

Лекция 2.

Алгоритмизация и программирование. Графические элементы управления.

Этапы компьютерного решения задач

2

- 1) Постановка задачи
 - сбор информации о задаче
 - формулировка условия задачи
 - определение конечных целей решения задачи
 - определение формы выдачи результатов
 - описание данных (их типов, диапазонов величин, структуры и т.п.)
- 2) Анализ и исследование задачи, модели
 - анализ существующих аналогов
 - анализ технических и программных средств
 - разработка математической модели
 - разработка структур данных

Этапы компьютерного решения задач (продолжение)

3

- 3) Разработка алгоритма
 - выбор метода проектирования
 - проектирование алгоритма
- 4) Программирование
 - выбор языка программирования
 - запись алгоритма на выбранном языке программирования
- 5) Тестирование и отладка
 - синтаксическая отладка
 - отладка семантики и логической структуры
 - тестовые расчеты и анализ результатов тестирования
 - совершенствование программы

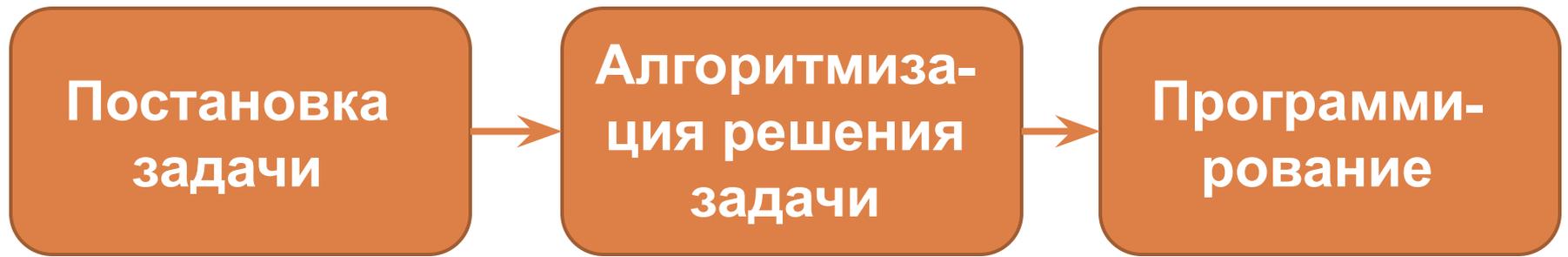
Этапы компьютерного решения задач (продолжение)

4

- 6) Анализ результатов решения задачи и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5
- 7) Сопровождение программы
 - доработка программы для решения конкретных задач
 - составление документации к решенной задаче, математической модели, алгоритму, программе, по их использованию

Процесс создания программ

5



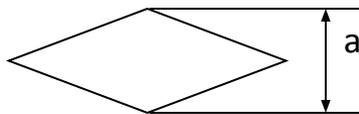
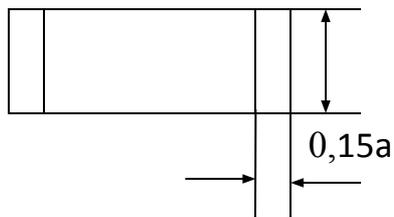
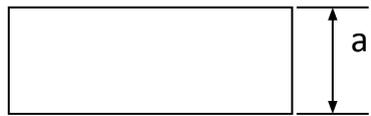
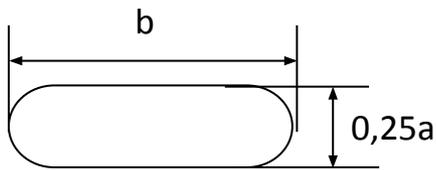
Процесс создания программ

6

- **Постановка задачи** - это точная формулировка решения задачи на компьютере с описанием входной и выходной информации
- **Алгоритм** - система точно сформулированных правил, определяющая процесс преобразования допустимых исходных данных (входной информации) в желаемый результат (выходную информацию) за конечное число шагов
- **Программирование** - создание последовательности команд, необходимой для решения определенной задачи

Графическая форма записи алгоритмов – БСА (блок-схема)

7

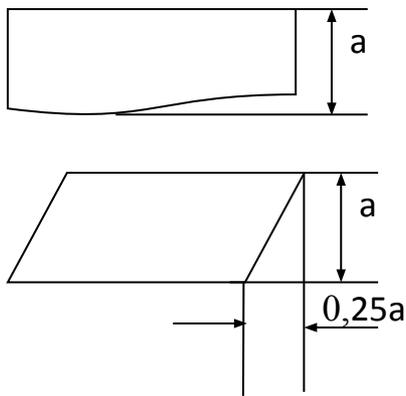


$$b=2*a$$

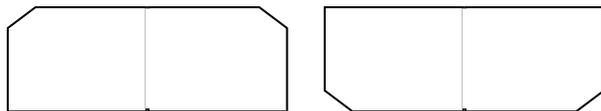
- Блок пуск – останов. Начало, конец процесса, вход и выход в программах
- Операционный блок (Процесс). Выполнение операции или группы операций.
- Предопределенный процесс. Использование отдельно описанных алгоритмов, процедур, подпрограмм.
- Блок принятия решения. Проверка условия.

