

**Министерство просвещения Российской Федерации
Министерство образования Иркутской области
Департамент образования города Иркутска
Муниципальное автономное общеобразовательное учреждение
Лицей ИГУ города Иркутска
МАОУ Лицей ИГУ г. Иркутска**

РАССМОТРЕНО

на заседании методического объединения
учителей информатики и технологии от
29.08.2023г. протокол №1.

Руководитель МО Л.Н. Шеметова

УТВЕРЖДЕНО

Приказ № 01-06-140 от
30.08.2023 г.

Директор Е.Ю. Кузьмина

ПРИНЯТО

решением педагогического совета

от 30.08.2023 г., протокол №1

ID -

**РАБОЧАЯ ПРОГРАММА
ОСНОВНОГО ОБЩЕГО ОБРАЗОВАНИЯ**

**ID ---
учебного курса**

«ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»

(для 8 классов)

Срок освоения – 1 год

Уровень сложности программы **БАЗОВЫЙ**

Количество часов по программе за весь период реализации - 34

Разработчики: Шеметова Л.Н., учитель информатики,
высшая кв.категория
Медведева Е.Н., учитель информатики

г.Иркутск, 2023 г.

АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ
учебного курса «Введение в программирование»
для учащихся 8 класса

Рабочая программа спецкурса «Введение в программирование» (8 класс) разработана в соответствии с требованиями ФГОС и ФОП основного общего образования и Положением «О рабочих программах учебных предметов, учебных курсов (в том числе внеурочной деятельности), учебных модулей в соответствии с требованиями ФГОС и ФОП основного общего образования» МАОУ Лицея ИГУ г.Иркутска, утвержденного приказом директора 01-06-132 от 30.08.2023 года и является частью основной образовательной программы основного общего образования.

Рабочая программа ориентирована на целевые приоритеты, сформулированные в федеральной рабочей программе воспитания и в рабочей программе воспитания МАОУ Лицей ИГУ г. Иркутска.

Обучение информатике направлено на достижение следующих целей:

1. Сформировать представление об алгоритме и программе; линейных, разветвляющихся, циклических алгоритмах, с подпрограммами.
2. Познакомить с интерфейсом и особенностями работы в интегрированной среде программирования.
3. Сформировать представление о простых и составных типах.
4. Научить решать задачи на составление программ на языке программирования и проводить их тестирование..

Рабочая программа учебного курса «Введение в программирование» реализуется за счет часов учебного плана, формируемой участниками образовательных отношений., относится в предметной области «Математика и информатика».

Срок реализации программы – 1 год (8 класс)

Количество учебных часов, на которые рассчитана программа

	8 класс
Кол-во учебных недель	34
Кол-во часов в неделю	1
Кол-во часов в год	34

В программу включены содержание, планируемые результаты (личностные, метапредметные, предметные), тематическое планирование с учетом рабочей программы воспитания, оценочные и методические материалы.

Рабочая программа рассмотрена на заседании методического объединения учителей-предметников (протокол №1 от 29.08.2023 г.), согласована с заместителем директора МАОУ Лицей ИГУ г. Иркутска, утверждена приказом директора № 01-06-140 от 30.08.2023 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа разработана с учетом требований к планируемым результатам освоения основной образовательной программы МАОУ Лицей ИГУ г. Иркутска, реализующей ФГОС на уровне основного общего образования. Учитывает особенностей организации образовательной деятельности учащихся лицея - расширение содержания образования через введение курсов по выбору предпрофильной межпредметной направленности.

Рабочая программа включает содержание, тематическое планирование, планируемые результаты освоения учебного предмета. В программу включены оценочные материалы по годам обучения (Приложение 1), методические материалы (Приложение 2).

Место в учебном плане: часть, формируемая участниками образовательных отношений
Цели реализации курса

1. Сформировать представление об алгоритме и программе; линейных, разветвляющихся, циклических алгоритмах, с подпрограммами.
2. Познакомить с интерфейсом и особенностями работы в интегрированной среды программирования.
3. Сформировать представление о простых и составных типах.
4. Научить решать задачи на составление программ на языке программирования и проводить их тестирование.

