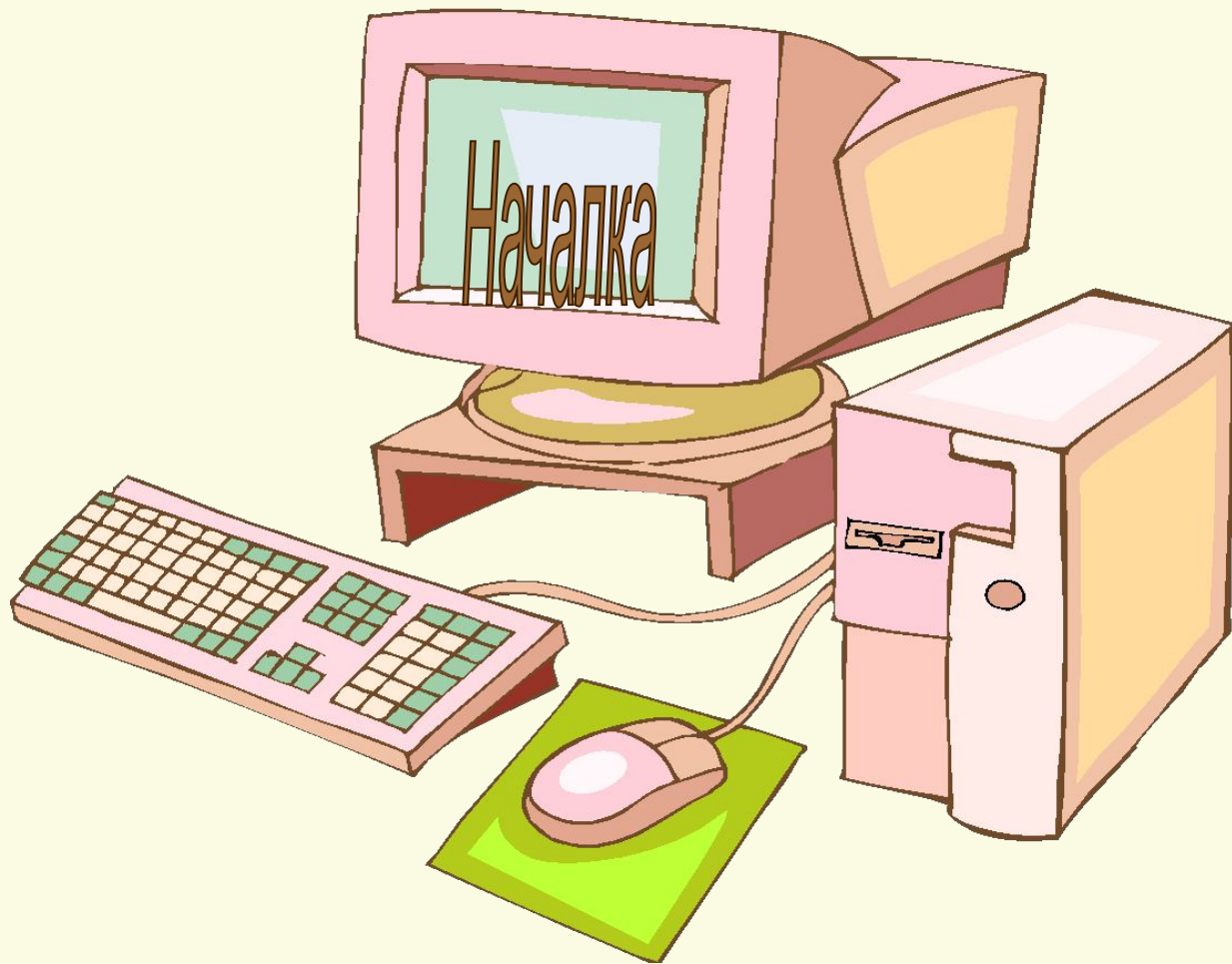


# Алгоритмы и типы алгоритмов



***Понятие алгоритма***

---

***Линейные алгоритмы***

***Разветвляющиеся алгоритмы***

***Словарь понятий***

***Задание***

# ***Понятие алгоритма***

**Алгоритм** - это строгоупорядоченная последовательность действий, приводящая к решению задачи.

Название "**алгоритм**" произошло от латинской формы среднеазиатского математика аль-Хорезми - Algorithmi.

