

Содержательная линия  
«Алгоритмизации и  
программирования»

Лектор: Ергалиев Ерлан

# Задание

- Приготовить ознакомительную презентацию по одной из тем содержательной линии
- Выступить с показом презентации
- Подготовить пример практического задания по выбранной теме

# Темы для выбора

- Алгоритмы. Исполнители. Виды алгоритмов
- Циклические алгоритмы
- Языки программирования
- Программа и ее структура
- Программирование условий(*if, then*)
- Операторы *while...do...*
- Операторы *repeat...until...*
- Оператор *for*

# Правильная презентация

# Дизайн и эффекты

- ✓ Светлый дизайн
- ✓ Темные надписи
- ✓ Единый стиль
- ✓ Наглядность

# Дизайн и эффекты

- Светлый дизайн
- Темные надписи

# ДИЗАЙН И ЭФФЕКТЫ

---

- ✓ Светлый дизайн
- ✓ Темные надписи
- ✓ Единый стиль
- ✓ Наглядность

# Дизайн и эффекты

- ✓ Светлый дизайн
- ✓ Темные надписи
- ✓ Единый стиль
- ✓ Наглядность



# Шрифт

- ✓ Стандартный шрифт
- ✓ Видимый размер

# *Шрифт*

- *Классический шрифт*
- *Видимый размер*



# Шрифт

- ✓ Стандартный шрифт
- ✓ Видимый размер

# Содержание

- Указывайте список тем (основные понятия, содержание выступления)
- Количество текста

# Перечень понятий

- ✓ Алгоритм
- ✓ Свойства алгоритмов
- ✓ Исполнители алгоритмов
- ✓ Система команд исполнителя
- ✓ Программирование
- ✓ Языки программирования
- ✓ Вспомогательные алгоритмы

# Содержание

Виды алгоритмов как логико-математических средств отражают указанные компоненты человеческой деятельности и тенденции, а сами алгоритмы в зависимости от цели, начальных условий задачи, путей её решения. Следует подчеркнуть принципиальную разницу между алгоритмами вычислительного характера, преобразующими некоторые входные данные в выходные (именно их формализацией являются упомянутые выше машины Тьюринга, Поста, РАМ, нормальные алгоритмы Маркова и рекурсивные функции), и интерактивными алгоритмами (уже у Тьюринга встречается С-машина, от [англ. choice](#) — выбор, ожидающая внешнего воздействия, в отличие от классической А-машины, где все исходные данные заданы до начала вычисления и выходные данные недоступны до окончания вычисления). Последние предназначены для взаимодействия с некоторым объектом управления и призваны обеспечить корректную выдачу управляющих воздействий в зависимости от складывающейся ситуации, отражаемой поступающими от объекта управления сигналами [\[13\]](#). В некоторых случаях алгоритм управления вообще не предусматривает окончания работы (например, поддерживает бесконечный цикл ожидания событий, на которые выдается соответствующая реакция), несмотря на это, являясь полностью правильным.

Можно также выделить алгоритмы:

- **Механические алгоритмы**, или иначе **детерминированные, жесткие** (например, алгоритм работы машины, двигателя и т. п.) — задают определенные действия, обозначая их в единственной и достоверной последовательности, обеспечивая тем самым однозначный требуемый или искомый результат, если выполняются те условия процесса, задачи, для которых разработан алгоритм.
- **Гибкие алгоритмы**, например, стохастические, то есть вероятностные и эвристические.
- **Вероятностный** (стохастический) алгоритм дает программу решения задачи несколькими путями или способами, приводящими к вероятному достижению результата.
- **Эвристический алгоритм** (от греческого слова «[эврика](#)») — алгоритм, использующий различные разумные соображения без строгих обоснований [\[13\]](#).
- **Линейный алгоритм** — набор команд (указаний), выполняемых последовательно во времени друг за другом.
- **Разветвляющийся алгоритм** — алгоритм, содержащий хотя бы одно условие, в результате проверки которого может осуществляться разделение на несколько альтернативных ветвей алгоритма.
- **Циклический алгоритм** — алгоритм, предусматривающий многократное повторение одного и того же действия (одних и тех же операций) над новыми исходными данными. К циклическим алгоритмам сводится большинство методов вычислений, перебора вариантов. Цикл программы — последовательность команд (серия, тело цикла), которая может выполняться многократно (для новых исходных данных) до удовлетворения некоторого условия.
- **Вспомогательный (подчиненный) алгоритм (процедура)** — алгоритм, ранее разработанный и целиком используемый при алгоритмизации конкретной задачи. В некоторых случаях при наличии одинаковых последовательностей указаний (команд) для различных данных с целью сокращения записи также выделяют вспомогательный алгоритм. На всех этапах подготовки к алгоритмизации задачи широко используется структурное представление алгоритма.
- **Структурная блок-схема, граф-схема алгоритма** — графическое изображение алгоритма в виде схемы связанных между собой с помощью стрелок (линий перехода) блоков — графических символов, каждый из которых соответствует одному шагу алгоритма. Внутри блока дается описание соответствующего действия. Графическое изображение алгоритма широко используется перед программированием задачи вследствие его наглядности, так как зрительное восприятие обычно облегчает процесс написания программы, её корректировки при возможных ошибках, осмысление процесса обработки информации.

