**1ноября информатика 1 курс юристы.**

**Выписать все определения и свойства в тетрадь и прислать мне.**

**Методическая разработка**

**теоретического (лекционного) занятия № 6**

**Алгоритмы и способы их описания, основные алгоритмические конструкции**

**Входной контроль (фронтальный опрос) с эталонами ответа**

1. Что такое информационный процесс?

Информационный процесс - это совокупность последовательных действий, производимых над информацией с целью получения результата. Это поиск - сбор - хранение - передача - обработка - использование – защита информации.

2. Назовите типы информационных процессов.

Основные типы информационных процессов

|  |  |
| --- | --- |
| **Получение** информации | Это сбор сведений из каких либо источников (извлечение данных из хранилища/источника данных, наблюдение за событиями и явлениями, общение, СМИ и масс-медиа). Получение информации основано на отражении различных свойств процессов, объектов и явлений окружающей среды. Этот процесс выражается в восприятии с помощью органов чувств. Для улучшения восприятия информации человек придумал различные индивидуальные приспособления и приборы — очки, бинокль, микроскоп, стетоскоп, различные датчики и т. д. |
| **Хранение**  информации | Хранение информации имеет большое значение для многократного использования информации и передачи информации во времени. Для долговременного хранения используются книги, в настоящее время — компьютерные носители, устройства внешней памяти и др. Информация чаще всего хранится для неоднократной дальнейшей работы с ней. В этом случае для ускорения поиска информация должна быть как-то упорядочена. В библиотеках — это картотеки, при хранении с использованием компьютера — размещение информации в определенных папках, в более сложных случаях — это базы данных, информационно-поисковые системы и т. д. |
| **Обработка**  информации | Обработка информации подразумевает преобразование ее к виду, отличному от исходной формы или содержания информации. Процесс изменения информации может включать в себя, например, такие действия как численные расчёты, редактирование, упорядочивание, обобщение, систематизация и т. д. Результаты обработки информации в дальнейшем используются в тех или иных целях, например: получение новой информации из уже известной путем логических рассуждений или математических вычислений (например, решение геометрической задачи); изменение формы представления информации без изменения ее содержания (например, перевод текста с одного языка на другой); упорядочение (сортировка) информации (например, упорядочение расписания движения поездов по времени их отправления). |
| **Передача**  информации | Передача информации необходима для её распространения. Основными устройствами для быстрой передачи информации на большие расстояния в настоящее время являются телеграф, радио, телефон, телевизионный передатчик, телекоммуникационные сети на базе вычислительных систем. Такие средства связи принято называть каналами передачи информации. Следует отметить, что в процессе передачи информации, она может искажаться или теряться. Это происходит в тех случаях, когда информационные каналы плохого качества или на линии связи присутствуют помехи. Передача информации — это всегда двусторонний процесс, в котором есть источник и есть приемник информации. Источник передает информацию, а приемник её получает. |

3. Что такое обработка информации?

Обработка информации - процесс получения новой информации на базе уже имеющейся. Преобразование информации может быть связано с изменением ее содержания или формы представления. В последнем случае говорят о кодировании информации. Например, шифрование информации или перевод текстов на другой язык. Упорядочивание информации (расписания), поиск нужной информации в информационном массиве (номер телефона в телефонной книге) являются другими вариантами обработки. Редактирование текста, математические вычисления, логические умозаключения - примеры процедур получения новой информации.

Обработка информации может производиться формально, руководствуясь правилами по заданному алгоритму. Либо применяется эвристический подход, при котором создаётся новая система действий или открываются неизвестные ранее закономерности изучаемой информации.

5. Какие системы счисления использует компьютер, почему?

Двоичная система счисления является стандартом при конструировании компьютеров. Десятичная система счисления используется для организации ввода / вывода информации, двоичная система счисления – для организации машинных операций по преобразованию информации. 8-миричная и 16-тиричная системы используются для более короткой и удобной записи, т. к. требует меньше разрядов(для записи программ в машинных кодах). Важную роль при рассмотрении арифметических основ работы компьютера играют единицы измерения количества информации, которые мы также рассматривали при изучении подходов к измерению количества информации.