Курс «Введение в программирование» решает следующие задачи с учетом программы воспитания Лицея ИГУ :

- формирование основ мировоззрения, соответствующего современному уровню развития науки информатики;
- развитие алгоритмического мышления как необходимого условия профессиональной деятельности в современном информационном обществе;
- формирование и развитие знаний, умений и навыков работы с информацией, программирования в условиях обеспечения информационной безопасности личности обучающегося;
- воспитание ответственного и избирательного отношения к информации с учётом правовых и этических аспектов её распространения, стремления к продолжению образования в области информационных технологий и созидательной деятельности с применением средств информационных технологий.

СОДЕРЖАНИЕ

1. Введение в языки программирования

Первые компьютеры и языки программирования. История возникновения Паскаля.

2. Основы программирования на языке Паскаль

Основные элементы языка программирования

Основные понятия программирования: алгоритм и программа; трансляторы, кодирование и отладка. Интегрированная среда языка программирования.

Алфавит и зарезервированные слова языка программирования. Понятия переменной, константы и оператора. Основные типы переменных. Операторы ввода и вывода. Форматированный вывод.

Операторы языка

Основные арифметические операции. Оператор присваивания. Встроенные функции. Преобразование типов.

Условный оператор. Полная и краткая формы записи. Логические операции. Простые и составные условия. Правила выполнения логических операций. Составной оператор. Оператор выбора.

Повторяющиеся действия в алгоритмах. Цикл с предусловием. Циклы с постусловием и со счетчиком.

Массивы: определение, виды, описание.

Двумерные массивы.

Символьный тип: использование кодовой таблицы ASCII.

Строковый тип. Процедуры и функции работы со строковыми переменными.

Текстовый и графический режимы. Инициализация и команды графического режима.

Тематическое поурочное планирование

№ п/п	Наименование разделов и тем программы	Количество часов			Электронные (цифровые) образовательные ресурсы	Формы контроля
		Всего	Контрольные работы	Практические работы		
1.	Первые компьютеры и языки программирования. История возникновения языка программирования.	1			http://acmp.ru/	Текущий
Раздел 1. Основные элементы языка программирования (3 часа)						
2.	Основные понятия программирования: алгоритм и программа; трансляторы, кодирование и отладка. Интегрированная среда языка программирования.	1			http://acmp.ru/	Текущий
3.	Алфавит и зарезервированные языка программирования. Первая программа.	1		1	http://acmp.ru/	Текущий
4.	Понятия переменной, константы и оператора. Операторы ввода и вывода. Форматированный вывод.	1		1	http://acmp.ru/	Текущий
Раздел 2. Операторы языка (30 часов)						

5.	Основные арифметические операции. Оператор присваивания: перестановка переменных и обратная запись числа.	1		1	http://acmp.ru/	Текущий
6.	Встроенные функции. Преобразование типов.	1		1	http://acmp.ru/	Текущий
7.	Условный оператор. Полная и краткая формы записи.	1		1	http://acmp.ru/	Текущий
8.	Логические операции. Простые и составные условия. Правила выполнения логических операций.	1		1	http://acmp.ru/	Текущий
9.	Логический тип переменных. Задача определения принадлежности точки заданному множеству точек на координатной плоскости.	1		1	http://acmp.ru/	Текущий
10.	Составной оператор: использование операторных скобок.	1		1	http://acmp.ru/	Текущий
11.	Программирование сложных ветвлений: оператор выбора.	1		1	http://acmp.ru/	Текущий

12.	Повторяющиеся действия в алгоритмах. Цикл с предусловием.	1		1	http://acmp.ru/	Текущий
13.	Циклы с постусловием.	1		1	http://acmp.ru/	Текущий
14.	Циклы со счетчиком.	1		1	http://acmp.ru/	Текущий
15.	Программы с циклами: поиск совершенного числа.	1		1	http://acmp.ru/	Текущий
16.	Программы с циклами: НОД и НОК	1		1	http://acmp.ru/	Текущий
17.	Бинарный поиск.	1		1	http://acmp.ru/	Текущий
18.	Решение задач: построение и реализация алгоритмов с ветвлениями и циклами.	1		1	http://acmp.ru/	Текущий
19.	Решение задач: построение и реализация алгоритмов цикл в цикле.	1		1	http://acmp.ru/	Текущий
20.	Массивы: определение, виды, описание.	1		1	http://acmp.ru/	Текущий
21.	Линейный поиск в массиве.	1		1	http://acmp.ru/	Текущий
22.	Поиск минимального элемента в одномерном массиве.	1		1	http://acmp.ru/	Текущий

