

2 курс

ПЛАН – КОНСПЕКТ
проведения практического занятия № 2
по дисциплине «Информатика»

Раздел 1. «Автоматизированная обработка информации.»

Тема № 1.2: «Технология обработки информации»

Подготовил: преподаватель
В.Н. Борисов

Рязань 2024

**Практическое занятие № 2 «Вычисление количества информации сообщения.»
по Теме № 1.2. «Технология обработки информации».**

Цель занятия: изучить со студентами основные сведения об этапах, стадиях подготовки и обработки информации на вычислительной технике, технологических решениях обработки информации, телекоммуникационных технологиях, об основных структурах алгоритмов, основные определения теории информации, теории вероятностей, практическое применение полученных знаний – выполнение задания, включающего в себя составление алгоритма решения задачи (блок-схемы) по условию задачи, вычисление количества информации сообщения.

Вид занятия: классно-групповое, комбинированное (по проверке знаний, умений по пройденному материалу, по изучению и первичному закреплению нового материала, применению на практике полученных знаний).

Методы проведения занятия: доведение теоретических сведений, выполнение практических заданий.

Время проведения: 2 ч (90 мин.)

Основные вопросы:

1. Ознакомление с этапами подготовки и обработки информации на вычислительной технике (ВТ).
2. Знакомство с основными структурами алгоритмов.
3. Вычисление количества информации сообщения.
4. Применение на практике изученного материала (выполнение практического задания – составление алгоритма решения задачи (блок-схемы) по условию задачи, вычисление количества информации сообщения.

Литература:

1. [2 учебник раздела «Основной учебной литературы» рабочей программы изучения дисциплины]: Гаврилов, М. В. Информатика и информационные технологии : учебник для среднего профессионального образования / М. В. Гаврилов, В. А. Климов. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 355 с. — (Профессиональное образование). — ISBN 978-5-534-15930-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/viewer/informatika-i-informacionnye-tehnologii-510331#page/1>, главы 1,2.

Примерный расчет времени:

1. Вступительная часть – 20 мин.
2. Основная часть – 60 мин.
3. Заключительная часть – 10 мин.

Вступительная часть:

Занятия начать с объявления темы занятия, основных рассматриваемых вопросов, времени изучения темы (нового материала), закрепления на практике полученных знаний, перечисления литературы.

Основная часть (доведение теоретических сведений):

Первый вопрос: Ознакомление с этапами подготовки и обработки информации на вычислительной технике (ВТ).

Сведения по данному вопросу представлены в Теоретических сведениях по теме 1.2.

Второй вопрос: Знакомство с основными структурами алгоритмов. Основные алгоритмические конструкции.

Понятие и свойства алгоритмов.

Алгоритм – заранее заданное понятное и точное предписание возможному исполнителю совершить определенную последовательность действий для получения решения задачи за конечное число шагов.

Это – не определение в математическом смысле слова, а, скорее, описание интуитивного понятия алгоритма, раскрывающее его сущность.

Понятие алгоритма является не только одним из главных понятий математики, но одним из главных понятий современной науки. Основные свойства алгоритмов следующие:

1. **Понятность** для исполнителя – исполнитель алгоритма должен понимать, как его выполнять. Иными словами, имея алгоритм и произвольный вариант исходных данных, исполнитель должен знать, как надо действовать для выполнения этого алгоритма.
2. **Дискретность** (прерывность, раздельность) – алгоритм должен представлять процесс решения задачи как последовательное выполнение простых (или ранее определенных) шагов (этапов).
3. **Определенность** – 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.

4. Результативность (или конечность) состоит в том, что за конечное число шагов алгоритм либо должен приводить к решению задачи, либо после конечного числа шагов останавливаться из-за невозможности получить решение с выдачей соответствующего сообщения, либо неограниченно продолжаться в течение времени, отведенного для исполнения алгоритма, с выдачей промежуточных результатов.

5. Массовость означает, что 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.

Способы описания алгоритма.

На практике наиболее распространены следующие формы представления алгоритмов:

- *словесная* (запись 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).

Словесный способ записи алгоритмов представляет собой описание последовательных этапов обработки данных. Алгоритм задается в произвольном изложении на естественном языке.