6. Что такое алгебра логики? Какие логические операции вы знаете?

Процессор выполняет арифметические и логические операции над двоичными кодами. Поэтому необходимо познакомиться с основными логическими элементами, лежащими в основе его построения. Начнем с алгебры логики.

Алгеброй логики называется аппарат, который позволяет выполнять действия над высказываниями. Высказывание - это предложение, относительно которого имеет смысл говорить истинно оно или ложно. Высказывания могут быть представлены с помощью математических, химических и прочих знаков.

Алгебру логики называют также алгеброй Буля, или булевой алгеброй, по имени английского математика Джорджа Буля, разработавшего в XIX веке ее основные положения. В булевой алгебре высказывания принято обозначать прописными латинскими буквами: А, В, X, Y. В алгебре Буля введены три основные логические операции с высказываниями: сложение, умножение, отрицание. Определены аксиомы (законы) алгебры логики для выполнения этих операций. Действия, которые производятся над высказываниями, записываются в виде логических выражений. Алгебра логики рассматривает высказывания не с точки зрения их содержания, а с точки зрения их истинности или ложности. И в этом смысле можно сказать, что высказывание может принимать только два значения: ИСТИНА (обозначим 1) или ЛОЖЬ (обозначим 0).

Логические выражения могут быть простыми и сложными. Простое логическое выражение состоит из одного высказывания и не содержит логические операции. В простом логическом выражении возможно только два результата — либо «истина», либо «ложь». Сложное логическое выражение содержит высказывания, объединенные логическими операциями. По аналогии с понятием функции в алгебре сложное логическое выражение содержит аргументы, которыми являются высказывания. В качестве основных логических операций в сложных логических выражениях используются следующие: НЕ (логическое отрицание, инверсия); ИЛИ (логическое сложение, дизъюнкция); И (логическое умножение, конъюнкция).

**Алгоритмы и способы их описания**

Алгоритм – система точных и понятных предписаний (команд, инструкций, директив) о содержании и последовательности выполнения конечного числа действий, необходимых для решения любой задачи данного типа. Как всякий объект, алгоритм имеет название (имя). Также алгоритм имеет начало и конец.

В качестве исполнителя алгоритмов можно рассматривать человека, любые технические устройства, среди которых особое место занимает компьютер. Компьютер может выполнять только точно определенные операции, в отличии от человека, получившего команду и имеющего возможность сориентироваться в ситуации.

Алгоритм обладает следующими свойствами.

1. Дискретность (от лат. discretus – разделенный, прерывистый) указывает, что любой алгоритм должен состоять из конкретных действий, следующих в определенном порядке.

2. Детерминированность (от лат. determinate – определенность, точность) указывает, что любое действие алгоритма должно быть строго и недвусмысленно определено в каждом случае.

3. Конечность определяет, что каждое действие в отдельности и алгоритм в целом должны иметь возможность завершения.

4. Результативность требует, чтобы в алгоритме не было ошибок, т.е. при точном исполнении всех команд процесс решения задачи должен прекратиться за конечное число шагов и при этом должен быть получен ответ.

5. Массовость заключается в возможности применения алгоритма к целому классу однотипных задач, различающихся конкретными значениями исходных данных (разработка в общем виде).

Способы описания алгоритмов:

• словесный (на естественном языке);

• графический (с помощью стандартных графических объектов (геометрических фигур) – блок-схемы);

• программный (с помощью языков программирования).

Виды алгоритмов: линейный, циклический, разветвляющийся.

Линейный алгоритм – описание действий, которые выполняются однократно, при этом четко друг за другом.

Циклический алгоритм – описание действий, которые должны повторяться определенное количество раз или пока не выполнится условие.

