

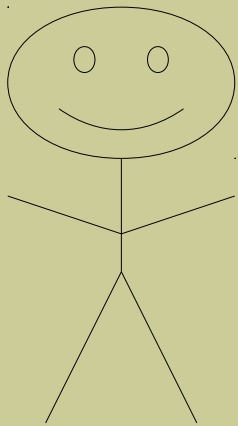
Лекция 1

Содержание

- Цели и задачи курса.
- Основные понятия предметной области Информатика.
- Краткий обзор курса.
- Требования к студентам при освоении предмета.
- Порядок работы и условия сдачи экзамена.
- Алгоритмы (введение).

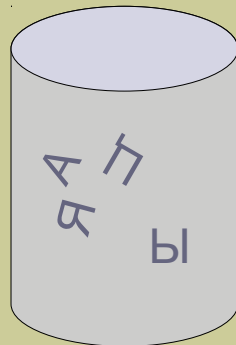
Основные понятия информатики

ИНФОРМАЦИЯ



отправитель

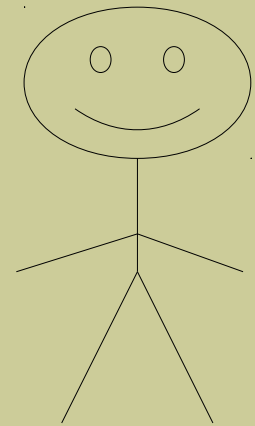
СООБЩЕНИЕ



символы, знаки, алфавит

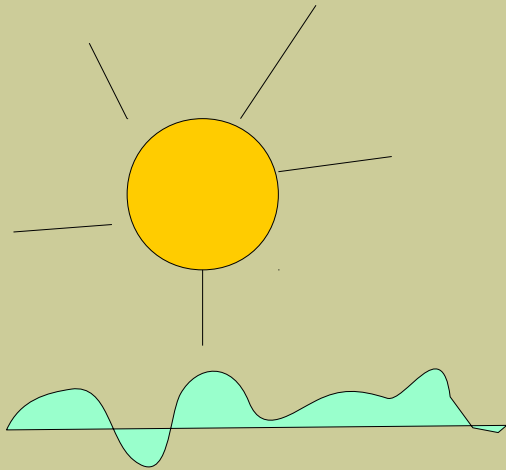
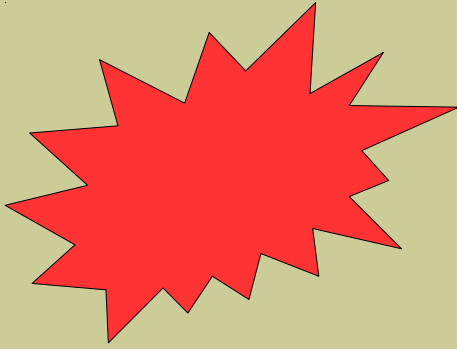


сигналы



получатель

Измерение информации



Количество информации:

$$I = \log_2 1/p$$

Усредненное по состояниям количество информации:

$$S = -\sum p_i \log_2 1/p_i$$

Содержание курса

1. Обзорная лекция, введение в предмет.

2. Понятие алгоритма. Определение алгоритма. Представление алгоритмов: псевдокод, блок-схема, *компьютерная программа*. Базовые алгоритмические конструкции. Итерация и рекурсия. Разбор примеров: алгоритмы последовательного и двоичного поиска, алгоритмы сортировки.

3. Биты и манипулирование ими. Булева алгебра. Абстрактные устройства, представляющие логические операции, - вентили, и их технические реализации. Триггеры и хранение бит. Структура оперативной памяти - ячейки, байты, адресация.

4. Цифровое кодирование информации. Двоичная система счисления. Представление отрицательных целых чисел - дополнительный код, код со смещением. Представление вещественных чисел с фиксированной и плавающей точкой.

5. Введение в архитектуру компьютера. Архитектура фон Неймана. Принцип автоматической обработки информации. Функциональные узлы компьютера. Машинные команды. Машинный цикл выполнения программы. Взаимодействие компьютера с периферийными устройствами. Обзор внешних интерфейсов.

6. Обзор современных архитектур ВС. Параллельная обработка данных. Конвейеры. Мультипроцессоры. Графические ускорители.

7. Операционные системы. Основные компоненты операционной системы и их назначение. Типы операционных систем. Управление процессами. Взаимодействие между процессами. Этапы загрузки операционной системы. Линейки операционных систем MS Windows и GNU/Linux.

