

Дисциплина: информатика (практические работы) с делением на подгруппы

Преподаватель:

Дариенко Татьяна Викторовна

СМ-12

Задание для всей группы.

Практическая работа №4

Тема: Программный принцип работы компьютера. Примеры компьютерных моделей различных процессов. Проведение исследования в социально-экономической сфере на основе использования готовой компьютерной модели.

Цель работы: изучить программный принцип работы компьютера, изучить способы представления алгоритмов в виде блок – схем, изучить основные правила составления алгоритмов, рассмотреть примеры компьютерных моделей различных процессов.

Краткие теоретические сведения Программный принцип работы компьютера

Алгоритм — точное и понятное предписание исполнителю совершить последовательность действий, направленных на решение поставленной задачи.

Исполнитель алгоритма — это некоторая абстрактная или реальная (техническая, биологическая или биотехническая) система, способная выполнить действия, предписываемые алгоритмом.

Основные свойства алгоритмов следующие:

Понятность для исполнителя — т.е. исполнитель алгоритма должен знать, как его выполнять.

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

Определенность — т.е. каждое правило алгоритма должно быть четким, однозначным и не оставлять места для произвола. Благодаря этому свойству выполнение алгоритма носит механический характер и не требует никаких дополнительных указаний или сведений о решаемой задаче.

Результативность (или конечность) — т.е. алгоритм должен приводить к решению задачи за конечное число шагов.

Массовость — т.е. алгоритм решения задачи разрабатывается в общем виде, т.е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными. При этом исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.

На практике наиболее распространены следующие **формы представления алгоритмов:**

- словесная (записи на естественном языке);
- графическая (изображения из графических символов);
- псевдокоды (полуформализованные описания алгоритмов на условном алгоритмическом языке, включающие в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические обозначения и др.);
- программная (тексты на языках программирования).

Название символа	Обозначение и пример заполнения	Пояснение
Процесс		Вычислительное действие или последовательность действий
Решение		Проверка условий
Модификация		Начало цикла
Предопределенный процесс		Вычисления по подпрограмме, стандартной подпрограмме
Ввод-вывод		Ввод-вывод в общем виде
Пуск-останов		Начало, конец алгоритма, вход и выход в подпрограмму
Документ		Вывод результатов на печать

Блок **"процесс"** применяется для обозначения действия или последовательности действий, изменяющих значение, форму представления или размещения данных. Для улучшения наглядности схемы несколько отдельных блоков обработки можно объединять в один блок. Представление отдельных операций достаточно свободно.

Блок **"решение"** используется для обозначения переходов управления по условию. В каждом блоке "решение" должны быть указаны вопрос, условие или сравнение, которые он определяет.

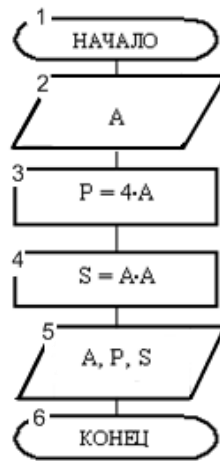
Блок **"модификация"** используется для организации циклических конструкций. (Слово модификация означает видоизменение, преобразование). Внутри блока записывается параметр цикла, для которого указываются его начальное значение, граничное условие и шаг изменения значения параметра для каждого повторения.

Блок **"предопределенный процесс"** используется для указания обращений к вспомогательным алгоритмам, существующим автономно в виде некоторых самостоятельных модулей, и для обращений к библиотечным подпрограммам.

Линейные алгоритмы

Линейный алгоритм — это алгоритм, в котором блоки выполняются последовательно сверху вниз от начала до конца.

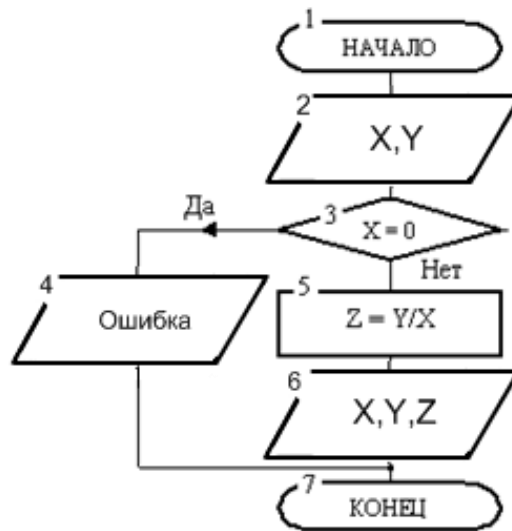
На рисунке приведен пример блок-схемы алгоритма вычисления периметра P и площади S квадрата со стороной длины A .



Разветвляющиеся алгоритмы

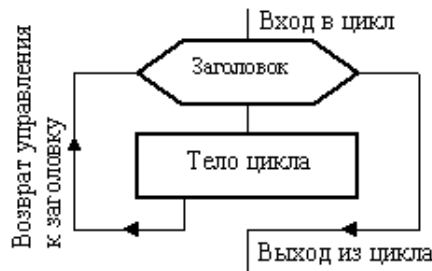
Разветвляющийся алгоритм — это алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий.