23.	Сортировка элементов массива.	1		1	http://acmp.ru/	Текущий
24.	Сортировка элементов массива.	1		1	http://acmp.ru/	Текущий
25.	Решение задач с использованием одномерных массивов.	1		1	http://acmp.ru/	Текущий
26.	Простые типы данных: порядковые и вещественные. Символьный тип: использование кодовой таблицы ASCII.	1		1	http://acmp.ru/	Текущий
27.	Решение задач с символьным типом.	1		1	http://acmp.ru/	Текущий
28.	Строковый тип. Процедуры и функции работы со строковыми переменными.	1		1	http://acmp.ru/	Текущий
29.	Решение задач со строковым типом: слияние слов, поиск подстроки в строке, преобразование слов.	1		1	http://acmp.ru/	Текущий
30.	Решение задач: поиск палиндрома.	1		1	http://acmp.ru/	Текущий
31.	Решение задач: преобразование строковый тип в числовой и наоборот.	1		1	http://acmp.ru/	Текущий

32.	Решение задач со строковым типом.	1		1	http://acmp.ru/	Текущий
33.	Повторение пройденного материала.	1			http://acmp.ru/	Текущий
34.	Соревнование юных программистов.	1	1		http://acmp.ru/	ИТГОВЫЙ

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОГРАММЫ

Личностные образовательные результаты:

Личностные результаты имеют направленность на решение задач воспитания, развития и социализации обучающихся средствами учебного предмета.

В результате изучения информатики на уровне основного общего образования у обучающегося будут сформированы следующие личностные результаты в части:

- сформированность мировоззренческих представлений об информации, информационных процессах и информационных технологиях, соответствующих современному уровню развития науки и общественной практики и составляющих базовую основу для понимания сущности научной картины мира;
- интерес к обучению и познанию, любознательность, готовность и способность к самообразованию, осознанному выбору направленности и уровня обучения в дальнейшем;
- овладение основными навыками исследовательской деятельности, установка на осмысление опыта, наблюдений, поступков и стремление совершенствовать пути достижения индивидуального и коллективного благополучия;
- сформированность информационной культуры, в том числе навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий, а также умения самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;

Метапредметные образовательные результаты:

- умение самостоятельно планировать пути достижения цели, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач.
- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения.

Предметные образовательные результаты:

- формирование информационной и алгоритмической культуры;
- формирование представления об основных изучаемых понятиях: алгоритм и его свойства; линейный алгоритм, разветвляющийся алгоритм, циклический алгоритм; подпрограмма, "программа", "переменная".
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для конкретного исполнителя; формирование знаний об алгоритмических конструкциях, логических значениях и операциях; знакомство с одним из языков программирования и основными алгоритмическими структурами — линейной, условной и циклической.

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

ОБЯЗАТЕЛЬНЫЕ УЧЕБНЫЕ МАТЕРИАЛЫ ДЛЯ УЧЕНИКА

1. Информатика, 8 класс/ Поляков К.Ю., Еремин Е.А., Акционерное общество «Издательство «Просвещение»
2. Информатика, 9 класс/ Поляков К.Ю., Еремин Е.А., Акционерное общество «Издательство «Просвещение»
3. Информатика и ИКТ. Задачник-практикум: в 2 т. / Залогова Л.А. и др.; под ред. И.Г.Семакина, Е.К.Хеннера. – М.: БИНОМ. Лаборатория знаний, 2012

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ДЛЯ УЧИТЕЛЯ

1. Стефан Р. Дэвис - С++ Для чайников. - М.: Вильямс, 2003 . - 336 с.
2. Прата С. - Язык программирования С++. 6 Издание. - М.: Вильямс, 2011 . - 1244 с.
3. Литвиненко Н. А. - Технология программирования на С++. - СПб.: БХВ-Петербург, 2010.- 281 с.

ЦИФРОВЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ И РЕСУРСЫ СЕТИ ИНТЕРНЕТ

1. <http://acmp.ru/>
2. <http://acm.timus.ru/>
3. <http://codeforces.ru>
4. <http://acm.mipt.ru/>
5. <http://habrahabr.ru/>
6. <http://e-maxx.ru/>

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Примерный вариант задания итоговой работы

Составитель Шеметова Л.Н.

Case3. Дан номер месяца — целое число в диапазоне 1–12 (1 — январь, 2 — февраль и т. д.). Вывести название соответствующего времени года ("зима", "весна", "лето", "осень").

If3. Даны три целых числа, одно из которых отлично от двух других, равных между собой. Определить порядковый номер числа, отличного от остальных.

Minmax3. Дано целое число N и набор из N прямоугольников, заданных своими сторонами — парами чисел (a, b) . Найти максимальный периметр прямоугольника из данного набора.

Matrix3. Дана матрица размера $M \times N$. Для каждой строки матрицы найти сумму ее элементов.

Proc3. Описать процедуру $\text{Mean}(X, Y, \text{AMean}, \text{GMean})$, вычисляющую среднее арифметическое $\text{AMean} = (X+Y)/2$ и среднее геометрическое $\text{GMean} = (X \cdot Y)^{1/2}$ двух положительных чисел X и Y (X и Y — входные, AMean и GMean — выходные параметры вещественного типа). С помощью этой процедуры найти среднее арифметическое и среднее геометрическое для пар (A, B) , (A, C) , (A, D) , если даны A, B, C, D .

String3. Дан символ C и строка S . Удвоить каждое вхождение символа C в строку S .

Text3. Дано имя файла и целое число N ($0 < N < 27$). Создать текстовый файл с указанным именем и записать в него N строк длины N ; строка с номером K ($K = 1, \dots, N$) должна содержать K начальных прописных (то есть заглавных) латинских букв, дополненных справа символами "*" (звездочка). Например, для $N = 4$ файл должен содержать строки "A***", "AB**", "ABC*", "ABCD"

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Материал для урока по теме «Программирование на Паскале. Условный оператор»

Составитель: Шеметова Л.Н.

Условный оператор позволяет проверить некоторое условие и в зависимости от результатов проверки выполнить то или иное действие.

Структура условного оператора имеет следующий вид:

Полная форма

IF <условие>

THEN

Begin

<оператор1>;

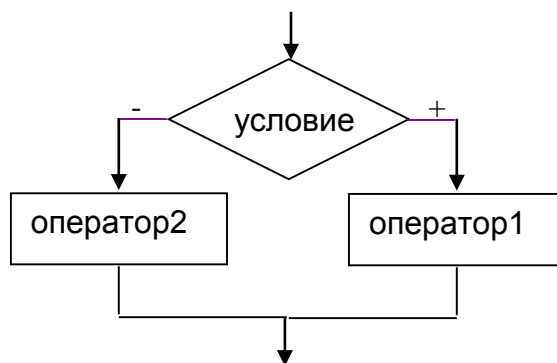
end

ELSE

Begin

<оператор2>;

end;



Неполная форма

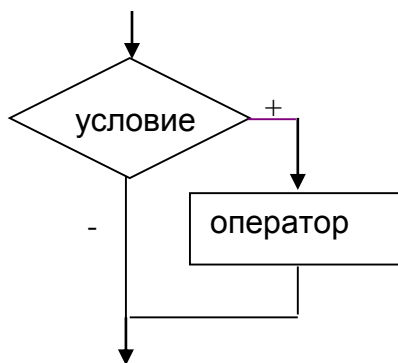
IF <условие>

THEN

Begin

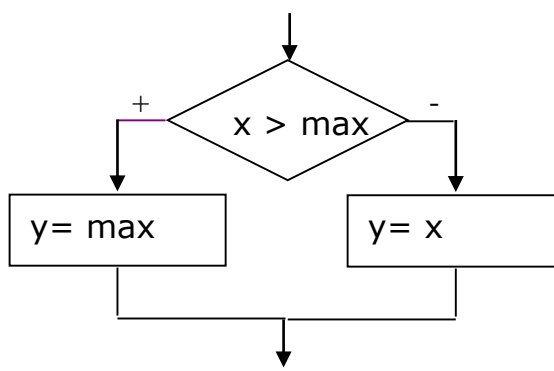
<оператор>;

