

Охраняется законом об авторском праве. Воспроизведение всего пособия или его части, а также реализация тиража запрещается без письменного разрешения издателя.

**АЛГОРИТМИЗАЦИЯ и
ПРОГРАММИРОВАНИЕ на языке BASIC
в схемах и таблицах**

Автор-составитель:

Марина Михайловна Хабибулина,
учитель информатики и математики высшей категории

Вёрстка: В.Х. Хабибулин

Печатно-издательский центр «ОБРАЗ» МБОУ «Боханская СОШ№1»

669311 п.Бохан Иркутской обл., ул. Советская, д.15

тел. 8(39538)25307

e-mail: bohansoch1@mail.ru

ПИЦ «ОБРАЗ»
МБОУ «Боханская СОШ№1»

An orange speech bubble with a black outline and a drop shadow, containing the main title text.

**АЛГОРИТМИЗАЦИЯ и
ПРОГРАММИРОВАНИЕ
на языке BASIC**

в схемах и таблицах

СОДЕРЖАНИЕ:

1. АЛГОРИТМ. СВОЙСТВА АЛГОРИТМА
2. ФУНКЦИОНАЛЬНЫЕ ЭЛЕМЕНТЫ БЛОК-СХЕМ
3. ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ И ИХ ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ QBASIC
4. ЯЗЫК ПРОГРАММИРОВАНИЯ BASIC. ОСНОВНЫЕ ОПЕРАТОРЫ И ФУНКЦИИ BASIC
5. СТРОКОВЫЕ ОПЕРАТОРЫ И ФУНКЦИИ BASIC
6. ОПЕРАТОРЫ И ФУНКЦИИ ГРАФИКИ BASIC

ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА И ДРУГИЕ ИСТОЧНИКИ:

1. Есипов А.С. Информатика. Учебник по базовому курсу общеобразовательных учебных заведений. – СПб: Наука и Техника, 2001 г. – 384 с., ил.
2. Программирование. 7-11 классы: информационно-познавательная деятельность учащихся / авт.-сост. М.Н. Капранова. – Волгоград: Учитель, 2012 г. – 143 с.
3. Чернов Б.И. Программирование на алгоритмических языках Бейсик, Фортран, Паскаль / Б.И.Чернов. – М.: Просвещение, 1991 г. – 190 с.

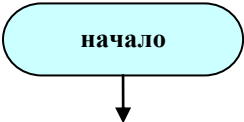
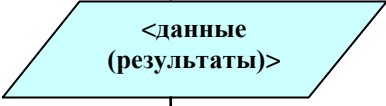
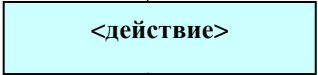
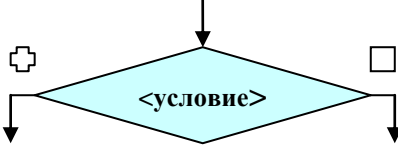
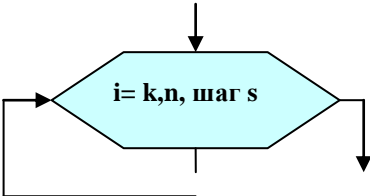

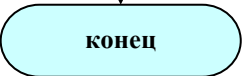
ОСНОВНЫЕ ОПЕРАТОРЫ И ФУНКЦИИ ГРАФИКИ BASIC