Словесный способ не имеет широкого распространения, так как такие описания:

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

Графический способ представления алгоритмов является более компактным и наглядным по сравнению со словесным. *При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий.*

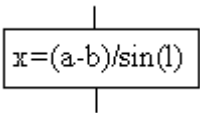
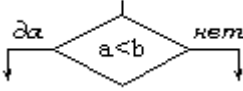
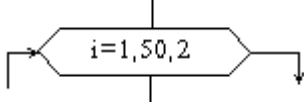
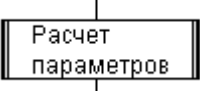
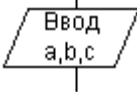
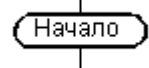

Такое графическое представление называется схемой алгоритма или **блок-схемой**. В блок-схеме каждому типу действий (вводу исходных данных,

вычислению значений выражений, проверке условий, управлению повторением действий, окончанию обработки и т.п.) соответствует геометрическая фигура, представленная в виде *блочного символа*. Блочные символы соединяются *линиями переходов*, определяющими очередность выполнения действий.

Таблица блочных символов.

В таблице приведены наиболее часто употребляемые блочные символы.

В таблице приведены наиболее часто употребляемые символы:

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.

Базовые алгоритмические структуры.

Алгоритмы можно представлять как некоторые структуры, состоящие из отдельных **базовых** (т.е. основных) **элементов**. Естественно, что при таком подходе к алгоритмам изучение основных принципов их конструирования должно начинаться с изучения этих базовых элементов. Для их описания будем использовать язык схем алгоритмов и простейший псевдокод.

Логическая структура любого алгоритма может быть представлена комбинацией трех базовых структур: следование, ветвление, цикл. Характерной особенностью базовых структур является наличие в них **одного входа и одного выхода**.

1.Базовая структура "следование". Образуется последовательностью действий, следующих одно за другим:

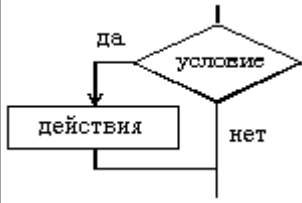
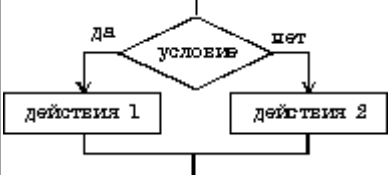
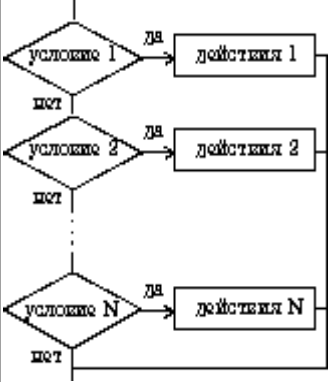
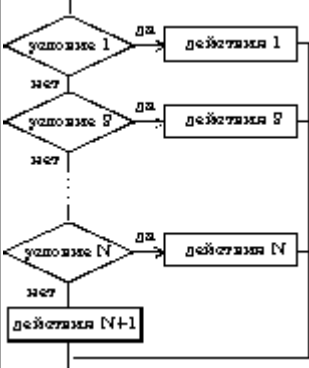
Псевдокод	Язык блок–схем
действие 1 действие 2 действие n	

2.Базовая структура "ветвление". Обеспечивает в зависимости от результата проверки условия (**да** или **нет**) выбор одного из альтернативных путей работы алгоритма. Каждый из путей ведет к *общему выходу*, так что работа алгоритма будет продолжаться независимо от того, какой путь будет выбран.

Структура *ветвление* существует в четырех основных вариантах:

- если–то;

- если–то–иначе;
- выбор;
- выбор–иначе.

