствами компьютера
- *Оперативная память (ОЗУ)*
- *ПЗУ* – микросхема, предназначенная для длительного хранения данных, в том числе и когда компьютер включён
- Разъёмы для подключения дополнительных устройств (*слоты)*

Шины

- Контроллеры и прерывания, стек процессора
- Шины устройств (помимо системной): PCI (звуки, сетевые адаптеры, внутр. модемы), AGP, PCI Express (видео), USB (для подкл. внешн. устройств), ISA (клавиатура, мышь, дисковод гибких дисков), IDE, SATA (для дисковых устройств), возм., другие

Жёсткий диск

- Представляет собой основное устройство для долговременного хранения больших объёмов данных и программ
- Управление работой выполняет *контроллер жёсткого диска*
- *Скорость обмена данными* с материнской платой: IDE – несколько Мбайт/с, EIDE – до 13-16 Мб/с, SCSI – до 80 Мб/с, SATA – 50 Мб/с более
- *Среднее время доступа* – 9-10 мкс для дисков с частотой 5400 об/мин и 7-8 мкс для 7200 об/мин. У наиболее продвинутых устройств 4-6 мкс

CD и DVD дисководы

- Принцип действия обоих устройств состоит в считывании двоичных данных с помощью лазерного луча, отражающегося от поверхности диска.
- Для CD информация на диске записывается в виде спиральной дорожки из питов (англ. *pit* — углубление), выдавленных (выжженных) в поликарбонатной основе. Каждый пит имеет примерно 100 нм в глубину и 500 нм в ширину. Длина пита варьируется от 850 нм до 3,5 мкм. Промежутки между питами называются лендом (англ. *land* — пространство, основа). Шаг дорожек в спирали составляет 1,6 мкм
- Для DVD используется лазер с меньшей длиной волны, что позволяет вести запись информации с большей плотностью
- Основным параметром является скорость чтения данных. Для CD за единицу принята скорость чтения музыкальных компакт-дисков (150 Кб/с). Имеются устройства со скоростью 48-56x
- Единица скорости для DVD - 1 385 000 байт/с приблизительно в 9 раз больше единицы для CD. Есть устройства 16x

Операционные системы, их основные функции

- Совокупность системных программ, необходимых для согласованной работы всех устройств и прикладных программ компьютера, называют ***операционной системой компьютера***

Основные функции операционных систем

- Распределение ресурсов компьютерной системы: различение прерываний, обеспечение безопасности обращений к памяти, обеспечение разделения времени при обращениях программ к процессору
- Обеспечение единого механизма обращения к накопителям данных разного типа
- Обеспечение единого механизма работы пользователя посредством программ-приложений
- Обобщение программных ресурсов для выполнения типовых действий

Системное программное обеспечение

- BIOS (ПЗУ)
- Драйверы устройств, средства поддержки файловой системы, средства обеспечения интерфейса пользователя

Служебное программное обеспечение

- Программы для обслуживания компьютерной системы, её диагностики и настройки. Некоторые служебные программы могут напрямую обращаться к устройствам, для обслуживания которых они предназначены

Прикладное программное обеспечение

- Класс программ для конкретных практических работ с данными: создание, преобразование, транспортировка, воспроизведение

Многоуровневость программного обеспечения

- Прикладная программа

↓↓

- Драйвер устройства

↓↓

- Работа устройства

Windows – объектно-ориентированная операционная система

- Объекты обладают свойствами и методами – стандартными действиями, связанными с объектами
- 5 типов объектов в Windows: устройства, файлы, программы, информационные связи, объекты контейнерного типа
- Управление компьютером в Windows заключается: в выборе объектов, изменении их свойств, в выборе и активации их методов

Интерфейс пользователя. Объекты и элементы управления

- Все объекты Windows представляются своими графическими образами
- Для доступа к свойствам и методам объектов используется принцип графического управления
- Графическими образами объектов являются: значки объектов, ярлыки объектов, графические элементы управления и окна

Типы графических образов объектов

- Значки объектов представляют устройства, файлы, программы, внешние информационные связи и закрытые контейнеры
- Ярлыки служат для представления внешних информационных связей с объектами
- Графические элементы управления служат для представления внутренних информационных связей объекта. При воздействии на элемент управления происходит либо изменение свойств объекта, либо активизация какого-либо метода, связанного с объектом

Окна

- **Окно папки** – графический контейнер для группового отображения содержимого папки
- **Диалоговое окно** – графический контейнер для группировки элементов управления *Windows*
- **Окно приложения** – графический контейнер, в состав которого входит область воспроизведения документа и элементы управления его содержанием, оформлением и режимом воспроизведения

Принцип графического управления

- Графическое управление Windows основано на так называемом **событийном механизме**.
- Роль активного элемента выполняет **указатель мыши**
- Роль пассивных – графические образы объектов Windows и графические элементы управления

Приёмы графического управления

- Наведение мыши
- Щелчок
- Двойной щелчок
- Специальный («правый») щелчок
- Перетаскивание (для перемещения и копирования)
- Специальное перетаскивание
- Протягивание (в отличие от перетаскивания происходит изменение формы объекта)
- Перекат

Графические элементы управления

- Командные кнопки
- Меню
- Поля ввода (отличие от надписей и подписей)
- Списки выбора
- Раскрывающиеся («выпадающие») списки
- Комбинированные списки
- Флажки
- Переключатели
- Счётчики
- Движки
- Вкладки

Компоненты рабочего стола

- Фоновый рисунок
- Значки рабочего стола
- Панель задач
- Кнопка «Пуск» (неотъемлемая часть Панели задач)
- Панель индикации (неотъемлемая часть Панели задач)
- Панель быстрого запуска

Панель задач. Структура главного меню

- Имя пользователя и его значок
- Список наиболее часто используемых приложений
- Все программы
- Мои документы
- Недавние документы
- Мой компьютер
- Сетевое окружение
- Панель управления
- Подключение
- Принтеры и факсы
- Справка и поддержка
- Поиск
- Выполнить