8. Компьютерные сети и распределенные вычислительные системы. Сетевое оборудование и сетевые протоколы. Стек протоколов TCP/IP. IP-адресация. Двухуровневая архитектура клиент-сервер. Сетевые службы DHCP, DNS, POP3, SMTP. Компьютерные кластеры. Распределенные вычисления.

9. Структуры баз данных. СУБД. Реляционные базы данных. Иерархические базы данных. XML – технология.

Рекомендуемая литература

- 1) Информатика и вычислительная техника. [Текст] : учеб. пособие / Дж. Брукшир. - 7-е изд. стер. - Москва : ПИТЕР, 2004. - 619 с.
- 2) Информатика. [Текст] : учебник / Б.В. Соболев и др. - 3-е издание, дополненное и переработанное. - Ростов-на-Дону: Феникс, 2005. -446 с.

Электронные ресурсы

- 1) Информатика [Электронный ресурс]: учебное пособие/ С.В. Тимченко [и др.].Электрон. текстовые данные.— Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2011.— 160 с.— Режим доступа: <http://www.iprbookshop.ru/13935>.— ЭБС «IPRbooks», по паролю.
- 2)Шапоров С. [Информатика. Теоретический курс и практические занятия](#) Санкт-Петербург: БХВ-Петербург , 480 с.— Режим доступа: <http://ibooks.ru/reading.php?productid=18483>— ЭБС«ibooks», по паролю.
- 3)**ИНТУИТ**, например: <http://www.intuit.ru/studies/courses/103/103/lecture/27963>

Алгоритмы (часть 1)

Определение алгоритма. Представление алгоритма: псевдокод, блок-схема. Базовые алгоритмические конструкции.

Определение алгоритма.

Алгоритм – это *упорядоченный* набор *конечного* числа строго *определенных выполнимых* шагов для решения *задачи определенного типа*.

Конечность.

Задача. Даны два отрезка разной длины a и b . Построить c - наибольший из отрезков, укладывающихся целое число раз в данных отрезках.

Алгоритм(?). Пусть $|a| > |b|$.

Шаг 1. Отложим отрезок b на отрезке a наибольшее количество раз.

Шаг 2. Если b точно отложился на a целое число раз, то выполнение алгоритма прекращается, задача решена, искомый отрезок – это b .

Шаг 3. Принять в качестве отрезка b остаток отрезка a , куда не помещался отрезок b , а в качестве отрезка a отрезок b и перейти к Шагу 1.

Является ли данный набор шагов алгоритмом?

Если существует некий отрезок, пусть очень малой длины, укладывающийся целое число раз в отрезках a и b , то можно среди таких отрезков найти и наибольший, используя приведенный выше набор шагов. Но может не существовать такого отрезка, в этом случае говорят, что отрезки несоизмеримы (отношение их длин выражается бесконечной непериодической десятичной дробью). Тогда последовательность приведенных шагов становится бесконечной. Таким образом, в общем случае вышеприведенный набор шагов не является алгоритмом. Для соизмеримых отрезков этот набор является алгоритмом (геометрический аналог алгоритма Евклида).

Вопросы:

Является ли метод деления столбиком алгоритмом нахождения частного?
(бесконечные периодические дроби)

Является ли метод деления столбиком алгоритмом нахождения частного с заданной точностью?

То же для вавилонского метода оценки квадратного корня x из целого числа y :

$$x := (x + y/x)/2.$$

Определенность.

Задача. Найти длину гипотенузы прямоугольного треугольника, зная длину его катетов.

Шаг 1. Возвести в квадрат длину 1-го катета

Шаг 2. Возвести в квадрат длину 2-го катета

Шаг 3. Сложить полученные числа.

Шаг 4. Извлечь квадратный корень из полученного числа.

Является ли этот набор шагов алгоритмом?

Не определен шаг №4. При извлечении корня получается два числа и только одно из них положительное – арифметический квадратный корень.

Необходимо детализировать этот шаг с тем, чтобы результат его выполнения был однозначным.

Выполнимость.

...

Шаг N . Умножить полученное число на сумму $x+y+z$, где (x,y,z) из N^3 является решением уравнения $x^4+y^4=z^4$ с наименьшим значением x .

...

Является ли этот набор шагов алгоритмом?

Во-первых, шаг N неоднозначен, при одном x может быть несколько решений с разными суммами. Во-вторых, и это главное, шаг N содержит невыполнимые действия. Дело в том, что в соответствии с уже доказанной *теоремой Ферма* таких решений вообще не существует.