SCREEN	<i>установка режима экрана дисплея</i>		SCREEN [режим] [, [цвет]] [, [активная страница]] [, [видимая страница]]		
режим	разрешение	формат текста	матрица символов	число цветов	
0	360x400	40x25	9x16	64	
1	320x200	40x25	8x8	16	
2	640x200	80x25	8x8	16	
7	320x200	40x25	8x8	16	
8	640x200	80x25	8x8	16	
9	640x350	80x25	8x14	64	
12	640x480	80x30	8x16	256К	
13	320x200	40x25	8x8	256Л	
WINDOW	<i>определение координатной плоскости (с левого нижнего к правому верхнему углу)</i>		WINDOW (x1,x2) – (y1,y2)		
COLOR	<i>определение цвета</i>		для режимов 7-10: COLOR [передний план] [, [фон]] для режимов 11-13: COLOR [передний план]		
номер цвета	цвет	номер цвета	цвет	номер цвета	цвет
0	Черный	6	Коричневый	12	Светло-красный
1	Синий	7	Белый	13	Светло-пурпурный
2	Зелёный	8	Серый		
3	Голубой	9	Светло-синий	14	Жёлтый
4	Красный	10	Светло-зелёный	15	Ярко-белый
5	Пурпурный	11	Светло-голубой		
LINE	<i>построение линий или прямоугольников</i>		LINE [[STEP] (x1,y1)] – [STEP] (x2,y2) [, [цвет] [, B [F]]]		
CIRCLE	<i>построение окружностей, эллипсов и дуг</i>		CIRCLE [STEP] (x,y), радиус [, [цвет] [, [начало дуги], [конец дуги] [отношение]]		
PSET PRESET	<i>рисование точек рисование точек фона</i>		PSET [STEP] (x,y) [, цвет] PRESET [STEP] (x,y) [, цвет]		
PAINT	<i>закрашивание области экрана</i>		PAINT [STEP] (x,y) [, цвет внутри] [, цвет границы]		
DRAW	<i>перемещение текущей позиции</i>		DRAW “команды”		
жесткие команды		описание		префиксы	
Up		вверх		B	
Dn		вниз			
Ln		влево		N	
Rn		вправо			
En		по диагонали вверх и вправо			
Fn		по диагонали вниз и вправо			
Gn		по диагонали вниз и влево			
Hn		по диагонали вверх и влево			
мягкие команды		описание			
An		установка угла поворота кратного 90n, где n=0,1,2,3			
TAn		поворот на угол n, где n= от -360 до +360; если n – положительный, то поворот против часовой стрелки, n – отрицательный – поворот по часовой стрелке			
Sn		установка масштабного коэффициента n (от 1 до 255), по умолчанию n=4			
Cn		установка цвета с номером n рисуемых линий (цвет переднего плана)			

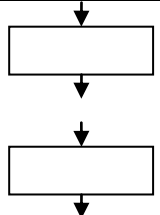
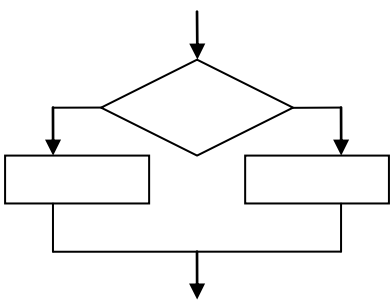
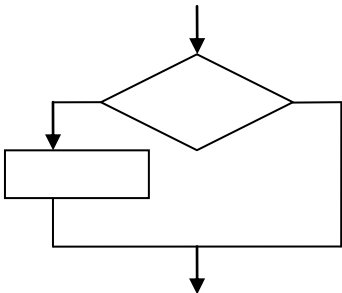
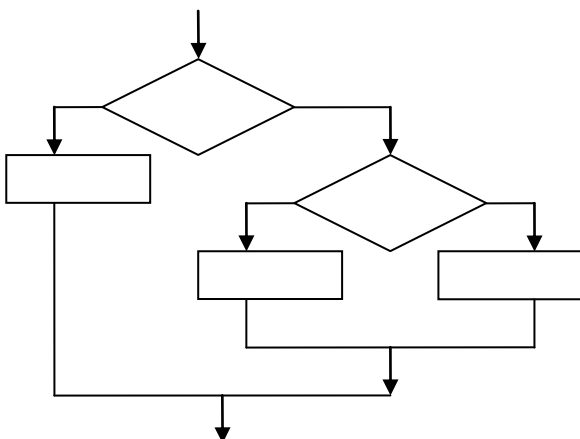
АЛГОРИТМ. СВОЙСТВА АЛГОРИТМА