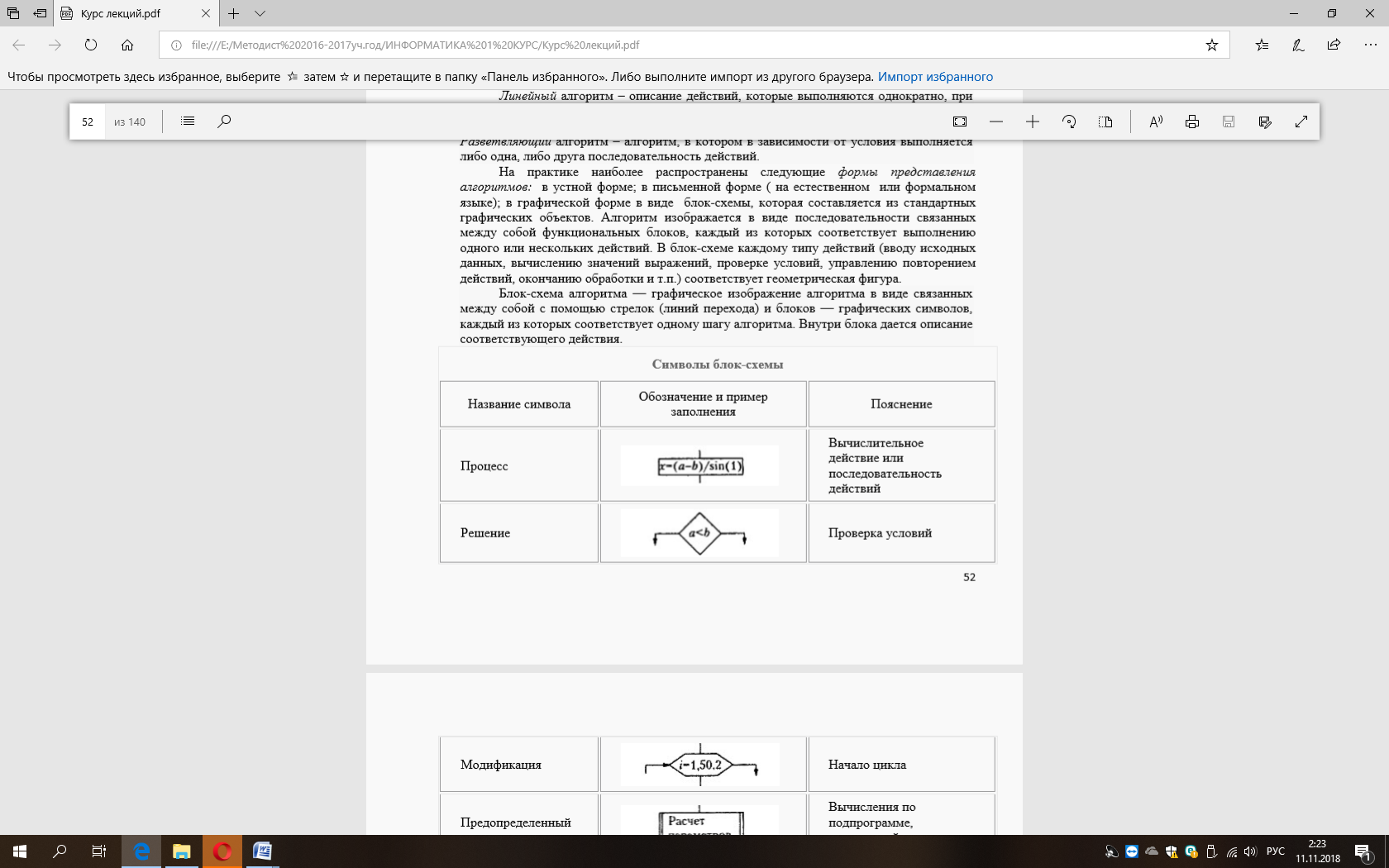
Разветвляющий алгоритм – алгоритм, в котором в зависимости от условия выполняется либо одна, либо друга последовательность действий.