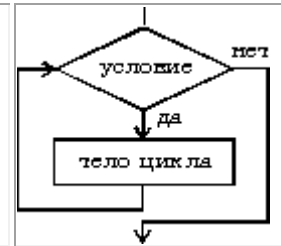
Псевдокод	Язык блок–схем
1. если–то	
если условие то действия все	
2. если–то–иначе	
если условие то действия 1 иначе действия 2 все	
3. выбор	
выбор при условии 1: действия 1 при условии 2: действия 2 при условии N: действия N все	
4. выбор–иначе	
выбор при условии 1: действия 1 при условии 2: действия 2 при условии N: действия N иначе действия N+1 все	

3. Базовая структура "цикл" (повторение). Обеспечивает многократное выполнение некоторой совокупности действий, которая называется телом цикла. Основные разновидности циклов представлены в таблице:

Псевдокод	Язык блок–схем
-----------	----------------

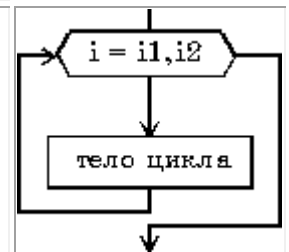
Цикл типа пока. Предписывает выполнять тело цикла до тех пор, пока выполняется условие, записанное после слова пока.

нц пока условие тело цикла (последовательность действий) кц

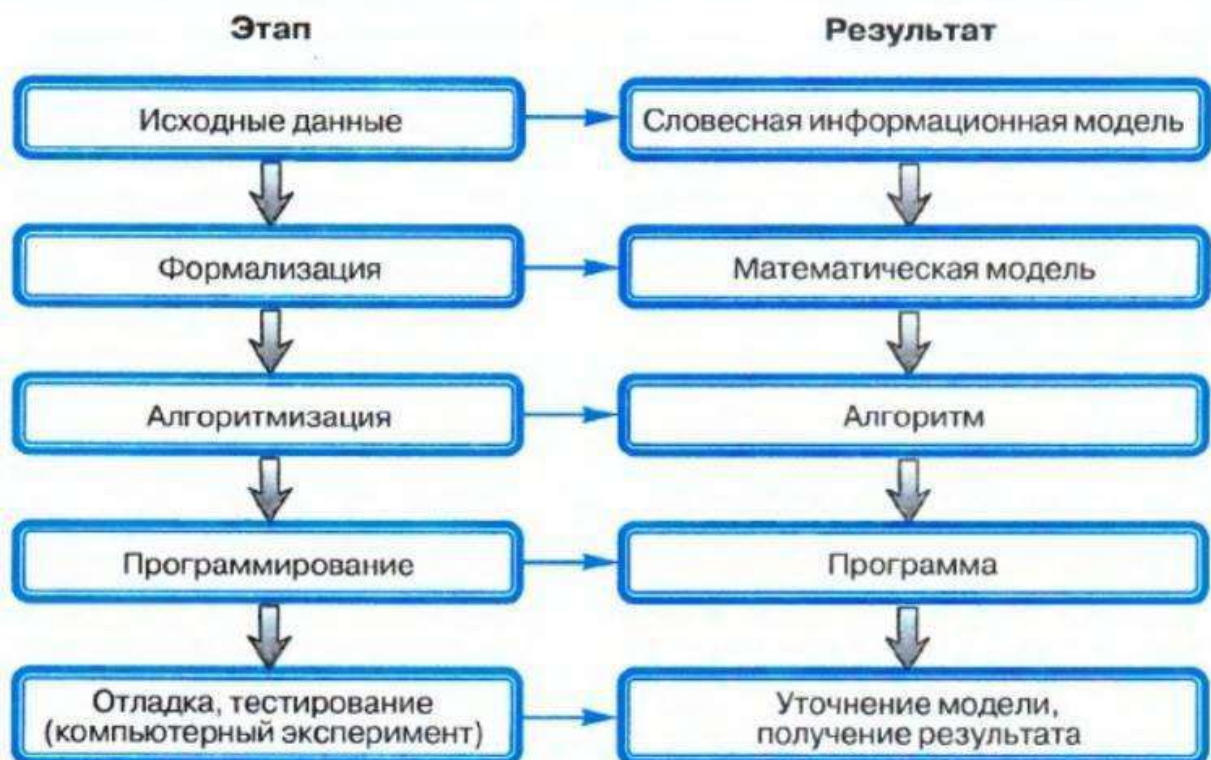


Цикл типа для. Предписывает выполнять тело цикла для всех значений некоторой переменной (параметра цикла) в заданном диапазоне.

нц для i от i_1 до i_2 тело цикла (последовательность действий) кц



Этапы решения задачи на компьютере



Расчет результатов выполнения алгоритма.