**Алгоритм** - одно из основных понятий информатики и математики.

**Исполнитель алгоритма** - это некоторая техническая, биологическая или биотехническая система, способная выполнить действия, предписываемые алгоритмом.

**Исполнителя алгоритма** характеризуют: среда, элементарные действия, система команд, отказы.

# ***Понятие алгоритма***

**Среда** (или обстановка) — это "место обитания" исполнителя.

---

Каждый исполнитель может выполнять команды только некоторого строго заданного списка - **системы команд** исполнителя.

Для каждой команды должны быть заданы **условия применимости** (в каких состояниях среды может быть выполнена команда) и описаны **результаты выполнения команды**. Например, команда Робота "вверх" может быть выполнена, если выше Робота нет стены. Её результат - смещение Робота на одну клетку вверх.

**Отказы** исполнителя возникают, если команда вызывается при недопустимом для нее состоянии среды.

**Исполнитель** ничего не знает о цели алгоритма. Он выполняет все полученные команды.

**Компьютер** - универсальный исполнитель алгоритмов.

# ***Свойства алгоритма***

**Понятность** для исполнителя - исполнитель алгоритма должен знать, как его выполнять.

**Дискретность** (прерывность, отдельность) - алгоритм должен представлять процесс решения задачи как последовательное выполнение простых ( или ранее определенных ) шагов ( этапов ).

**Определенность** - каждое правило алгоритма должно быть четким, однозначным и не оставлять места для произвола.

**Результативность** (конечность) состоит в том, что алгоритм должен приводить к решению задачи за конечное число шагов.

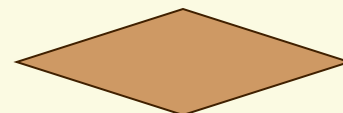
**Массовость.** Алгоритм решения задачи разрабатывается в общем виде. Он должен быть применим для некоторого класса задач, различающихся лишь исходными данными. Исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.

# Способы записи алгоритмов

Словесный.

1. Перевезти козу
2. Возвратиться самому
3. Перевезти волка
4. Возвратиться вместе с козой
5. Перевезти капусту
6. Возвратиться самому
7. Перевезти козу

Графический – в виде блок-схем



На алгоритмическом языке  
(языке программирования).

Алг

нач

кон

# ***Виды алгоритмов***

**Линейный алгоритм** - это такой алгоритм, действия которого следуют друг за другом.

**Разветвляющийся алгоритм** - это такой алгоритм, действия которого выполняются в зависимости от условия (вопрос на который можно ответить "да" или "нет").

**Циклический алгоритм** - это такой алгоритм, действия которого повторяются.



Алгоритм решения задачи называется *линейным*,  
если исполнитель все команды алгоритма  
исполняет одну за другой в порядке их записи.

Задача о волке, козе и  
капусте.

Человеку                      нужно  
переправить                      на  
противоположный              берег  
волка, козу и капусту.

В                      лодку человек  
может поместиться          либо с  
козой, либо с волком, либо с  
капустой.

На берегу нельзя оставить  
волка с козой или козу с  
капустой.

## Алг Переправа

**нач**

1. Перевезти козу
2. Возвратиться самому
3. Перевезти волка
4. Возвратиться вместе с козой
5. Перевезти капусту
6. Возвратиться самому
7. Перевезти козу