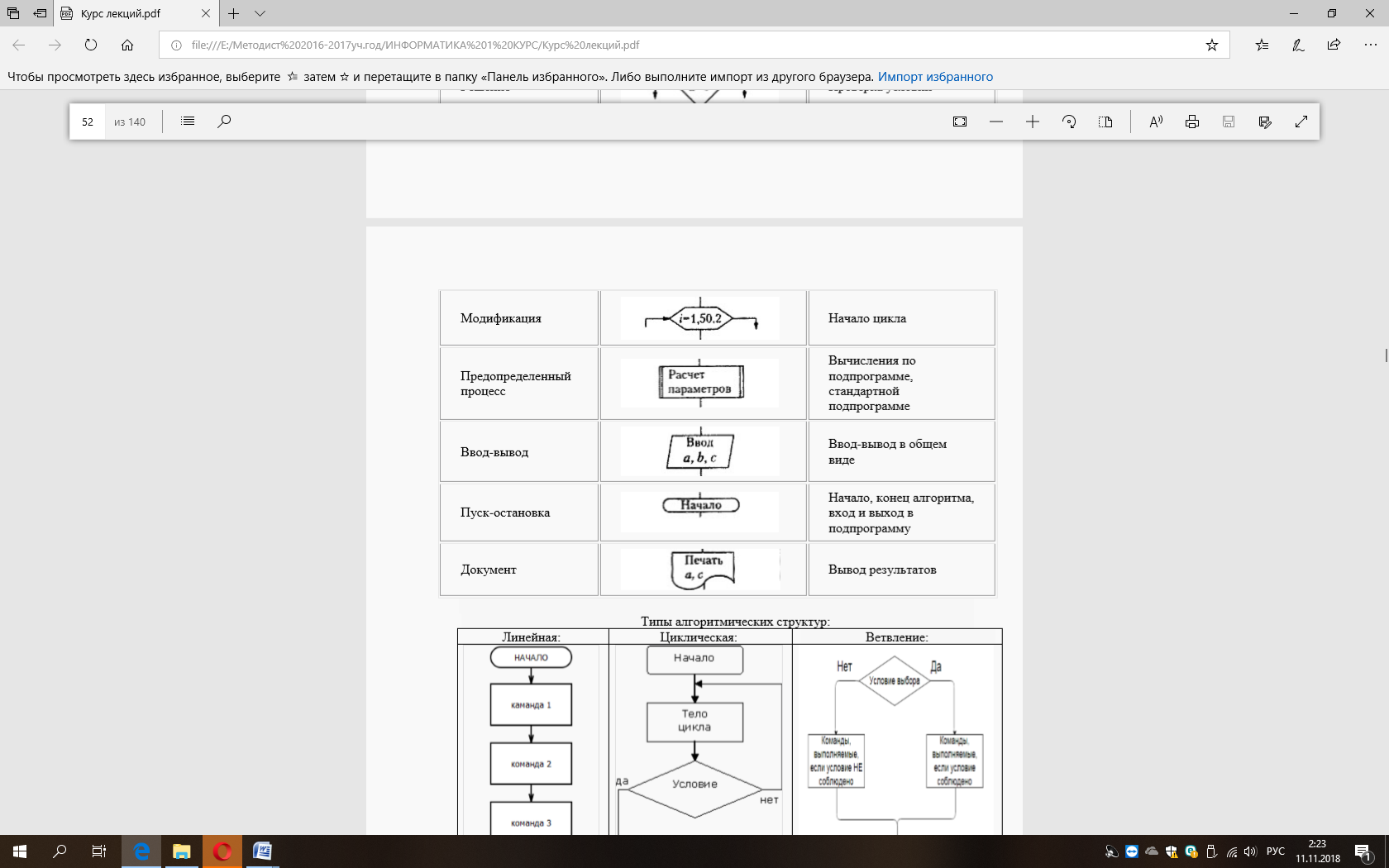
На практике наиболее распространены следующие формы представления алгоритмов: в устной форме; в письменной форме (на естественном или формальном языке); в графической форме в виде блок-схемы, которая составляется из стандартных графических объектов.

Алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий. В блок-схеме каждому типу действий (вводу исходных данных, вычислению значений выражений, проверке условий, управлению повторением действий, окончанию обработки и т.п.) соответствует геометрическая фигура.