[алгоритмически неразрешимые задачи; 10-я проблема Гильберта; вычислимость [А. Тьюринг]]

Представление алгоритма.

Виды представлений.

Псевдокод, *pidgin Pascal*, C и т.п. **Блок-схема**. [диаграмма активности в UML] Программы на языке высокого уровня. **Программа** – последовательность нулей и единиц.

Базовые алгоритмические конструкции.

Присвоение.

<имя переменной>:=<выражение>

Следование.

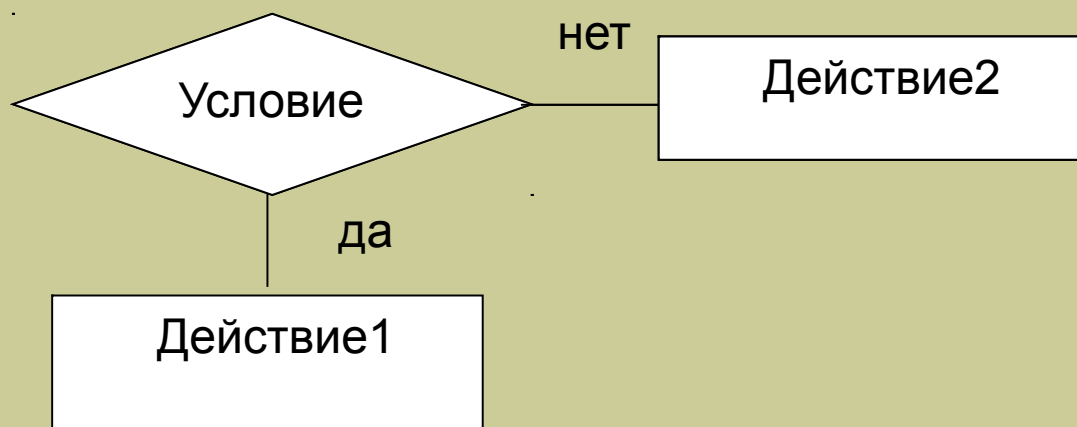
<Действие 1>

<Действие 2>



Ветвление.

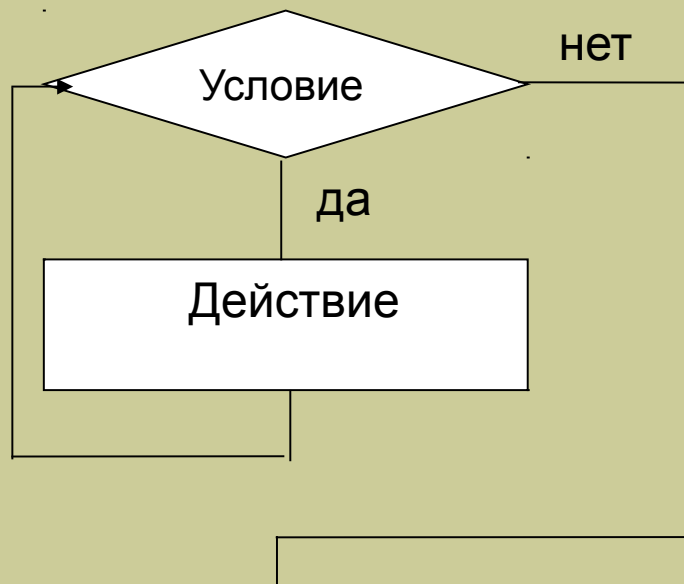
Если <условие>
то <действие 1>
иначе <действие 2>
Конец-если



Цикл-пока.