АЛГОРИТМ - описание строгой последовательности конечного набора действий (команд), приводящей к результату поставленной задачи.		
<pre> graph LR A[Исходные данные] --> B[Алгоритм] B --> C[Результат] D[ИСПОЛНИТЕЛЬ: - формальный, - неформальный] <--> B </pre>		
Свойства алгоритма	<ul style="list-style-type: none"> - дискретность - понятность - определённость - результативность - массовость 	<p><i>пошаговая запись и выполнение</i></p> <p><i>команды понятны для восприятия и выполнения исполнителем (СКИ – система команд исполнителя)</i></p> <p><i>чёткость, строгость и однозначность команд</i></p> <p><i>обязательное получение результата поставленной задачи</i></p> <p><i>возможность применения алгоритма для решения класса однотипных задач при различных значениях исходных данных</i></p>
Способы записи алгоритма		
словесные	графические	на алгоритмических языках
<ul style="list-style-type: none"> - словесное описание - построчная запись 	<ul style="list-style-type: none"> - рисунки, чертежи - схемы - блок-схемы - структурограммы 	<ul style="list-style-type: none"> - школьный алгоритмический язык - языки программирования
Величины (данные) алгоритма		
константы	переменные	массивы
<i>число, текст или логическое значение, которые не изменяются</i>	<i>величины, которым могут присваиваться различные значения</i>	<i>множество однотипных элементов, объединённых одним именем (идентификатором)</i>
	типы переменных: <ul style="list-style-type: none"> - числовые - символьные (литерные, строковые) - логические 	типы массивов: <ul style="list-style-type: none"> - числовые - символьные - логические размерность массива: <ul style="list-style-type: none"> - одномерные, двухмерные и т.д. размер массива: <ul style="list-style-type: none"> - кол-во элементов массива в каждом измерении

ФУНКЦИОНАЛЬНЫЕ ЭЛЕМЕНТЫ БЛОК-СХЕМ

наименование блока	обозначение	описание
НАЧАЛА	 <p style="text-align: center;">(прямоугольник с закругленными углами)</p>	Обозначение начала алгоритма, входа во вспомогательный алгоритм
ВВОДА (ВЫВОДА)	 <p style="text-align: center;">(параллелограмм)</p>	Описание ввода данных или вывода данных (результатов)
ПРОЦЕССА РЕШЕНИЯ	 <p style="text-align: center;">(прямоугольник)</p>	Описание последовательности действий, которые могут быть простыми или составными командами алгоритма
ПРИНЯТИЯ РЕШЕНИЯ	 <p style="text-align: center;">(ромб)</p>	Обозначение условия в алгоритмических структурах ветвления или повторения (выходы «да» или «нет»)
МОДИФИКАЦИИ, ИЗМЕНЕНИЯ ПАРАМЕТРА	 <p style="text-align: center;">(шестиугольник)</p>	Обозначение цикла с параметром (от k до n с шагом s)
КОММЕНТАРИЕВ	 <p style="text-align: center;">(прямоугольник со срезанным углом)</p>	Объявление переменных или ввода комментариев
КОНЦА	 <p style="text-align: center;">(прямоугольник с закругленными углами)</p>	Обозначение конца алгоритма, выхода из вспомогательного алгоритма

ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ И ИХ ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ QBASIC

СЛЕДОВАНИЕ	
- алгоритмическая конструкция, отражающая последовательный порядок действий. Алгоритмы, в которых используется только структура «следование», называются линейными алгоритмами .	
	