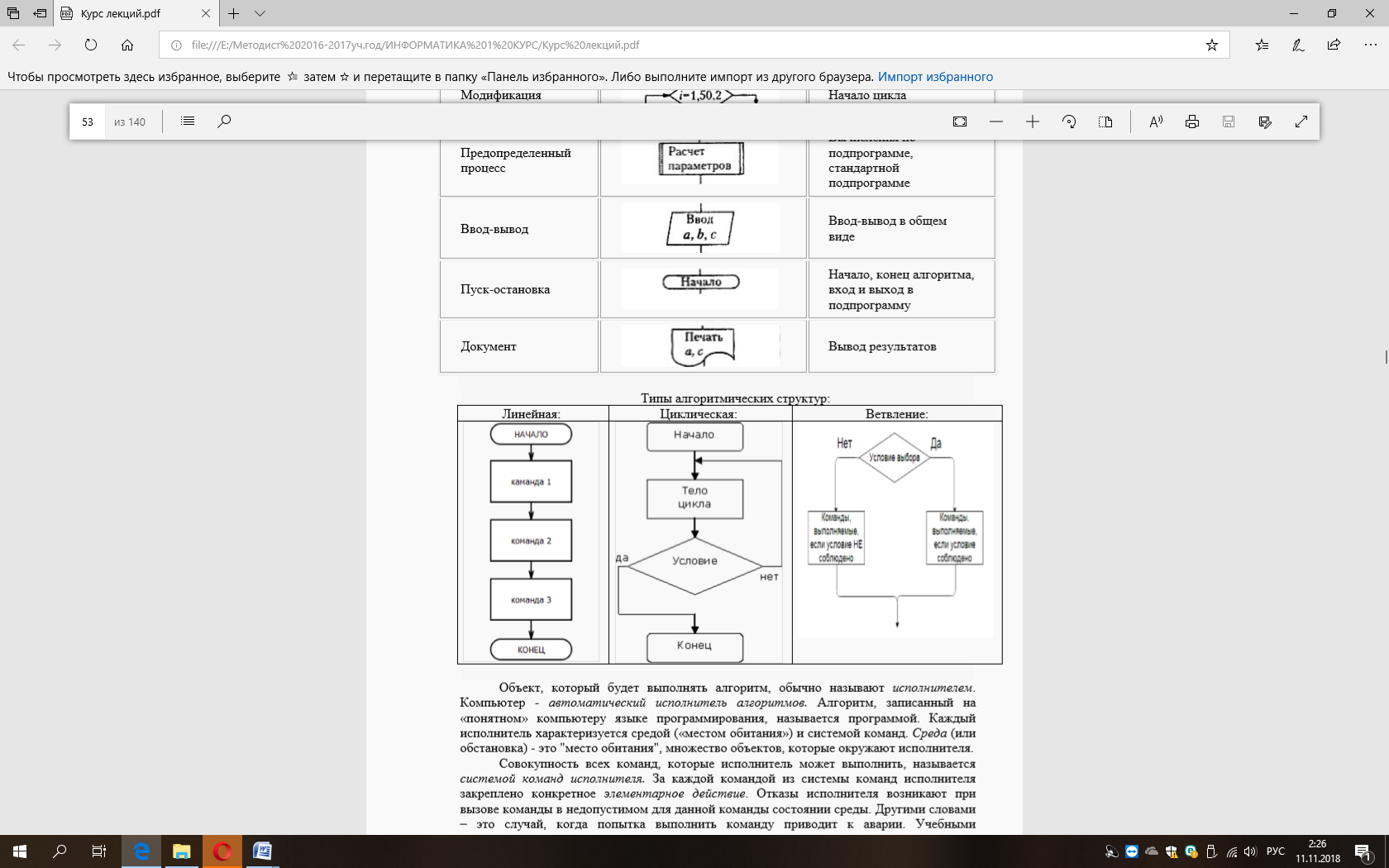
**Блок-схема алгоритма**

Блок-схема алгоритма — графическое изображение алгоритма в виде связанных между собой с помощью стрелок (линий перехода) и блоков — графических символов, каждый из которых соответствует одному шагу алгоритма. Внутри блока дается описание соответствующего действия.