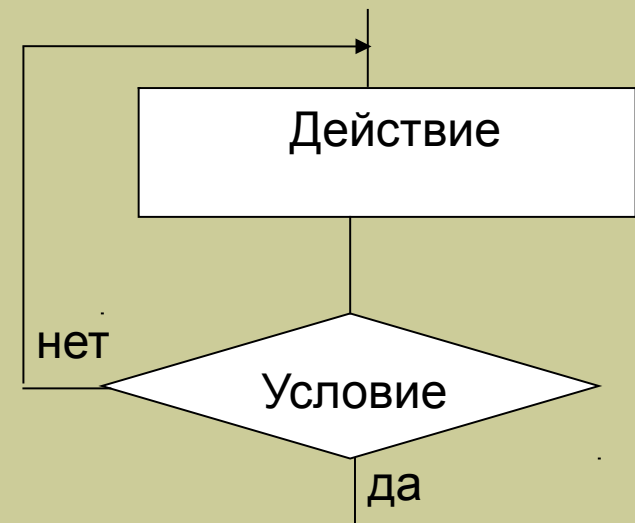
Цикл-пока <условие>

<действие>
Конец-цикл



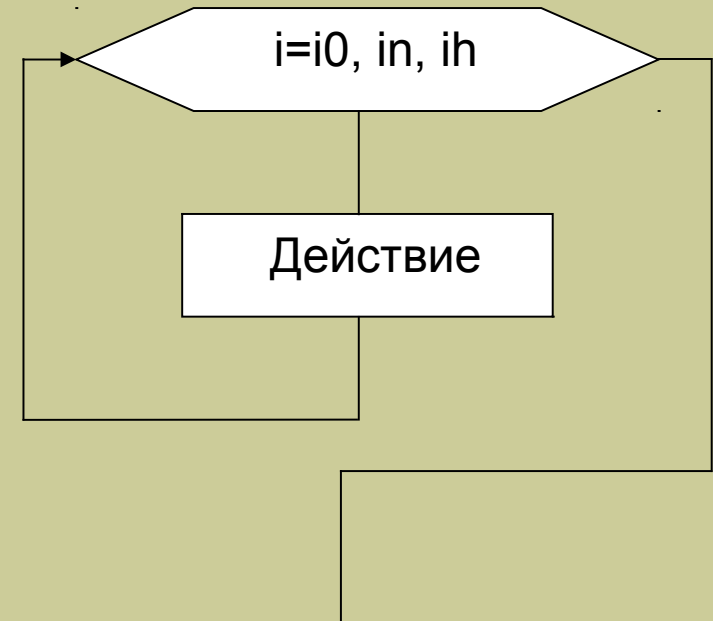
Цикл-до (repeat).

Выполнять
 <Действие>
До <условие>



Счетный цикл

Для <индекс>=<i0, in, ih>
 <Действие>
Конец-цикл



Выбор.

Выбор <код>

<код 1>:

<действие 1>

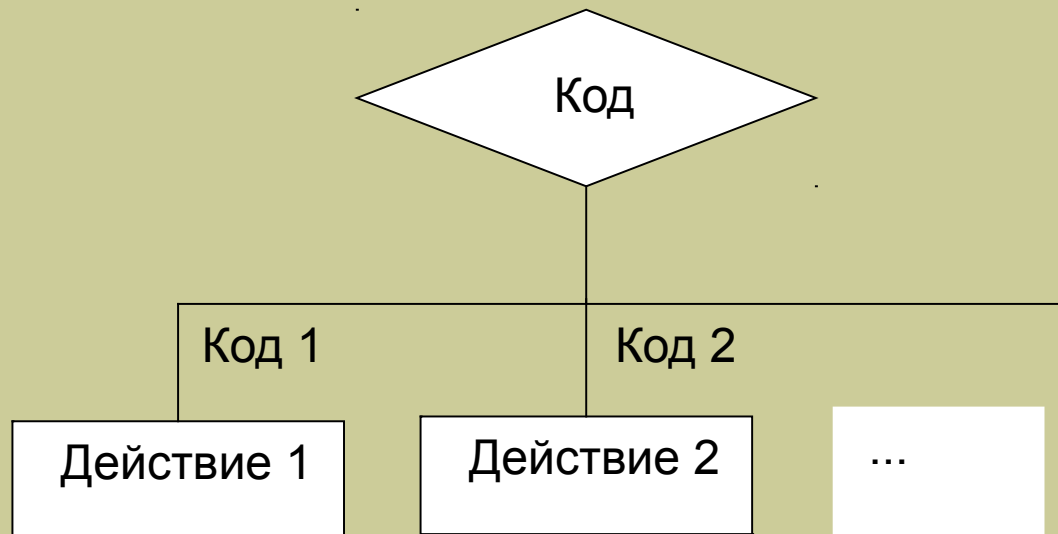
<код 2>:

<действие 2>

...

Конец-выбор

...