Эффективность. Рассматривая различные алгоритмы решения одной и той же задачи, полезно проанализировать, сколько вычислительных ресурсов они требуют (время работы, память), и выбрать наиболее эффективный.

Под временем работы (running time) алгоритма будем подразумевать число элементарных шагов, которые он выполняет. Положим, что одна строка псевдокода требует не более чем фиксированного числа операций (если только это не словесное описание каких-то сложных действий – типа «отсортировать все точки по x -координате»). Следует также различать *вызов (call)* процедуры (на который уходит фиксированное число операций) и её исполнение (*execution*), которое может быть долгим.

Сложность алгоритма – это величина, отражающая порядок величины требуемого ресурса (времени или дополнительной памяти) в зависимости от размерности задачи.

Анализ трудоёмкости алгоритмов.

В качестве критерия оптимальности алгоритма выбирается трудоёмкость алгоритма, понимаемая как количество элементарных операций, которые необходимо выполнить для решения задачи с помощью данного алгоритма.

Функцией трудоёмкости называется отношение, связывающие входные данные алгоритма с количеством элементарных операций.

Одним из упрощенных видов анализа, используемых на практике, является асимптотический анализ алгоритмов. Целью асимптотического анализа является сравнение затрат времени и других ресурсов различными алгоритмами, предназначенными для решения одной и той же задачи, при больших объемах входных данных.

Используемая в асимптотическом анализе оценка функции трудоёмкости, называемая сложностью алгоритма, позволяет определить, как быстро растёт трудоёмкость алгоритма с увеличением объема данных. В асимптотическом анализе алгоритмов используются обозначения, принятые в математическом асимптотическом анализе

Третий вопрос: Вычисление количества информации сообщения.

Сведения по данному вопросу, сведения о стадиях подготовки и обработки информации на вычислительной технике, технологических решениях обработки информации, телекоммуникационных технологиях, основные определения теории информации, теории вероятностей, представлены в Теоретических сведениях по теме 1.2.

Основные понятия.

1 Сообщение несет информацию для человека, если содержащиеся в нем сведения являются для него новыми и понятными.

2 Сообщение, уменьшающее неопределенность знаний в два раза, несет 1 бит информации.

3 Неопределенность знаний о некотором событии — это количество возможных результатов события.

4 Количество информации, содержащееся в сообщении о том, что произошло одно из N равновероятных событий, определяется из решения показательного уравнения: $2^i = N$.

5 Количество информации, содержащейся в сообщении о результатах нескольких (независимых) выборов, должно быть равно сумме количеств информации, содержащейся в сообщениях об этих выборах по отдельности

6 При алфавитном подходе к измерению информации количество информации зависит не от содержания, а от размера текста и мощности алфавита.

7 Алфавит - множество символов, используемых при записи текста. Мощность (размер) алфавита - полное количество символов в алфавите.

8 Если мощность алфавита обозначить N , тогда, согласно известной формуле $N = 2^i$, каждый символ алфавита несет i бит информации. Количество информации одного символа называется весом символа

9 Чтобы найти количество информации во всем тексте, нужно посчитать число символов в нем и умножить на вес одного символа. $J = K \cdot i$ (K – количество символов в тексте, J – количество информации текста или информационный объем текста)

10 Скорость передачи информации (скорость передачи данных) – это количество бит, передаваемых за единицу времени, измеряется в бит/с: $V = \frac{J}{t}$

11 Если события не являются равновероятными, то для вычисления количества информации события необходимо использовать понятие вероятности (отношение благоприятных исходов к общему количеству исходов события)

12 Количественная зависимость между вероятностью события p и количеством возможных исходов события N выражается формулой: $N = 1/p$

Таблица
Количество информации в сообщении
об одном из N равновероятностных событий