**кОН**



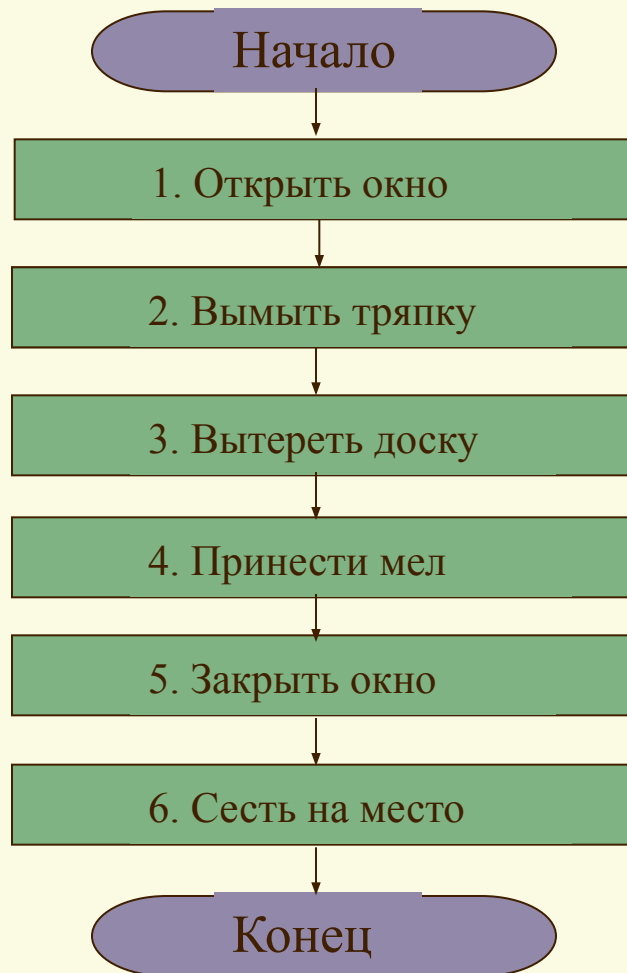
## Блок-схема для линейного алгоритма

### Алг Дежурик

**нач**

1. Открыть окно
2. Вымыть тряпку
3. Вытереть доску
4. Принести мел
5. Закрыть окно
6. Сесть на место

**КОН**



Алгоритм называется *разветвляющимся*, если после проверки условия в разных ситуациях исполняются разные наборы команд.

Составим алгоритм правописания приставок на «з», «с».

Например:

Разговор (пишем «з»)

Рассказ ( пишем «с»)

## Алг Правописание

**нач**

1. Выделяем корень слова.

2. Если корень начинается со звонкой перейти к пункту 3, в противном случае к пункту 4