БСА (блок-схема алгоритма, продолжение)

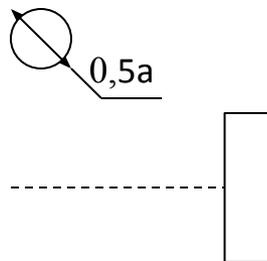
8



- Документ. Вывод результатов на дисплей, печать.
- Ввод-вывод данных. Преобразование данных в форму, пригодную для обработки (ввод) или отображение результатов обработки (вывод).



Начало и конец цикла.



- Соединитель. Указание связи между прерванными линиями потока.
- Комментарий

Язык программирования

- Различают два уровня языков программирования:
 - ▢ *язык программирования низкого уровня* - язык программирования, ориентированный на конкретный тип процессора, т.е. операторы языка близки к машинному коду и ориентированы на конкретные команды процессора (ассемблер).
 - ▢ *язык программирования высокого уровня* значительно ближе и понятнее человеку, нежели компьютеру. Особенности конкретных компьютерных архитектур в них не учитываются, поэтому создаваемые программы на уровне исходных текстов легко переносимы на другие платформы, для которых создан транслятор этого языка.

Системы программирования

10

- Компоненты:
 - Текстовый редактор – для записи программного кода на выбранном языке программирования;
 - Компилятор – переводит исходный текст программы в машинный код;
 - Сборщик - выполняет связывание модулей и библиотек и формирует на выходе работоспособное приложение - исполнимый код для конкретном платформы.

Процесс компиляции

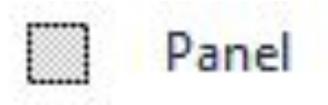
- ▣ **Лексический анализ.** ЛЕКСИКА языка программирования -это правила "правописания слов" программы (идентификаторы, константы, служебные слова, комментарии).
- ▣ **Синтаксический анализ.** СИНТАКСИС языка программирования - это правила составления предложений языка из отдельных слов (операции, операторы, определения функций и переменных).
- ▣ **Семантический анализ.** СЕМАНТИКА языка программирования - это смысл, который закладывается в каждую конструкцию языка. Семантический анализ -это проверка смысловой правильности конструкции.
- ▣ **ГЕНЕРАЦИЯ КОДА** -это преобразование элементарных действий, полученных в результате лексического, синтаксического и семантического анализа программы, в некоторое внутреннее представление.

Графические элементы управления

12

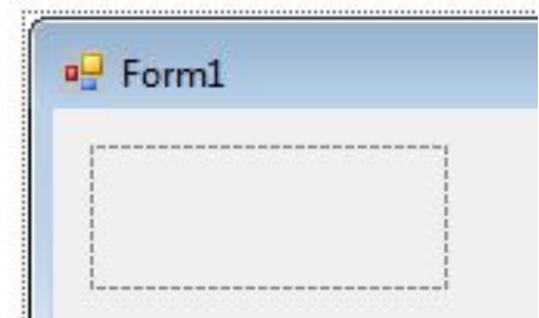
- Уже рассмотрены: Forms, Label, TextBox, Button
- Будут рассмотрены:
 - Panel (Панель) ,
 - GroupBox (Рамка),
 - CheckBox (Флажок),
 - RadioButton (Переключатель),
 - ListBox (Список),
 - ComboBox (Выпадающий список),
 - HScrollBar, VScrollBar (Полоса прокрутки)

Элемент управления Panel (Панель)



13

- **Panel (Панель)** – это контейнер, который служит для объединения элементов в группу, после чего, помещенными в него объектами, можно управлять как единым целым.
- **Основные свойства:** Name, BackColor, ForeColor, Font, Enabled, Visible, Size и др.

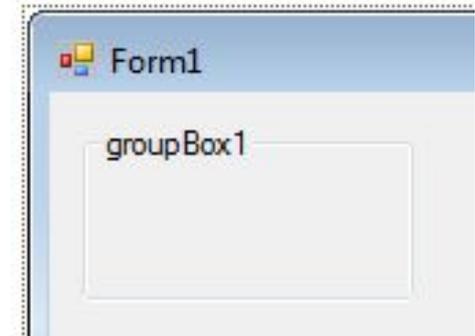


Элемент управления GroupBox (Рамка)

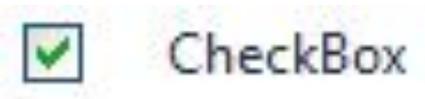


14

- ▣ **GroupBox (Рамка)** – это контейнер, который служит для объединения элементов в группу, после чего, помещенными в него объектами, можно управлять как единым целым.
- ▣ **Основные свойства:** Name, Text, BackColor, ForeColor, Font, Enabled, Visible, Size и др.