end



где IF, THEN, ELSE - зарезервированные (если, то, иначе); <условие> - произвольное выражение логического типа; <оператор1>, <оператор2> - любые операторы языка Паскаль.

Условный оператор работает по следующему алгоритму. Вначале



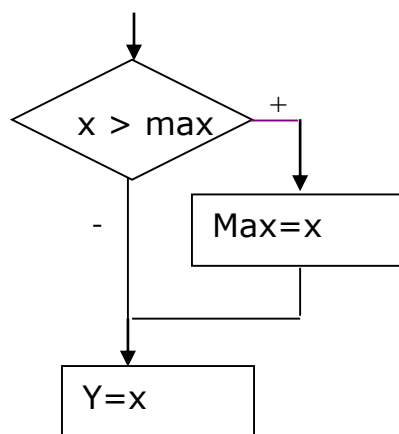
слова

Турбо

вычисляется условное выражение <условие>. Если результат есть TRUE (истина), то выполняется <оператор1>, а <оператор2> пропускается; если результат есть FALSE (ложь), наоборот, <оператор1> пропускается, а выполняется <оператор2>. Например:

```
var
x, y, max: Integer;
begin
.....
if x > max then
y := max else
y := x;
```

При выполнении этого фрагмента переменная Y получит значение переменной X, если только это значение не превышает MAX, в противном случае Y равно MAX.



станет

Часть ELSE <оператор2> условного оператора может быть опущена. Тогда при значении TRUE условного выражения выполняется <оператор1>, в противном случае этот оператор пропускается:

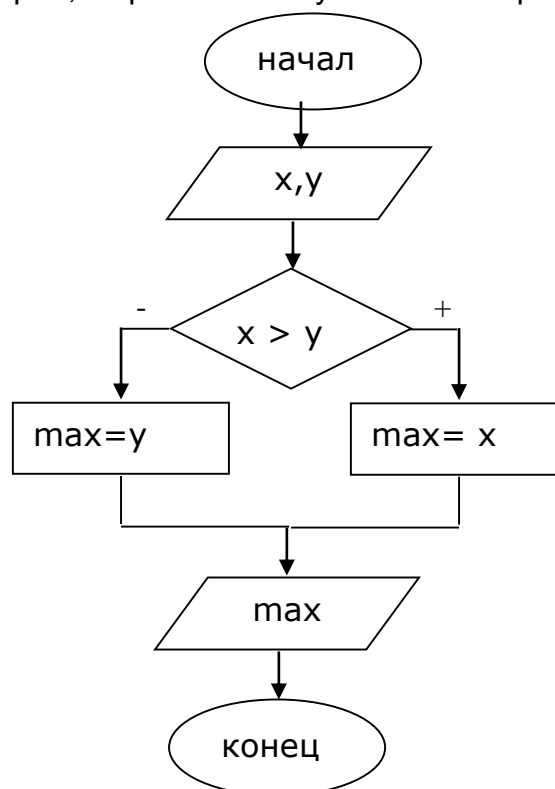
```
var
x, y, max: Integer;
begin
.....
if x > max then
max := x;
Y := x;
```

В этом примере переменная Y всегда будет иметь значение переменной X, а в MAX запоминается максимальное значение X.

Замечание: у оператора стоящего перед else точка с запятой в конце не ставится.

ПРИМЕР. Найти наибольшее из двух неравных чисел. Вывести значение наибольшего числа.

```
Program primer;
Var x,y,max:real;
Begin
Writeln ('введите 2 числа');
Read (x,y);
```



```

If x>y then
max:=x
else
max:=y;
writeln ('значение наибольшего числа', max);
end.