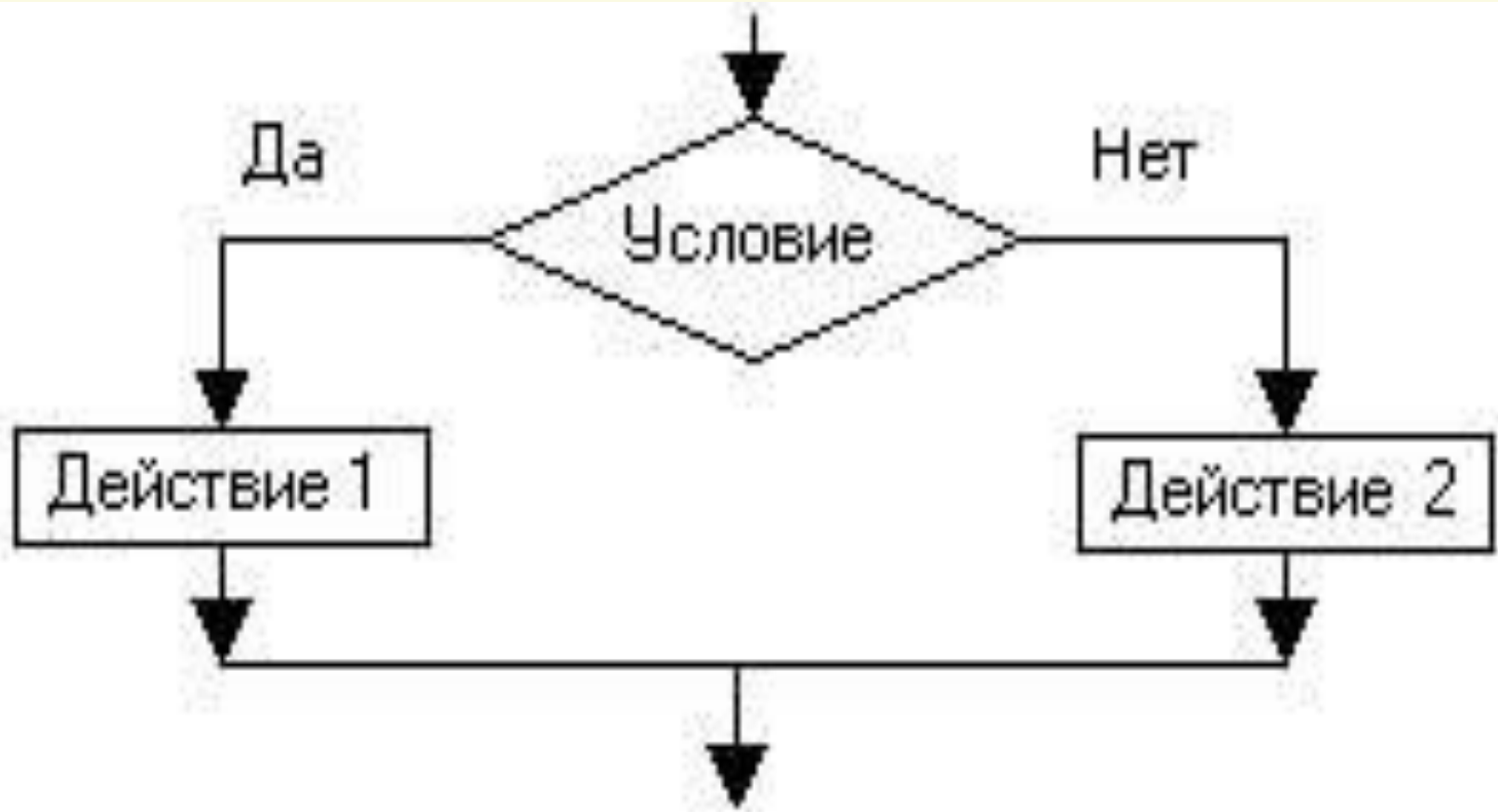
3. В приставке пишем «з». Перейти к пункту 5

4. В приставке пишем «с».

