



# Этапы решения задач на ПК

Способы представления  
алгоритмов

Классификация ЯП

# Этапы решения задач на ПК

- Процесс решения задач на ПК – это совместная деятельность человека и ЭВМ

этапы, связанные с творческой деятельностью – постановкой, алгоритмизацией, программированием задач и анализом результатов,

этапы обработки информации в соответствии с разработанным алгоритмом

# Этапы решения задач на ПК



Если для решения задачи хотят прибегнуть к помощи ПК, а готовой программы нет, то потребуются освоить весь процесс решения задачи на ПК, создав свою программу.

# I Постановка задачи

- Чтобы выбрать **метод решения**, нужно четко представлять, чем мы располагаем – каковы исходные данные и ограничения на них.
- правильная **постановка задачи** - ответ на вопросы: что дано, что требуется определить, какая связь между данными и результатом.

# II Математическая модель

## Шаги:

- 1) Математическая постановка задачи:
  - перечень исходных данных (что дано);
  - перечень результатов (что требуется найти);
  - ограничения на исходные данные.
- 2) Правила и законы, необходимые для получения результатов.
- 3) Метод решения – оптимальное использование имеющейся в распоряжении модели.

## || Например:

Задача. Решить квадратное уравнение  
 $ax^2+bx+c=0$

Итак, постановка этой задачи выглядит следующим образом:

Дано:  $a, b, c$  – коэффициенты уравнения.

Найти:  $x_1, x_2$  – корни уравнения.

Связь: При  $a \neq 0$  и  $D=b^2-4ac \geq 0$

$x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$ , иначе действительных корней нет.

# III Алгоритм

- В основу программы для ПК кладется алгоритм решения данной задачи.
- Определение?
- Исполнитель алгоритма?
- Какие бывают исполнители?
- Что такое СКИ?
- Что такое класс задач?

### III Свойства алгоритмов

- **Дискретность.** Структура алгоритма является прерывистой.
- **Понятность.** Ориентация на исполнителя и его СКИ.
- **Детерминированность (однозначность).**
- **Конечность.**
- **Результативность.**
- **Массовость.** Обеспечение решение всего класса задач данного типа.





### III Способы записи алгоритмов

1. словесная форма записи;
2. на языке блок-схем;
3. на алгоритмическом языке.

# 1. Словесная форма записи

Алгоритм  
записан  
словами  
и  
предназн  
ачен для  
человека

*Начало*

1. Налить воды в чайник.
2. Поставить чайник на электроплиту.
3. Включить электроплиту.
4. Подождать, пока вода закипит.
5. Выключить электроплиту.

*Конец*



## III 2. На языке блок-схем

<b>Наименование символа</b>	<b>Функция</b>
	Начало, конец программы
	Условие
	Ввод или вывод данных

1:1

## III 2. На языке блок-схем

Наименование символа	Функция
	Выполнение операции или группы операций
	Использование процедуры (вспомогательного алгоритма).
	Указание связи, разрыв алгоритма

Блок – схема	Псевдокоды	Pascal
<pre> graph TD     Start([Начало]) --&gt; Input[/Ввод n/]     Input --&gt; S0[S:=0]     S0 --&gt; I1[i:=1]     I1 --&gt; Loop{ i &lt;= n }     Loop -- да --&gt; Sum[S:=S+i]     Sum --&gt; Inc[i:=i+2]     Inc --&gt; Loop     Loop -- нет --&gt; Output[/Вывод S/]     Output --&gt; End([Конец]) </pre>	<pre> <b>алг</b> сумма нечетных чисел <b>нач</b>   <b>ввод</b> n   S:=0   i:=1   <b>нц пока</b> i&lt;=n     S:=S+i     i:=i+2   <b>кц</b>   <b>вывод</b> S <b>кон</b> </pre>	<pre> <b>program</b> summa_nech; <b>var</b> i, n, S: <b>integer</b>; <b>begin</b>   <b>readln</b> (n);   S:=0;   i:=1;   <b>while</b> i&lt;=n <b>do</b>     <b>begin</b>       S:=S+i;       i:=i+2;     <b>end</b>;   <b>writeln</b> (S) <b>end.</b> </pre>

## IV Программа

- Для автоматизации процесса решения задачи алгоритм должен быть записан на компьютере.
- Команды, предназначенные для ЭВМ необходимо записывать в понятной ей форме.
- С этой целью применяются языки программирования – искусственные языки, алфавит, словарный запас и структура которых удобны человеку и понятны компьютеру.

# IV Классификация языков программирования (ЯП)

**ЯП** – это фиксированная система обозначений и правил для описания алгоритмов и структур данных.

## Низкого уровня

Обеспечивают поддержку простейших операций, обычно на уровне системных устройств – процессора, памяти и т.д.

Например: Assembler

## Высокого уровня

Программы на таких языках более объемны и менее быстродейственны.

Например: Pascal, Си++, Basic, Lisp, Java и др.



## IV Программирование

– это процесс составления алгоритма решения поставленной задачи на языке программирования, ее тестирование (проверка соответствия поставленной задаче) и отладка (исправление ошибок).