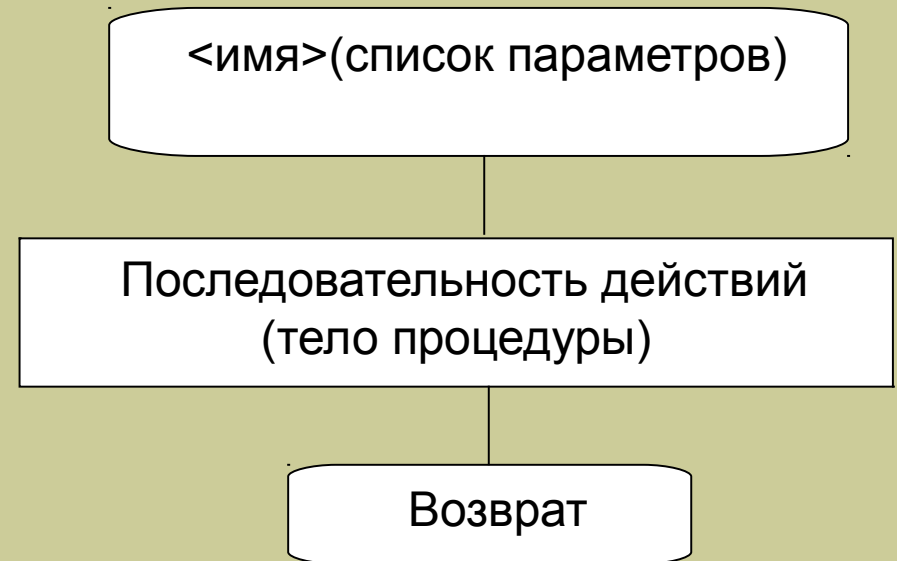
...

Процедура.

Описание процедуры:

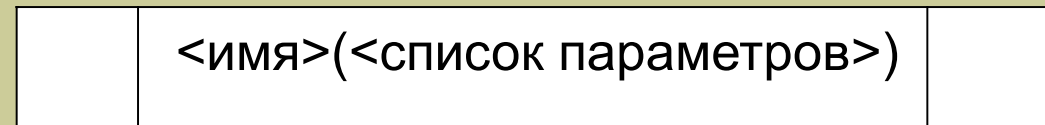
Процедура <имя>(<список
параметров>)
 <действие 1>
 <действие 2>

 [<имя>:=<выражение>]
Возврат



Вызов процедуры:

<имя>(<список фактических параметров>)



Упражнение

Представить алгоритм Евклида в виде псевдокода и блок-схемы.

Шаг1. Разделим m на n и пусть r – остаток.

Шаг2. Если $r=0$, то выполнение алгоритма прекращается; n – искомое значение.

Шаг3. Присвоить $m:=n$, $n:=r$ и вернуться к шагу1.

Программа Евклид

Ввод m, n

$r:=m\%n$

Цикл-пока r не равно 0

$m:=n$

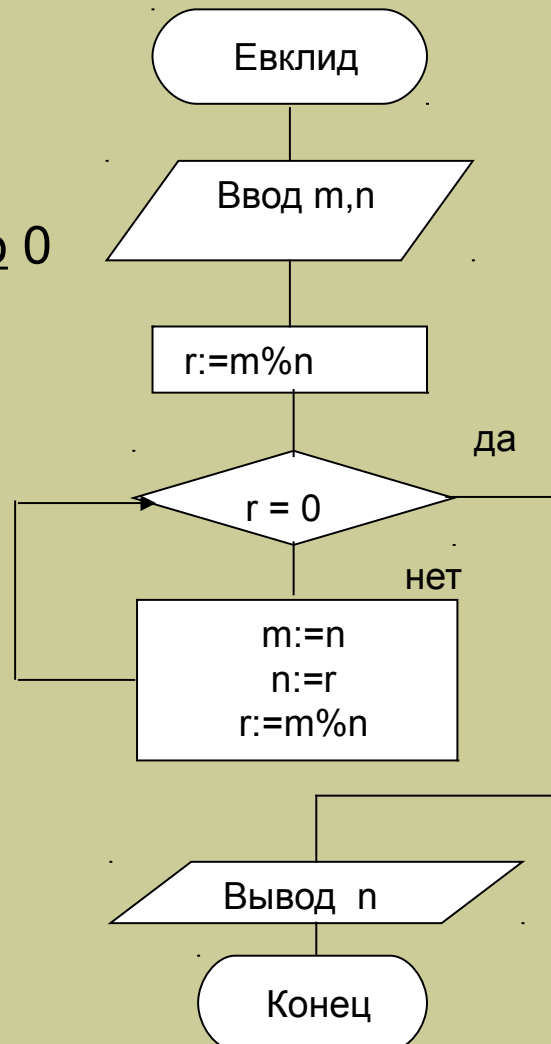
$n:=r$

$r:=m\%n$

Конец-цикл

Вывод n

Конец



Задание

- Будет ли алгоритм работать, если $m < n$?
- Всегда ли за конечное число шагов переменная r получит значение 0?
- Замените операцию деления по модулю - % (получение остатка) вычитанием.
- Оформите в виде процедуры совокупность действий, направленных на нахождение остатка.