Можно встретить даже такое утверждение: «Внешне алгоритм представляет собой схему — набор прямоугольников и других символов, внутри которых

# Содержание

Можно также выделить алгоритмы:

- **Механические алгоритмы**, или иначе **детерминированные, жесткие** (например, алгоритм работы машины, двигателя и т. п.) — задают определенные действия, обозначая их в единственной и достоверной последовательности, обеспечивая тем самым однозначный требуемый или искомый результат, если выполняются те условия процесса, задачи, для которых разработан алгоритм.
- **Гибкие алгоритмы**, например, стохастические, то есть вероятностные и эвристические.
- **Вероятностный** (стохастический) алгоритм дает программу решения задачи несколькими путями или способами, приводящими к вероятному достижению результата.
- **Эвристический алгоритм** (от греческого слова «[эврика](#)») — алгоритм, использующий различные разумные соображения без строгих обоснований<sup>[13]</sup>.
- **Линейный алгоритм** — набор команд (указаний), выполняемых последовательно во времени друг за другом.
- **Разветвляющийся алгоритм** — алгоритм, содержащий хотя бы одно условие, в результате проверки которого может осуществляться разделение на несколько альтернативных ветвей алгоритма.
- **Циклический алгоритм** — алгоритм, предусматривающий многократное повторение одного и того же действия (одних и тех же операций) над новыми исходными данными. К циклическим алгоритмам сводится большинство методов вычислений, перебора вариантов. Цикл программы — последовательность команд (серия, тело цикла), которая может выполняться многократно (для новых исходных данных) до удовлетворения некоторого условия.
- **Вспомогательный (подчиненный) алгоритм (процедура)** — алгоритм, ранее разработанный и целиком используемый при алгоритмизации конкретной задачи. В некоторых случаях при наличии одинаковых последовательностей указаний (команд) для различных данных с целью сокращения записи также выделяют вспомогательный алгоритм. На всех этапах подготовки к алгоритмизации задачи широко используется структурное представление алгоритма.
- **Структурная блок-схема, граф-схема алгоритма** — графическое изображение алгоритма в виде схемы связанных между собой с помощью стрелок (линий перехода) блоков — графических символов, каждый из которых соответствует одному шагу алгоритма. Внутри блока дается описание соответствующего действия. Графическое изображение алгоритма широко используется перед программированием задачи вследствие его наглядности, так как зрительное восприятие обычно облегчает процесс написания программы, её корректировки при возможных ошибках, осмысливание процесса обработки информации.

Можно встретить даже такое утверждение: «Внешне алгоритм представляет собой схему — набор прямоугольников и других символов, внутри

# Содержание

Можно также выделить алгоритмы:

- ✓ Механические алгоритмы, или иначе детерминированные, жесткие
- ✓ Гибкие алгоритмы
- ✓ Вероятностный (стохастический)
- ✓ Эвристический алгоритм
- ✓ Линейный алгоритм
- ✓ Разветвляющийся алгоритм
- ✓ Циклический алгоритм
- ✓ Вспомогательный (*подчиненный*) алгоритм
- ✓ Структурная блок-схема, граф-схема алгоритма



Источник

Всегда указывайте

**ИСТОЧНИК**

# Содержание

Можно также выделить алгоритмы:

- ✓ Механические алгоритмы, или иначе детерминированные, жесткие
- ✓ Гибкие алгоритмы
- ✓ Вероятностный (стохастический)
- ✓ Эвристический алгоритм
- ✓ Линейный алгоритм
- ✓ Разветвляющийся алгоритм
- ✓ Циклический алгоритм
- ✓ Вспомогательный (*подчиненный*) алгоритм
- ✓ Структурная блок-схема, граф-схема алгоритма

Ключевые слова

Выделяйте ключевые  
слова

# Исполнитель

- **Субъект**, выполняющий алгоритм.
- **Объект (или субъект)**, для управления которым составляется алгоритм.
- Основной характеристикой исполнителя, с точки зрения управления, является *система команд исполнителя* (СКИ). Это конечное множество команд, которые понимает исполнитель, т.е. умеет их выполнять.
- Робот, компьютер и человек.

## Выступление

Не пересказывайте то, что  
**можно прочесть**

# Способы описания алгоритмов

**Блок-схемы** – схема, описывающая алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.

**Учебный алгоритмический язык** - это текстовая форма описания алгоритма.

Вопросы???