## Программа

– это алгоритм (набор команд), записанный на языке программирования.



# ✓ Анализ результатов выполнения

Состоит в тестировании и отладке программы.

**Отладка** – это процесс выявления и исправления ошибок.

**Виды ошибок**, возникающих в процессе разработки программы:

- ✓ синтаксические;
- ✓ логические;
- ✓ динамические.

# Решение задач



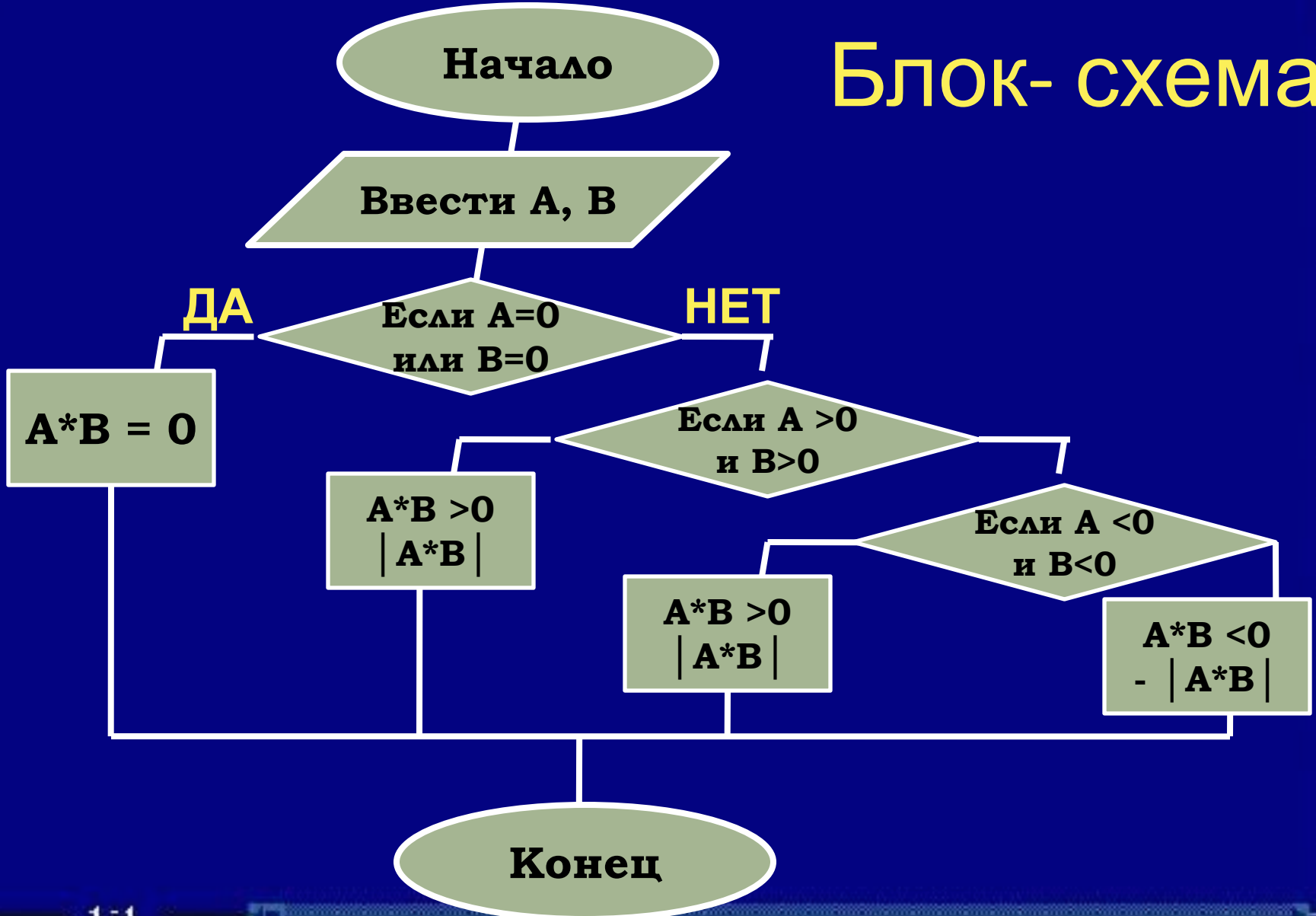
Решить задачи поэтапно (первые 3 этапа), составив алгоритм на языке блок-схем (поясните действия в каждой из фигур).

Найти произведение двух чисел  $A$  и  $B$ , определить знак произведения.

# Решение задач

- *Словесная форма записи:*
- Если ( $A = 0$  или  $B = 0$ ), то произведение = 0.  
Если ( $A > 0$  и  $B > 0$ ) или ( $A < 0$  и  $B < 0$ ), т.е. числа  $A$  и  $B$  имеют одинаковые знаки, то произведение положительное и определяется произведением модулей исходных чисел.
- В противном случае произведение равно произведению модулей со знаком минус.

# Блок-схема



# Домашнее задание

- Готовиться к проверочной
- Решить задачу (Первые 3 этапа из карточки)