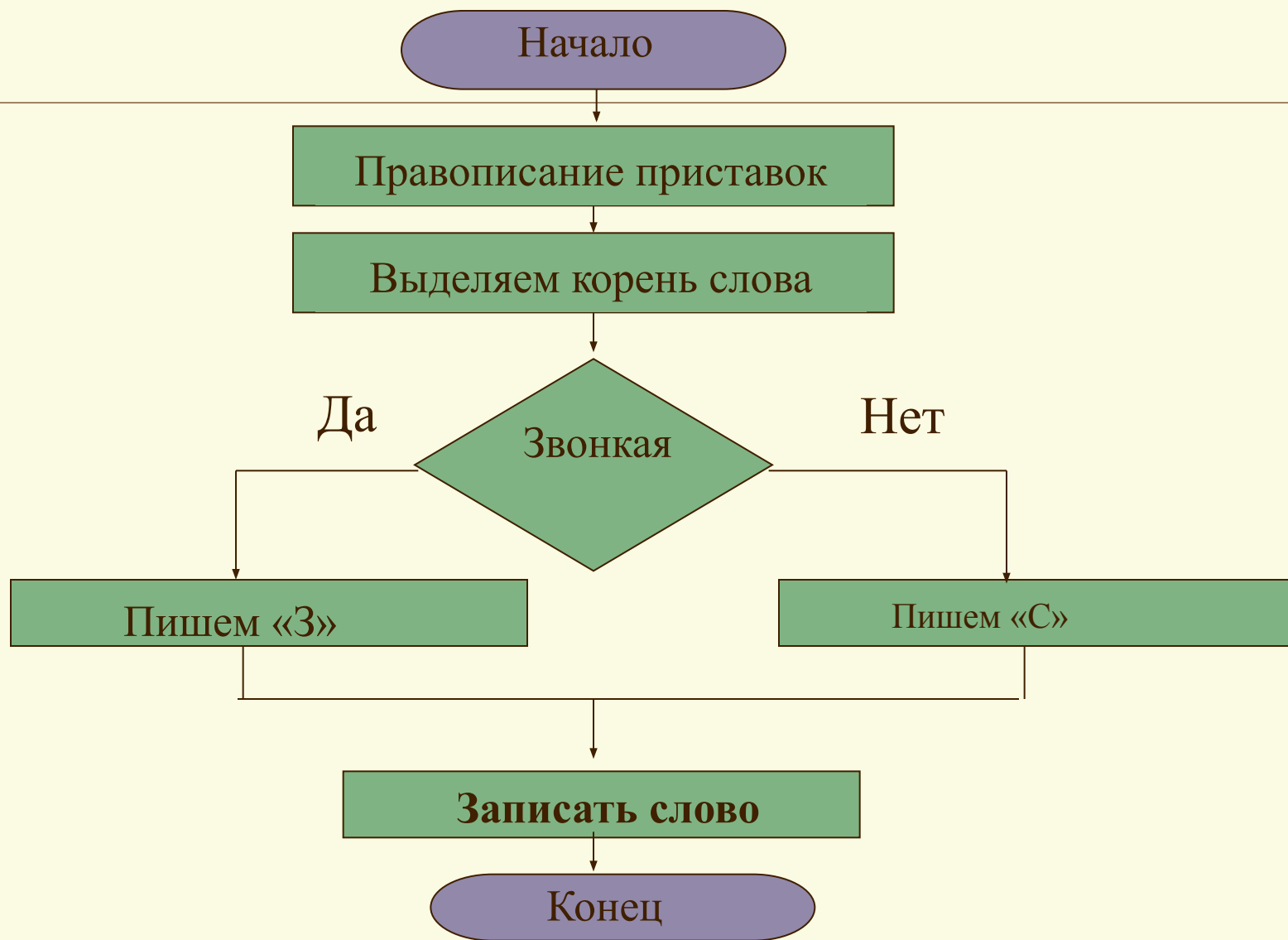
5. Записать слово

**кон**

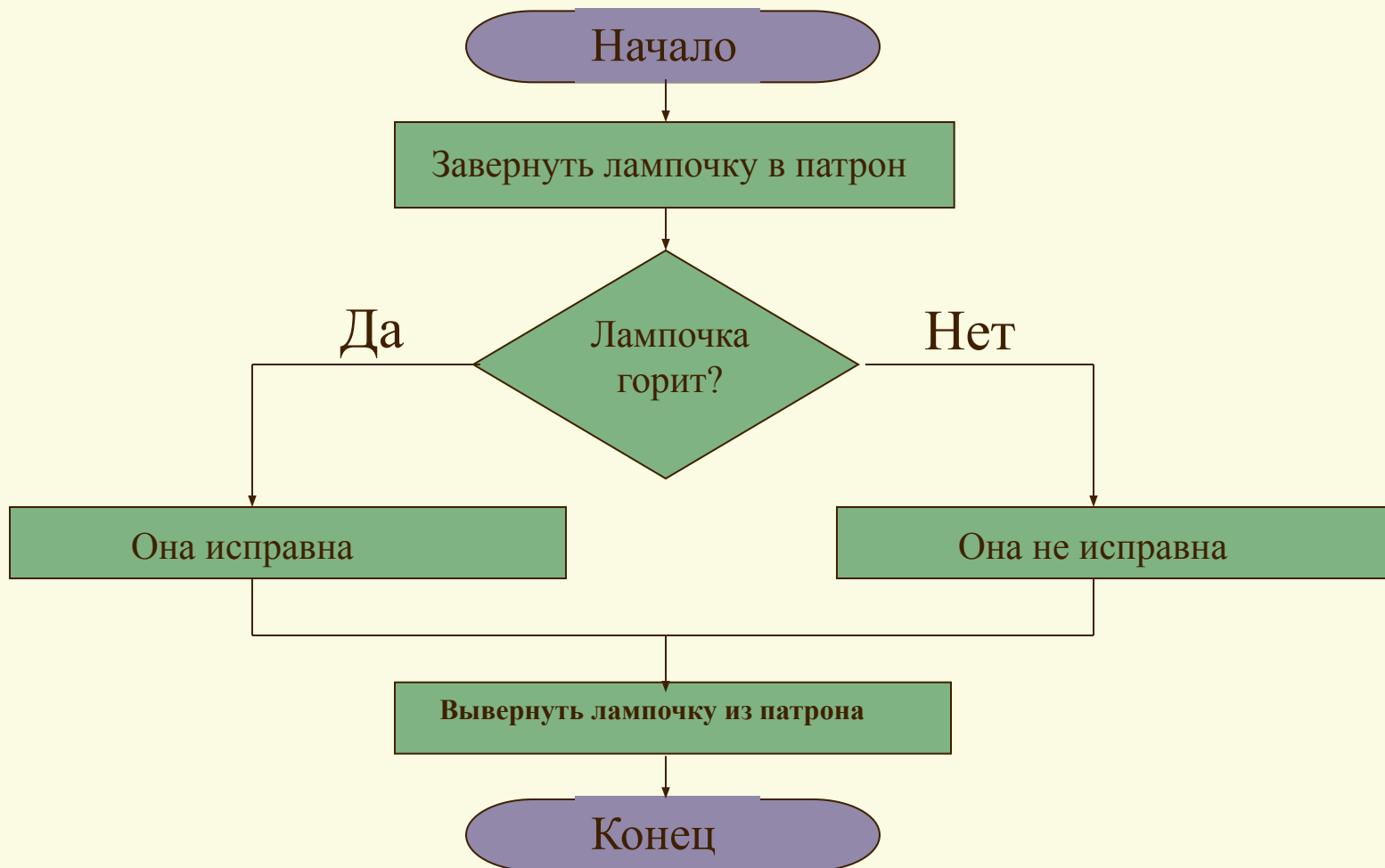
Алгоритм называется *разветвляющимся*, если после проверки условия в разных ситуациях исполняются разные наборы команд.