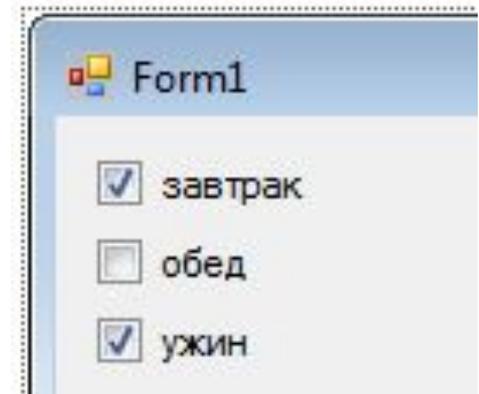


Элемент управления CheckBox (Флажок)



15

- ▣ **CheckBox (Флажок)** позволяет пользователю выбрать значение из списка параметров или сочетание параметров.
- ▣ **Основные свойства:** Name, Text, Appearance, ThreeState, Checked, CheckState, FlatStyle, BackColor, ForeColor, Font, Enabled, Visible и др.
- ▣ **Основное событие:** CheckedChanged.



Элемент управления

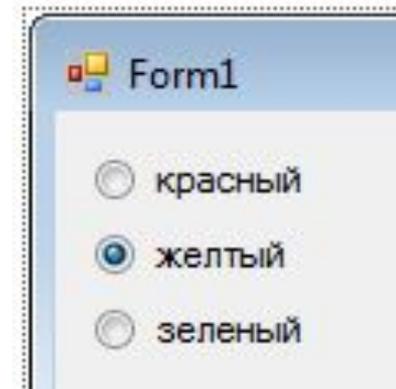
RadioButton (Переключатель)



RadioButton

16

- ▣ ***RadioButton (Переключатель)*** представляют собой набор как минимум из двух взаимоисключающих вариантов выбора для пользователя.
- ▣ ***Основные свойства:*** Name, Text, Appearance, Checked, FlatStyle, BackColor, ForeColor, Font, Enabled, Visible и др.
- ▣ ***Основные события:*** Click, CheckedChanged.

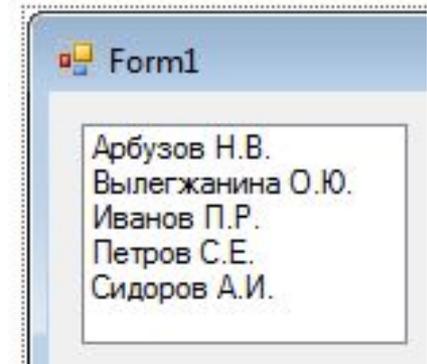


Элемент управления ListBox (Список)

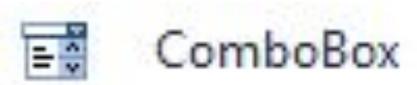


17

- ▣ **ListBox (Список)** позволяет отобразить список позиций, из которых пользователь может выбрать нужную щелчком мыши.
- ▣ **Основные свойства:** Name, Items, MultiColumn, SelectedItem, Font, Enabled, Visible и др.
- ▣ **Основные методы:** Add(), AddRange(), Remove(), RemoveAt(), Insert().



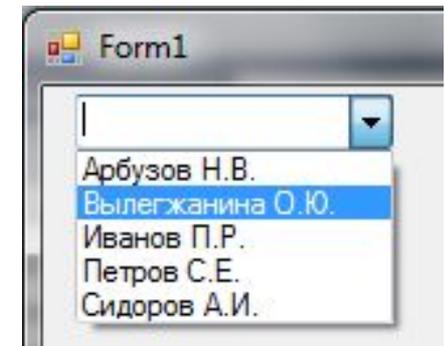
Элемент управления



ComboBox (Выпадающий список)

18

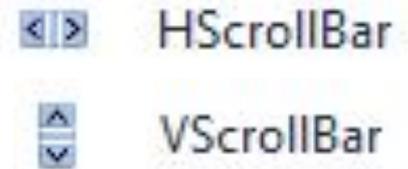
- ▣ **ComboBox (Выпадающий список)** отображает редактируемое поле вместе со списком ListBox и позволяет пользователю выбрать значение из списка или ввести новый текст.
- ▣ **Основные свойства:** Name, Items, DropDownStyle, SelectedItem, Font, Enabled, Visible и др.
- ▣ **Основные методы:** Add(), AddRange(), Remove(), RemoveAt(), Insert().



Элемент управления HScrollBar, VScrollBar (Полоса прокрутки)

19

- **HScrollBar (Горизонтальная полоса прокрутки), VScrollBar (Вертикальная полоса прокрутки)** используется для реализации возможности прокрутки в контейнерах, не предоставляющих собственных полос прокрутки, или для ввода числовых данных вручную.
- **Основные свойства:** Name, Value, Minimum, Maximum, SmallChange, Enabled, Visible и др.
- **Основное событие:** Scroll.



20

Спасибо за внимание

Вопросы...