```

Составной оператор

В тех случаях, когда в программе вместо нескольких необходимых операторов, по правилам языка можно поставить только один оператор, применяют составной оператор. Он представляет собой группу операторов ограниченных так называемыми операторными скобками Begin ... end. В этом случае данная группа операторов, состоящая из произвольного количества любых операторов, воспринимается компилятором как один оператор. Например,

```

Begin
z:=a+3;
Write('z=',z);
End;

```

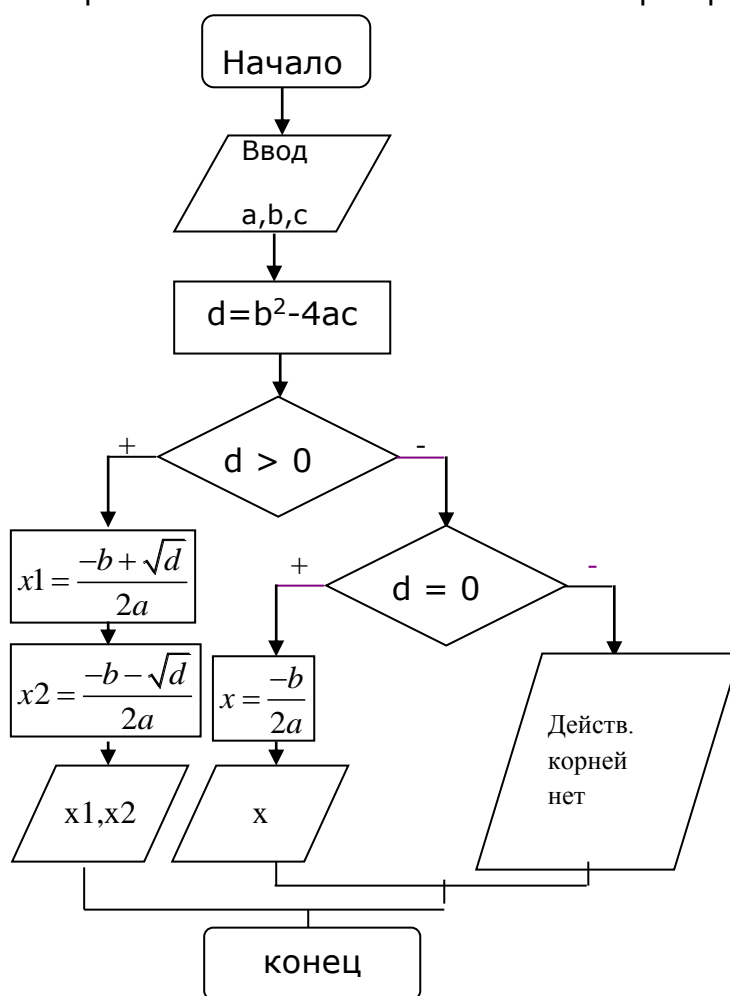
В данном примере два оператора, ограниченные операторными скобками, воспринимаются как один оператор.

ПРИМЕР. Рассмотрим программу, которая находит корни квадратного уравнения $ax^2+bx+c=0$.

```

Program demo;
var
a,b,c : real;
x,x1,x2 : real;
d:real;
begin
Write ( 'Введите коэффициенты уравнения' ) ;
ReadLn(a,b,c);
d:=b*b-4*a*c;
if d > 0 then
begin

```



```

x1:=(-b+sqrt(d))/(2*a);
x2:=(-b-sqrt(d))/(2*a);
writeln ('x1=',x1,'x2=',x2);
end
else
begin
if d=0 then
begin
x:=(-b)/(2*a);
writeln ('x=',x);
end
else
writeln ('действительных корней нет');
end;
end.

```

ЗАДАНИЕ.