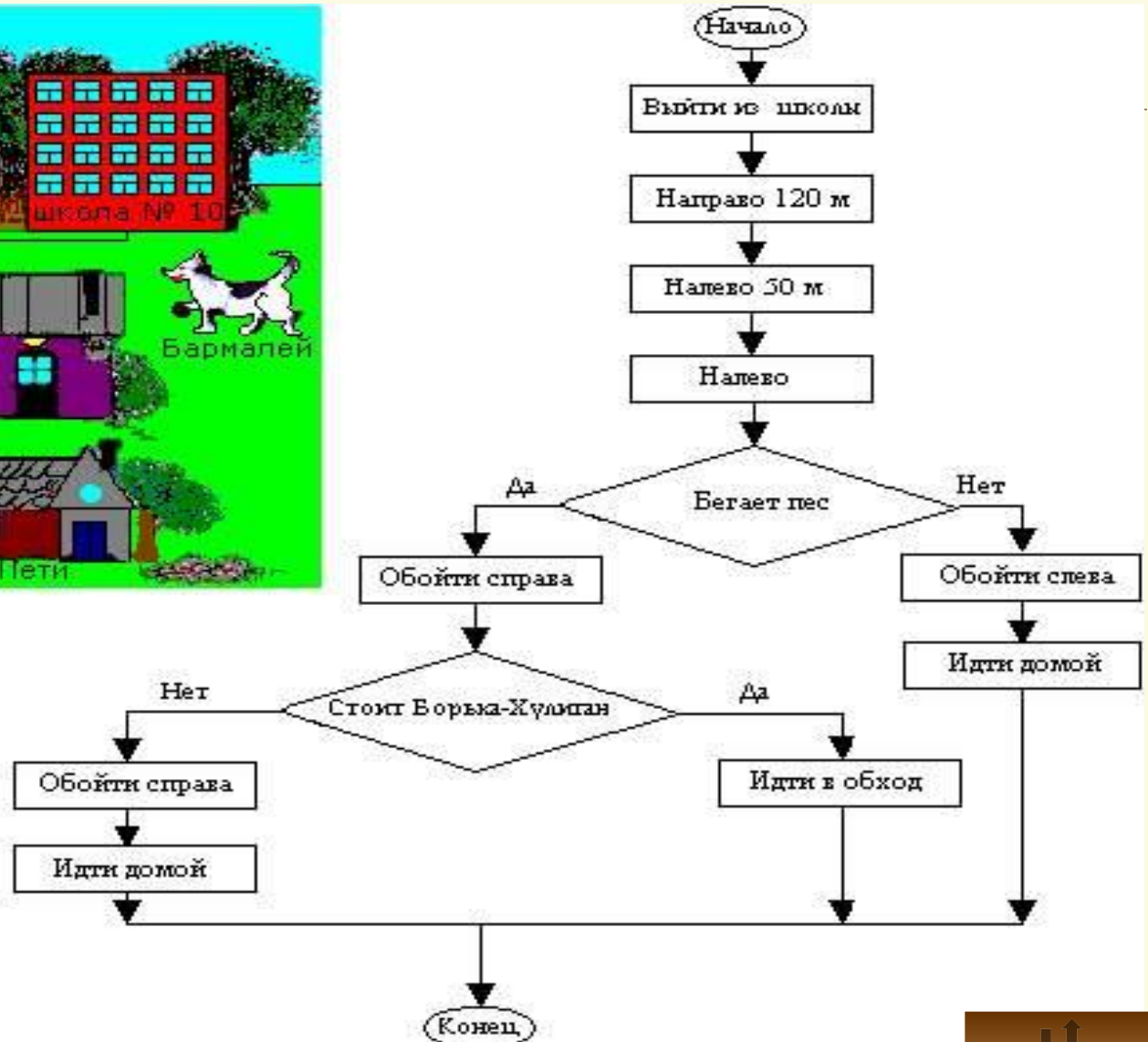
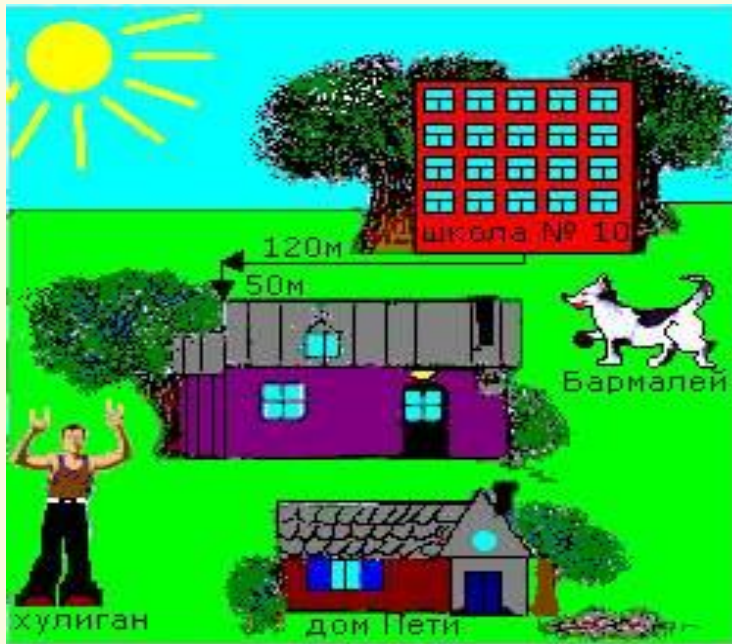
Построим блок-схему для разветвляющегося алгоритма



# Построим блок-схему для разветвляющегося алгоритма



# Построим блок-схему для разветвляющегося алгоритма



# Словарь понятий

---

**Алгоритм** - последовательность указаний (команд) исполнителю, выполнив которую он достигает поставленной цели или решает определенную задачу.