ВЕТВЛЕНИЕ	
- алгоритмическая конструкция, в которой в зависимости от результата проверки условия («да» или «нет») предусмотрен выбор одной из двух последовательностей действий (ветвей). Алгоритмы, в основе которых лежит структура «ветвление», называют разветвляющимися .	
полное ветвление	
	<p><u>если</u> <условие> <u>то</u> <действие 1> <u>иначе</u> <действие 2></p> <hr/> <p>IF <условие> THEN <действие 1> ELSE <действие 2></p>
неполное ветвление	
	<p><u>если</u> <условие> <u>то</u> <действие ></p> <hr/> <p>IF <условие> THEN <действие ></p>
вложенное ветвление	
	<p><u>если</u> <условие 1> <u>то</u> <действие 1> <u>иначе если</u> <условие 2> <u>то</u> <действие 2> <u>иначе</u> <действие 3> <u>все</u></p> <hr/> <p>IF <условие 1> THEN <действие 1> ELSEIF <условие 2> THEN <действие 2> ELSE <действие 3> END IF</p>

ветвление «выбор-иначе» [«выбор»]	
	SELECT CASE выражение для проверки CASE список 1 Блок операторов 1 CASE список 2 Блок операторов 2 ... [CASE ELSE Блок операторов n] END SELECT
ПОВТОРЕНИЕ	
- алгоритмическая конструкция, представляющая собой последовательность действий, выполняемых многократно. Алгоритмы, содержащие конструкцию повторения, называют циклическими или циклами . Последовательность действий, многократно повторяющаяся в процессе выполнения цикла, называется телом цикла .	
цикл типа «для» (со счётчиком)	
	для счётчик=начало до конец [с шагом шаг] операторы следующий [счётчик] [, счётчик]
FOR счётчик=начало TO конец [STEP шаг] операторы NEXT [счётчик] [, счётчик]	
цикл типа «пока» (с предусловием)	
	пока <условие> нц выполнение действий, входящих в тело цикла кц
DO [{ WHILE UNTIL } условие] операторы LOOP	
цикл типа «до» (с постусловием)	
	делаем нц выполнение действий, входящих в тело цикла кц пока <условие>
DO операторы LOOP [{ WHILE UNTIL } условие]	

ЯЗЫК ПРОГРАММИРОВАНИЯ BASIC

*«Скажи мне, и я забуду;
Покажи мне, я запомню;
Дай сделать – я научусь!»*
китайская мудрость

BASIC (англ. - основной, фундаментальный, существенный) Beginner All-purpose Symbolic Instruction Code – многоцелевой язык символьческих команд для начинающих (действует с 1965г.)				
ЯЗЫК ПРОГРАММИРОВАНИЯ: ИНТЕРПРЕТАТОРНЫЙ, ПРОЦЕДУРНЫЙ, ИМПЕРАТИВНЫЙ				
Алфавит языка Basic				
наборы символов		знаки арифметических операций		знаки операций отношений
A-Z, a-z 0, 1, 2, ..., 9		+	сложение	= равно
		-	вычитание	> больше
		*	умножение	< меньше
		/	деление	>= больше или равно
		\	деление нацело	<= меньше или равно
		^	остаток от деления возведение в степень	<> не равно
знаки препинания, символы и разделители		символы объявления типа величин		логические операции
., : ; - ! ? «» ‘ () [] _ пробел { } @ ~ №		%	целое число одинарной точности (процент)	AND И (логическое умножение)
		&	целое число двойной точности (амперсант)	OR ИЛИ (логическое сложение)
		!	вещественное число одинарной точности	NOT НЕ (логическое отрицание)
		#	вещественное число двойной точности	EQV эквивалентность
		\$	знак строкой величины	IMP импликация XOR сумма операндов по модулю 2
Стандартные функции				
SIN(x)	sin(x)	x задан в радианах	INT (x)	округление до ближайшего целого
COS(x)	cos(x)	x задан в радианах		
TAN(x)	tg(x)	x задан в радианах	CINT(x)	возвращение значения аргумента, округлённого до целого
ATN(x)	arctg(x)	x задан в радианах		
SQR(x)	\sqrt{x}	$x \geq 0$	FIX(x)	отброс дробной части
ABS(x)	x	x – числовое выражение	RND(1)	случайное число от 0 до 1
EXP(x)	e^x	e=2,7	SGN(x)	знак x
LOG(x)	ln x	логарифм натуральный	x MOD y	остаток от деления x на y
Основные операторы				
оператор	назначение		синтаксис	
REM	<i>комментарий</i>		REM [комментарий] или ‘ [комментарий]	
LET	<i>присвоение значений</i>		LET переменная = выражение	
INPUT	<i>ввод данных</i>		INPUT [;] [“приглашение” {; ,}] переменные	
PRINT	<i>вывод данных</i>		PRINT [список] [{; ,}]	
END	<i>завершение программы, процедуры или блокового оператора</i>		END [{DEF FUNCTION IF SELECT SUB TYPE}]	