1. Даны два числа. Найти и вывести на экран те из них, которые отрицательные.
2. Даны действительные числа x и y . Меньшее из этих двух чисел заменить их полусуммой, а большее – их удвоенным произведением. Результат вычисления вывести на экран монитора.
3. Написать программу, вычисления значения функции

$$y = \begin{cases} x^2 + 3, & \text{при } x < 1 \\ \cos x + 1, & \text{в противном случае} \end{cases}$$

4. Написать программу, вычисления функции

$$y = \begin{cases} x + 1, & \text{при } x < -5 \\ x^2, & \text{при } -5 \leq x < 3 \\ \sin x, & \text{в противном случае} \end{cases}$$

Материалы к уроку по теме «Циклы с параметром»

Составитель: Шеметова Л.Н.

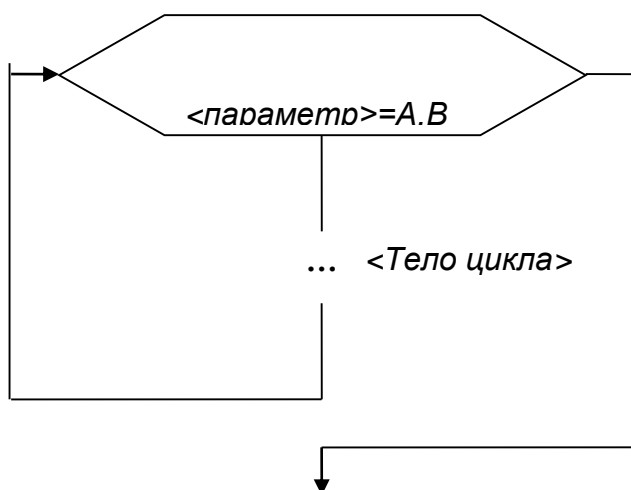
Алгоритм называется циклическим, если в нем повторяются несколько раз одни и те же действия.

Циклические алгоритмы в Паскале записываются с помощью циклических конструкций. Одна из таких конструкций – цикл с параметром **For ... to ... do**.

Структура записи цикла с параметром в программе:

For <параметра>:=A to B do	For <параметра>:=A downto B
Begin	do
... <Тело цикла>	Begin
end;	... <Тело цикла>
	end;

где A – начальное значение параметра,
B – конечное значение параметра.



В блок-схеме эта конструкция записывается с помощью следующего блока:

Оператор цикла с параметром применяется тогда, когда известно число повторений одного и того же действия.

Порядок выполнения циклической конструкции с параметром **For** <параметра>:=A to B do:

- вычисляются значения выражений A и B;
- если $A \leq B$, то параметр последовательно принимает значения A, A+1, A+2, ..., B-1, B и для каждого из этих значений выполняется <Тело цикла>;
- если $A > B$, то <Тело цикла> не будет выполнено ни разу.

Порядок выполнения циклической конструкции с параметром **For** <параметра>:=A downto B do:

- вычисляются значения выражений A и B;
- если $A \geq B$, то параметр последовательно принимает значения A, A-1, A-2, ..., B+1, B и для каждого из этих значений выполняется <Тело цикла>;

- если $A < B$, то *<Тело цикла>* не будет выполнено ни разу.

Если *<Тело цикла>* состоит из нескольких операторов, то операторы тела цикла заключаются в операторные скобки **Begin ...end**, если из одного, то они опускаются.

Пример. Вычислить сумму натуральных чисел от 1 до x . Значение x вводится пользователем с клавиатуры.

Решение:

Дано: x

Найти: Summa

Так как $Summa = 1 + 2 + 3 + 4 + \dots + x$, видим, что повторяется действие – прибавление к переменной последующей цифры. Если начальное значение вычисляемой переменной перед выполнением цикла мы возьмем $Summa = 0$, то за тем $Summa = Summa + 1$, $Summa = Summa + 2$, $Summa = Summa + 3$, $Summa = Summa + 4$, ..., $Summa = Summa + x$. Если в качестве параметра мы возьмем переменную i , которая будет изменяться от 1 до x с шагом 1 принимать значения 1, 2, 3, 4, ..., следовательно формулу для всех шагов цикла можно записать $Summa = Summa + i$.