**Типы алгоритмических структур**



Объект, который будет выполнять алгоритм, обычно называют исполнителем. Компьютер - автоматический исполнитель алгоритмов. Алгоритм, записанный на «понятном» компьютеру языке программирования, называется программой. Каждый исполнитель характеризуется средой («местом обитания») и системой команд. Среда (или обстановка) - это «место обитания», множество объектов, которые окружают исполнителя.

Совокупность всех команд, которые исполнитель может выполнить, называется системой команд исполнителя. За каждой командой из системы команд исполнителя закреплено конкретное элементарное действие. Отказы исполнителя возникают при вызове команды в недопустимом для данной команды состоянии среды. Другими словами – это случай, когда попытка выполнить команду приводит к аварии. Учебными исполнителями называют различные образы на экране компьютера, которыми можно управлять, отдавая команды. Используются они для обучения составлению управляющих алгоритмов.

Описание последовательности действий только тогда является алгоритмом, когда у разных исполнителей, точно выполняющих все предписания, всегда получается один и тот же результат. Это возможно только тогда, когда исполнитель не изменяет порядок действий, не дополняет систему команд исполнителя (СКИ) и не «улучшает» отдельные действия, не интересуется целью, не отступает от описания, не может дать отказ «не хочу». Такой исполнитель называется формальным. Заметим, что все алгоритмы создаются именно для формальных исполнителей.

В отличие от формального, неформальный исполнитель всегда интересуется, зачем он выполняет то или иное действие, стремится выполнить его лучше, эффективнее (с его точки зрения).

Технические устройства (в том числе и компьютер как совокупность аппаратного и программного обеспечения) могут быть только формальными исполнителями, в этом случае результат их действий предсказуем. Человек-пользователь не объясняет исполнителю-компьютеру свои цели и смысл команд программы. Очевидно, что компьютер и не сможет понять смысла совершаемых им действий. Более того, компьютер не обладает способностью к анализу результатов, например, относительно их соответствия постановке задачи.

Работа ни одного устройства компьютера не обходится без программы и исходных данных к ней. Не только устройства компьютера, но и каждое программное средство можно рассматривать в качестве формального исполнителя, который характеризуется своей системой команд, средой, системой отказов.

В любой программе пользовательский интерфейс можно рассматривать как среду исполнителя; сообщения об ошибочных действиях пользователя или о сбоях в работе программы или устройств компьютера как систему отказов; команды управления программой (параметры команд, система меню, «горячие клавиши» и т.д.) как систему команд исполнителя.

Компьютер не сможет выполнить команду, если она предназначается устройству, не подключенному к компьютеру, или отсутствует в его системе команд, или содержит синтаксическую ошибку. Если программа составлена правильно, то процессор выполнит ее за конечное число шагов и выдаст человеку результат решения задачи.

Человек, принимая во внимание этот факт, должен понимать ограниченность возможностей компьютера как исполнителя и предусматривать тонкость в задании команд, поручаемых компьютеру для исполнения, так как вся ответственность за использование компьютеров лежит только на людях.

**Примеры**

Пример 1. Линейным является алгоритм перевоза через реку Волка, Козы и Капусты, при условии, что в лодке можно перевозить только один из указанных объектов, а на берегу вместе не могут находиться Коза и Волк, а также Коза и Капуста:

1) перевези Козу;

2) вернись на исходный берег;

3) перевези Волка;

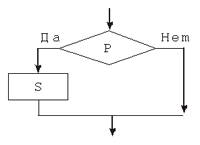
4) вернись на исходный берег с Козой;

5) перевези Капусту;

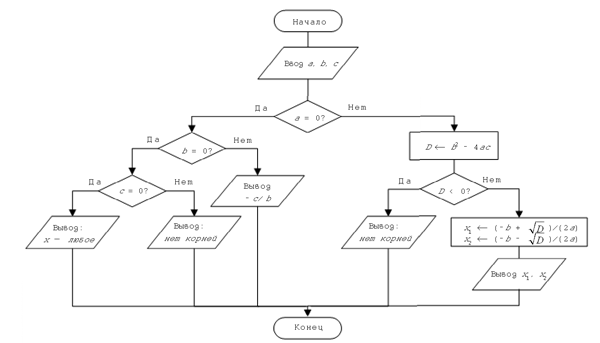
6) вернись на исходный берег;

7) перевези Козу.