ОСНОВНЫЕ СТРОКОВЫЕ ОПЕРАТОРЫ И ФУНКЦИИ BASIC

ASC	<i>функция перевода первого символа символьной строки в число, которым этот символ кодируется в ASCII</i>	x=ASC(символьное выражение)
CHR\$	<i>функция перевода целого числа в символ, который этим числом кодируется в компьютере</i>	x\$=CHR\$ (целое выражение)
LEN	<i>функция определения длины символьной строки</i>	x=LEN (символьное выражение)
MID	<i>оператор выделяет из заданной строки часть заданной длины, начиная с заданного символа (функция заменяет часть символов)</i>	x\$=MID(строка, начало [,длина]) MID\$ (строка1, начало [,длина])=строка2
STRING	<i>функция создания строки заданной длины из заданных символов</i>	x\$=STRING\$(длина, {код строка})
LOCATE	<i>перемещает курсор на заданную позицию</i>	LOCATE [строка][,][колонка] [,][курсor]]]
STR	<i>функция преобразования численных данных в символьную строку</i>	x\$=STR\$ (целое выражение)
VAL	<i>функция перевода символьной строки в число</i>	x=VAL(строка)
INKEY	<i>функция, дающая символ, введённый с клавиатуры</i>	x\$=INKEY\$
INSTR	<i>функция поиска первого включения одного текста в другой текст и возвращения номера позиции начала включения</i>	INSTR ([начало,] строка1, строка2)
LEFT	<i>функция выделения n символов в левой части символьной строки</i>	LEFT\$(x\$,n)
RIGHT	<i>функция выделения n символов в правой части символьной строки</i>	RIGHT\$(x\$,n)
SPACE	<i>функция формирования символьной строки из n пробелов</i>	SPACE\$(n)
LCASE	<i>функция возвращения строки x\$, в которой все прописные буквы заменены строчными</i>	LCASE\$(x\$)
UCASE	<i>функция замены строчных букв строки x\$ на прописные</i>	UCASE\$(x\$)
OCT (HEX)	<i>функция округления до целого в восьмеричной (шестнадцатеричной) системе счисления</i>	OCT\$(x) (HEX\$(x))

DATA	<i>хранение констант, считываемых этим оператором DATA</i>	DATA константа
READ	<i>считывание величин из списка, задаваемого оператором DATA и присваивание их соответствующим переменным</i>	READ список
RESTORE	<i>повторное считывание данных в операторе DATA</i>	RESTORE [{номер метка}]
CLS	<i>очистка экрана</i>	CLS
PRINT USING	<i>выведение строковых и числовых данных в соответствии с заданным форматом</i>	PRINT USING формат; список [{,;}]
DIM	<i>объявление массивов и переменных</i>	DIM переменная [(индексы)] [, переменная [(индексы)] ...]
RANDOMIZE	<i>переустановка базы генератора случайных чисел</i>	RANDOMIZE [база]
TIMER	<i>возвращение числа секунд от начала суток до текущего момента</i>	TIMER
TIME	<i>определение текущего времени</i>	в кач-ве оператора: TIME\$=символьное выр-е в кач-ве функции: TIME\$
SWAP	<i>обмен величин двух переменных</i>	SWAP переменная1, переменная2