**Команда** - указание исполнителю выполнить отдельное действие.

**Система команд исполнителя** - множество команд, которые понимает и может выполнить исполнитель.

**Графический способ** - **блок-схема** - запись алгоритма в виде последовательности специальных графических блоков-обозначений.

# Словарь понятий

Каждый **блок** соответствует выполнению одного или нескольких действий.

---


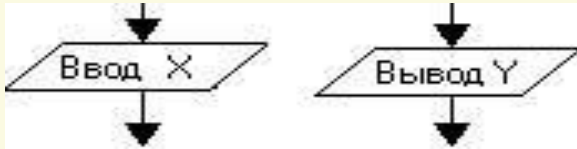
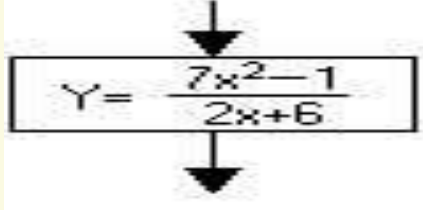
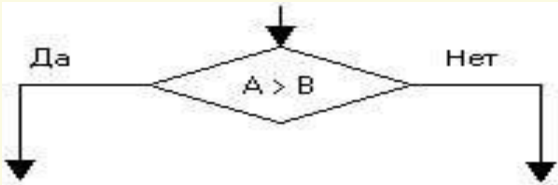
В **блок-схеме** каждому типу действий (вводу исходных данных, вычислению значений выражений, проверке условий, управлению повторением действий, окончанию обработки и т.п.) соответствует определенная геометрическая фигура.

**Словесное описание алгоритма** не имеет широкого распространения:

- строго не формализуемы;
- страдают многословностью записей;
- допускают неоднозначность толкования отдельных предписаний.



# Словарь понятий

Название символ	Обозначение и пример заполнения	Пояснение
Пуск — остановка		Начало, конец алгоритма
Ввод — вывод		Ввода данных в память или вывод на экран монитора или принтер.
Процесс (действие)		Вычислительное действие
Решение (условие)		Проверка условий

# Теперь попробуйте свои силы

(Задание выполняется в Word)

---

1. Придумайте, составьте и запишите разными способами линейный алгоритм

2. Известен такой способ проверки арбуза на спелость. Если бросить арбуз в воду, и он всплывет, значит арбуз - спелый, иначе - не спелый.

Запишите этот алгоритм всеми тремя способами.

# Работа с исполнителями

**1. Загрузите исполнитель Бук**, который считает буквы в тексте :  
`\azinf\computer\buka\buka.htm`.

---

**2. Проведите работу** по следующему плану:

- выберите текст.
- выберите букву.
- подсчитайте, сколько раз буква встречается в тексте.
- введите ответ.
- проверьте решение.

**3. Ответьте** на следующие вопросы:

Как называют план, по которому Бука выполняет вычисления?

Кто быстрее выполняет вычисления, человек или Бука?

Кто считает без ошибок, человек или Бука?

Если все же компьютер работает с ошибками, то в чем может быть причина?

**4. Бука может выводить** на экран следующие сообщения:

*Не понимаю! Не могу! Ваш ответ — неверный! Правильно!*

Объясните, по каким причинам Бука выводит то или иное сообщение.

# Работа с исполнителями

1. Запустите Исполнитель Конюх (horse\horse\horse).
2. Познакомьтесь с инструкцией по работе.
3. Выполните задание.
4. Последовательно откройте папки **Зимние вечера – WINTER**.
5. Исследуйте Исполнители, приведенные в этой папке.
6. Сравните системы команд исследованных Исполнителей и оформите результаты в виде таблицы в редакторе Word:

Исполнитель	Система команд

# Работа с ребусами

---

1. Запустите программу Rebus (rebus\rebus\rebus)
2. В редакторе Word подготовьте справочные материалы для себя:
  - создайте таблицу из двух столбцов и нескольких строк;
  - в первый столбец из папки pic вставляйте картинки, а во втором столбце введите ответы.