На рисунке приведен пример блок-схемы алгоритма вычисления деления двух чисел X, Y.

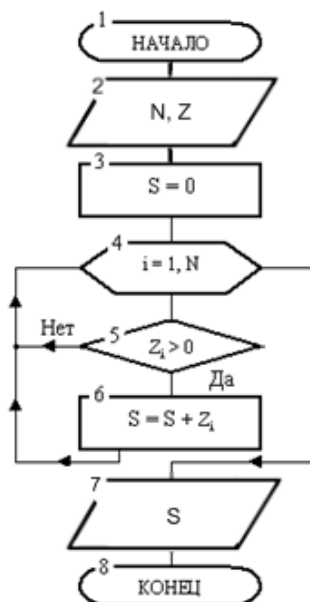


Циклические алгоритмы

Часто при решении задач приходится повторять выполнение операций по одним и тем же зависимостям при различных значениях входящих в них переменных и производить многократный проход по одним и тем же участкам алгоритма. Такие участки называются *циклами*. Алгоритмы, содержащие циклы, называются *циклическими*. Использование циклов существенно сокращает объём алгоритма.



На рисунке приведен пример блок-схемы алгоритма вычисления суммы положительных чисел массива.



Примеры компьютерных моделей различных процессов

Модель – общенаучное понятие, означающее как идеальный, так и физический объект анализа, который отражает существенные признаки изучаемого процесса, явления или объекта,

Важным классом идеальных моделей является математическая модель, в ней изучаемое явление или процесс представлены в виде абстрактных объектов или наиболее общих математических закономерностей, выражающих либо законы природы, либо внутренние свойства самих математических объектов, либо правила логических рассуждений.

Границы между моделями различных типов или классов, а также отнесение модели к какому-то типу или классу чаще всего условны. Наиболее распространенные признаки, по которым классифицируются модели:

- цель использования;
- область знаний;
- фактор времени;
- способ представления.

По целям использования выделяются модели *учебные, опытные, имитационные, игровые, научно-технические*.

По области знаний выделяются модели *биологические, экономические, исторические, социологические* и т.д.

По фактору времени разделяются модели *динамические* и *статические*. Статическая модель отражает строение и параметры объекта, поэтому ее называют также *структурной*. Она описывает объект в определенный момент времени, дает срез информации о нем. Динамическая модель отражает процесс функционирования объекта или изменения и развития процесса *во времени*.

Любая модель имеет конкретный вид, форму или *способ представления*, она всегда *из чего-то* и *как-то* сделана или представлена и описана. В этом классе, прежде всего, модели рассматриваются как материальные и нематериальные. *Материальные модели* – это материальные копии объектов моделирования. Они всегда имеют реальное воплощение, воспроизводят внешние свойства или внутреннее строение, либо действия объекта-оригинала. Примеры: глобус – модель формы земного шара, кукла – модель внешнего вида человека,. Материальное моделирование использует экспериментальный (опытный) метод познания.

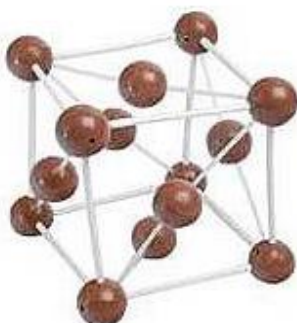
Нематериальное моделирование использует теоретический метод познания. По-другому его называют, *абстрактным, идеальным*. Абстрактные модели, в свою очередь, делятся на воображаемые и информационные.

Информационная модель – это совокупность информации об объекте, описывающая свойства и состояние объекта, процесса или явления, а также связи и отношения с окружающим миром.

Информационные модели представляют объекты в виде, словесных описаний, текстов, рисунков, таблиц, схем, чертежей, формул и т.д. Информационную модель нельзя потрогать, у нее нет материального воплощения, она строится только на информации. Ее можно выразить на языке описания (знаковая модель) или языке представления (наглядная модель).