Алгоритм *P* реализован с использованием ветвящейся алгоритмической конструкции(*ветвления*), если на каком-либо шаге последовательное выполнение алгоритма прерывается, и выбор следующего шага определяется входными данными алгоритма. *Ветвление* задает выполнение либо одной, либо другой группы операторов в зависимости от выполнения какого-либо условия (в зависимости от истинности или ложности соответствующего логического выражения. Затем исполнение алгоритма выходит на общее продолжение. Для конкретных входных данных ветвящаяся алгоритмическая конструкция сводится к последовательной алгоритмической конструкции. Ветвление бывает полным и неполным. В случае неполного ветвления при невыполнении условия никакие действия не выполняются.



Пример 2.Запишем в словесной форме алгоритм решения уравнения *ax*2 + *bx* + *c* = 0, где *a*, *b*, *c* — произвольные действительные числа, а *x* — искомая величина. Если *a* ≠ 0, то для нахождения корней используются известные формулы, при этом существование корней зависит от знака дискриминанта квадратного уравнения. Если *a* = 0, то уравнение становится линейным. Однако при *b* = 0 оно вырождается в равенство *c* = 0. Поэтому, если *c* действительно равно 0, то все действительные числа являются корнями такого уравнения, в противном случае — корней нет. Ниже приведена блок-схема данного алгоритма.



**Вопросы для закрепления нового материала с эталонами ответа**

1. Что такое алгоритм?

Алгоритм – система точных и понятных предписаний (команд, инструкций, директив) о содержании и последовательности выполнения конечного числа действий, необходимых для решения любой задачи данного типа. Как всякий объект, алгоритм имеет название (имя). Также алгоритм имеет начало и конец.

2. Назовите свойства алгоритма.

Алгоритм обладает следующими свойствами.

1. Дискретность (от лат. discretus – разделенный, прерывистый) указывает, что любой алгоритм должен состоять из конкретных действий, следующих в определенном порядке.

2. Детерминированность (от лат. determinate – определенность, точность) указывает, что любое действие алгоритма должно быть строго и недвусмысленно определено в каждом случае.

3. Конечность определяет, что каждое действие в отдельности и алгоритм в целом должны иметь возможность завершения.

4. Результативность требует, чтобы в алгоритме не было ошибок, т.е. при точном исполнении всех команд процесс решения задачи должен прекратиться за конечное число шагов и при этом должен быть получен ответ.

5. Массовость заключается в возможности применения алгоритма к целому классу однотипных задач, различающихся конкретными значениями исходных данных (разработка в общем виде).

3. Какие вы знаете способы описания алгоритма?

Способы описания алгоритмов:

• словесный (на естественном языке);

• графический (с помощью стандартных графических объектов (геометрических фигур) – блок-схемы);

• программный (с помощью языков программирования).

**Список использованной литературы**

1. Цветкова М. С., Великович Л. С. Информатика и ИКТ: учебник для студ. учреждений сред.проф. образования. — М., 2014.
2. Омельченко В. П., Демидова А. А.. Информатика. Учебник для медицинских училищ и колледжей. - М.: ГЭОТАР-Медиа, 2015 г.
3. Угринович Н.Д. Информатика и информационные технологии. Учебник 10–11 кл. – М.: БИНОМ. Лаборатория знаний, 2016.
4. Семакин И.Г., Хеннер Е.К. Информатика и ИКТ. Базовый курс. Учебник для 10-11 кл. – М.: Изд-во «БИНОМ. Лаборатория знаний», 2015
5. Макарова Н. В., Николайчук Г. С., Титова Ю. Ф., Информатика и ИКТ, 10 класс, Базовый уровень – СПб.: Питер, 2017
6. Макарова Н. В., Николайчук Г. С., Титова Ю. Ф., Информатика и ИКТ, 11 класс, Базовый уровень – СПб.: Питер, 2017