Program primer;

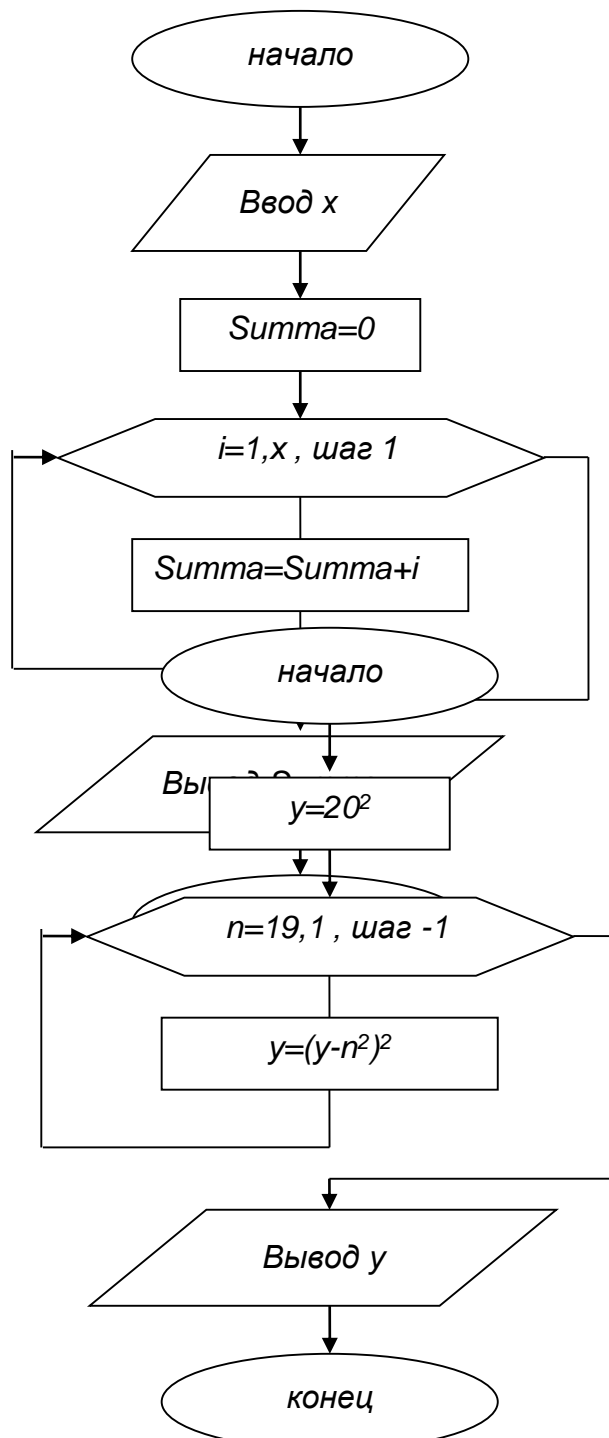
Uses crt;

Var i,x:integer;

Summa:longint;

Begin

Clrscr;



то мы

0, т.е

(т.е. x), то

```
Writeln ('Введите x');  
Read (x);  
Summa:=0;  
For i:=1 to x do  
Summa:=Summa+i;  
Writeln ('сумма =', Summa);  
Readkey;  
End.
```

Пример 2. Составить программу вычисления значения выражения

$$y=((\dots(20^2-19^2)^2-18^2)^2-\dots-1^2)^2$$

Решение. Пусть начальное значение выражения $y=20^2$. То за тем значение y будет вычисляться по формуле $y=(y-19^2)^2$, $y=(y-18^2)^2$, $y=(y-17^2)^2$, ..., $y=(y-1^2)^2$. Обозначим параметр цикла – n и он будет принимать значения от 19 до 1, с шагом -1. Тогда тело цикла будет состоять из формулы: $y=(y-n^2)^2$
Самостоятельно записать программу. Отладить и проверить.

Задание.

1. Составить программу для вычисления произведения натуральных чисел от n до m . Значения m , n – вводятся с клавиатуры.
2. Составить программу вычисления суммы кубов чисел от 25 до 125.
3. Из чисел от 10 до 99 вывести те, сумма цифр которых равна n . Значение n – вводится с клавиатуры.