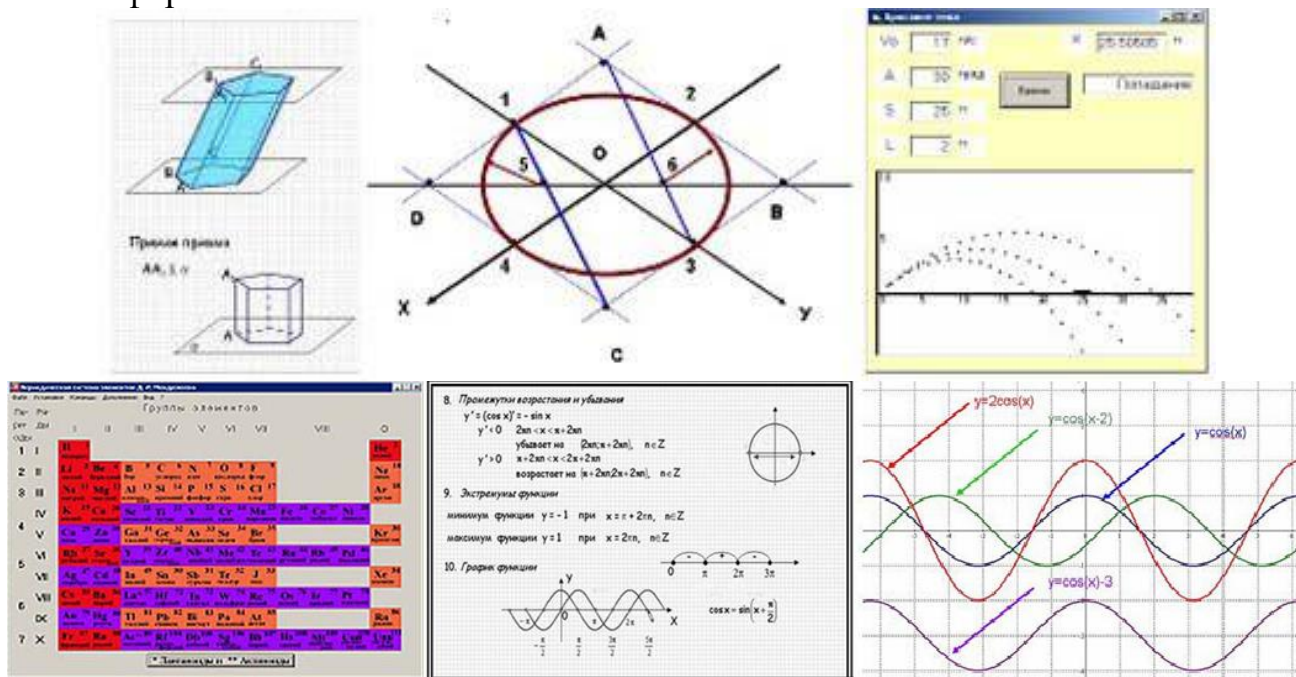


Образные модели: рисунки, фотографии и т. д. представляют зрительные образы, воспроизводят геометрические, физические и другие свойства объектов в материальном мире (например, глобус, муляжи, модели кристаллических решеток, зданий).



Знаковые модели строятся с использованием различных языков (знаковых систем), например, закон Ньютона, таблица Менделеева, карты, графики, диаграммы.

Смешанные модели: представляют объекты и процессы в образной или знаковой форме.



Последовательность компьютерного моделирования и решения задач осуществляется в несколько этапов:

1. Постановка задачи (определение исходных данных и результатов).
2. Построение качественной описательной модели (словесное описание зависимостей результатов от исходных данных на естественном языке).
3. Построение формальной модели (это модели, созданные на формальном языке, то есть научном, профессиональном или специализированном). Примеры формальных моделей: все виды формул, таблицы, графы, карты, схемы и т. д.
4. Компьютерная модель (это модель, реализованная средствами программной среды).
5. Исследование модели (анализ результата): проверка соответствия полученной компьютерной модели поставленной цели в начале моделирования.

Пути построения моделей:

1. текстовые редакторы;
2. графические редакторы;
3. презентации;
4. Macromedia Flash;
5. построение модели с помощью одного из приложений: электронных таблиц, СУБД;
6. построение алгоритма решения задачи и его кодировка на одном из языков программирования (Visual Basic, Паскаль, Basic и т. д.).

Содержание работы

ВНИМАНИЕ: номер варианта определяется по номеру в списке журнала.

Вариант 1 – для нечетных номеров по списку, **вариант 2** – для четных номеров по списку.

Задание №1. Составить математическую модель, построить блок – схему линейного алгоритма задачи.

Порядок работы:

1. Записать текст задачи в тетрадь.
2. Составить краткую запись.
3. Построить блок – схему линейного алгоритма.

Вариант1_Задача №1. Составить программу для вычисления значения периметра треугольника со сторонами А, В и С.

Вариант2_Задача № 1. Составить программу для вычисления значения периметра прямоугольника со сторонами А и В.

Задание №2. Составить математическую модель, построить блок – схему разветвляющегося алгоритма задачи.

Порядок работы:

1. Записать текст задачи в тетрадь.
2. Составить краткую запись.
3. Построить блок – схему разветвляющегося алгоритма.

Вариант1_Задача № 2. Составить программу для нахождения большего из двух чисел.

Вариант2_Задача № 2. Составить программу для нахождения меньшего из двух чисел.

Задание №3. Составить математическую модель, построить блок – схему циклического алгоритма задачи.

Порядок работы:

1. Записать текст задачи в тетрадь.
2. Составить краткую запись.
3. Построить блок – схему циклического алгоритма.

Вариант1_Задача № 3. Составить программу для вычисления произведения чисел от 1 до 10 с шагом 1.

Вариант2_Задача № 3. Составить программу для вычисления суммы чисел от 1 до 100 с шагом 1.

Контрольные вопросы

1. Что такое алгоритм?
2. Перечислить основные свойства алгоритмов.
3. Перечислить формы представления алгоритмов.
4. Что такое линейный алгоритм?
5. Что такое разветвляющийся алгоритм?
6. Что такое циклический алгоритм?
7. Что такое модель?
8. Перечислить виды моделей.
9. Перечислить этапы компьютерного моделирования и решения задач.
10. Перечислить пути построения моделей.