N	i	N	i	N	i	N	i
1	0,00000	17	4,08746	33	5,04439	49	5,61471
2	1,00000	18	4,16993	34	5,08746	50	5,64386
3	1,58496	19	4,24793	35	5,12928	51	5,67243
4	2,00000	20	4,32193	36	5,16993	52	5,70044
5	2,32193	21	4,39232	37	5,20945	53	5,72792
6	2,58496	22	4,45943	38	5,24793	54	5,75489
7	2,80735	23	4,52356	39	5,28540	55	5,78136
8	3,00000	24	4,58496	40	5,32193	56	5,80735
9	3,16993	25	4,64386	41	5,35755	57	5,83289
10	3,32193	26	4,70044	42	5,39232	58	5,85798
11	3,45943	27	4,75489	43	5,42626	59	5,88264
12	3,58496	28	4,80735	44	5,45943	60	5,90689
13	3,70044	29	4,85798	45	5,49185	61	5,93074
14	3,80735	30	4,90689	46	5,52356	62	5,95420
15	3,90689	31	4,95420	47	5,55459	63	5,97728
16	4,00000	32	5,00000	48	5,58496	64	6,00000

Задание

Исходные данные:

1 Вычислить количество информации в сообщении о том, что встреча состоится 15 сентября.

Решение:

а) Найдем количество информации сообщения о встрече в сентябре. Т.к. сентябрь – один из месяцев года, то неопределенность $N_1 = 12$, а значит количество информации $i_1 \approx 3,58 \text{ бит}$

б) Т.к. встреча состоится в один из дней сентября, то $N_2 = 31$, а $i_2 \approx 4,95 \text{ бит}$

в) Значит количество информации заданного события

$$i = i_1 + i_2 = 3,58 + 4,95 = 8,53 \text{ бит}$$

Таблица
Количество информации в сообщении об одном из N равновероятностных событий: $x = \log_2 N$

N	x	N	x	N	x	N	x
1	0,00000	17	4,08746	33	5,04439	49	5,61471
2	1,00000	18	4,16993	34	5,08746	50	5,64386
3	1,58496	19	4,24793	35	5,12928	51	5,67243
4	2,00000	20	4,32193	36	5,16993	52	5,70044
5	2,32193	21	4,39232	37	5,20945	53	5,72792
6	2,58496	22	4,45943	38	5,24793	54	5,75489
7	2,80735	23	4,52356	39	5,28540	55	5,78136
8	3,00000	24	4,58496	40	5,32193	56	5,80735
9	3,16993	25	4,64386	41	5,35755	57	5,83289
10	3,32193	26	4,70044	42	5,39232	58	5,85798
11	3,45943	27	4,75489	43	5,42626	59	5,88264
12	3,58496	28	4,80735	44	5,45943	60	5,90689
13	3,70044	29	4,85798	45	5,49185	61	5,93074
14	3,80735	30	4,90689	46	5,52356	62	5,95420
15	3,90689	31	4,95420	47	5,55459	63	5,97728
16	4,00000	32	5,00000	48	5,58496	64	6,00000

Кодирование текстовой информации

Кодирование – преобразование входной информации в машинную форму (в двоичный код).

Декодирование – преобразование двоичного кода в форму, понятную человеку.

Число символов алфавита (мощность алфавита): $N = 2^i$

где **i** – информационный вес одного символа;

Информационный объем текста: $I = i * K$

i – информационный вес одного символа (количество бит на кодирование одного символа);
K – число символов в тексте.

Объем данных при передаче: $I = V * t$,

где **t** — время передачи данных,

V — скорость передачи данных.

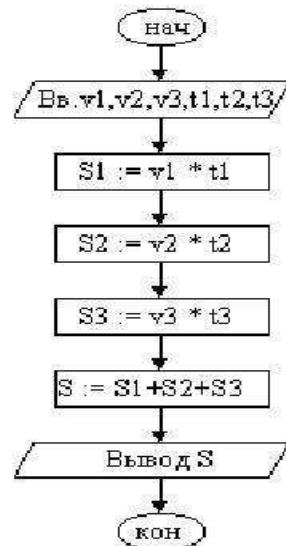
Заключительная часть.

1. Закончить изложение материала.
2. Выдать задание на практическую работу.
3. Ответить на возникшие вопросы.
4. Принять защиту выполненных ранее практических работ.
5. Подвести итоги занятия.
6. Выдать задание на самоподготовку (домашнее задание).

Четвёртый вопрос: Применение на практике изученного материала (выполнение практического задания – составление алгоритма решения задачи (блок-схемы) по условию задачи).

Основные алгоритмические структуры

• Следование



Пешеход шел по пересеченной местности. Его скорость движения по равнине v_1 км/ч, в гору — v_2 км/ч и под гору — v_3 км/ч. Время движения соответственно t_1 , t_2 и t_3 ч.

Какой путь прошел пешеход?

1. Ввести $v_1, v_2, v_3, t_1, t_2, t_3$.
2. $S_1 := v_1 * t_1$.
3. $S_2 := v_2 * t_2$.
4. $S_3 := v_3 * t_3$.
5. $S := S_1 + S_2 + S_3$.
6. Вывести значение S .
7. Конец.

Задача на линейный алгоритм

Вводятся объём и масса тела. Определить плотность материала этого тела.

Входные данные:

V – объём тела (real);

m – масса тела (real);

Выходные данные:

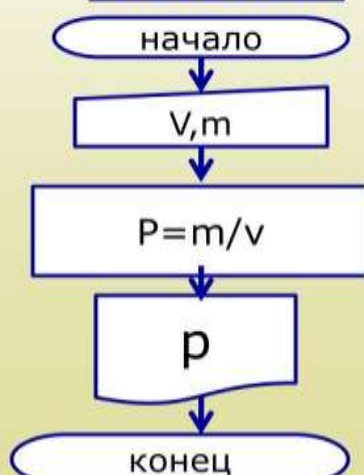
ρ – плотность тела (real).

Паскаль-программа

```

Program z1;
uses crt;
Var V,m,p:real;
BEGIN
Clrscr;
Write('Введите значения объёма и
массы тела: ');
ReadLn(v,m);
P:=m/V;
Write('Плотность тела
объёмом',V:7:1,' и массой' ,m:7:1,'
равна',p:8:2);
ReadLn;
END.
  
```

БЛОК-СХЕМА

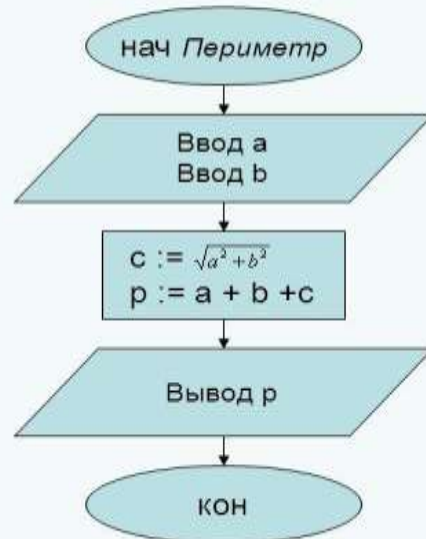


Линейные алгоритмы

Задача 2. Написать программу нахождения периметра прямоугольного треугольника по известным катетам **a** и **b** (**a** и **b** – целые числа, вводимые с клавиатуры).

```

алг периметр
цел a, b
вещ c, p
нач
    ввод a
    ввод b
    c :=  $\sqrt{a^2 + b^2}$ 
    p := a + b + c
    вывод p
кон
  
```



MyShared

Выполнение практического задания.

Цель работы: изучить основные сведения об этапах, стадиях подготовки и обработки информации на вычислительной технике, технологических решениях обработки информации, телекоммуникационных технологиях, об основных структурах алгоритмов, основные определения теории информации, теории вероятностей, практическое применение полученных знаний – выполнение задания, включающего в себя составление алгоритма решения задачи (блок-схемы) по условию задачи, вычисление количества информации сообщения.

Задание: (исходные данные):

1. составить задачу линейного программирования (аналогично представленным выше), составить схему алгоритма ее решения (блок-схему). Указанные задачи должны быть различны у разных подгрупп учебной группы.

2. Вычислить количество информации в сообщении о том, что встреча состоится некоторого числа некоторого месяца. Число и месяц различны у разных подгрупп учебной группы.

3. Подготовить, защитить отчёт о выполнении работы.

Задание на самоподготовку (домашнее задание):

1. Детально проработать, законспектировать материал занятия, размещенный в данном план-конспекте, в теоретических сведениях по теме 1.2, в учебнике, указанном на с.2 текущего документа.
2. Подготовить отчёты о выполнении практической работы, подготовиться к защите данной работы.
3. Подготовиться к опросу по пройденному материалу.