

АЛГОРИТМ

Алгоритм — это предписание исполнителю совершать определённую последовательность действий для получения решения задачи за конечное число шагов.

Алгоритмизация — процесс разработки алгоритма для решения задачи.

Свойства алгоритмов

1. **Понятность** — алгоритм должен быть написан на понятном для исполнителя языке.
2. **Дискретность** (прерывность, раздельность) — алгоритм должен представлять запись отдельных шагов. Необходимо использовать минимальное количество шагов. Каждый шаг должен приносить определённый результат.
3. **Определённость** — алгоритм должен быть чётким и восприниматься однозначно.
4. **Результативность** (или конечность) — алгоритм должен обязательно приводить к результату.
5. **Массовость** — алгоритм решения задачи применим для некоторого класса задач, различающихся лишь входными данными.

Способы записи алгоритмов

- Словесный (описание алгоритма на естественном языке).
- Графический (алгоритм составляется из специальных блоков — геометрических фигур).
- Псевдокоды (описания алгоритмов на условном алгоритмическом языке).
- Программный (тексты на различных языках программирования).

ВЕТВЛЕНИЕ

Ветвлением называется алгоритмическая структура, в которой выбирается один из возможных путей выполнения алгоритма с неизменным выходом на общее продолжение.

Каждый путь называется **ветвью алгоритма**.

Существует полное и неполное ветвление.

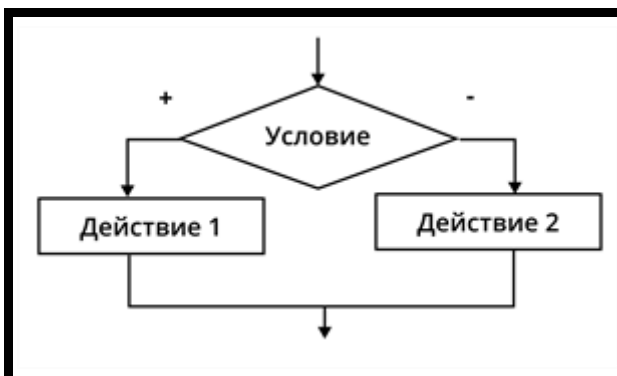


Рис. 1. Полное ветвление

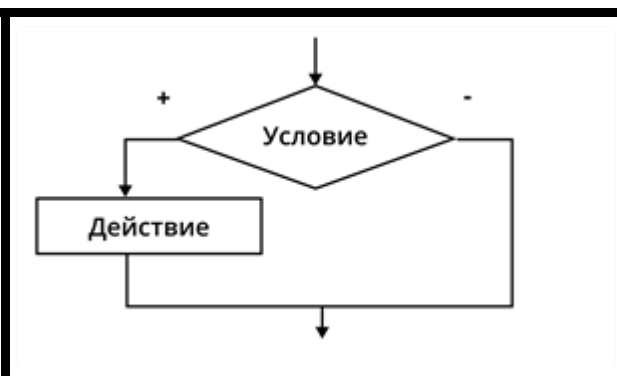


Рис. 2. Неполное ветвление

Программный код в Pascal и Python

Язык программирования Pascal

Полное ветвление	Неполное ветвление
if <условие> then <действие 1> else <действие 2>	if <условие> then <действие 1>

Язык программирования Python

Полное ветвление	Неполное ветвление
if <условие>: <действие 1> else : <действие 2>	if <условие>: <действие >

Алгоритмическая структура «Выбор»

В алгоритмической структуре «Выбор» выполняется одна из нескольких последовательностей команд при истинности соответствующего условия. Существуют следующие алгоритмические структуры «Выбор».

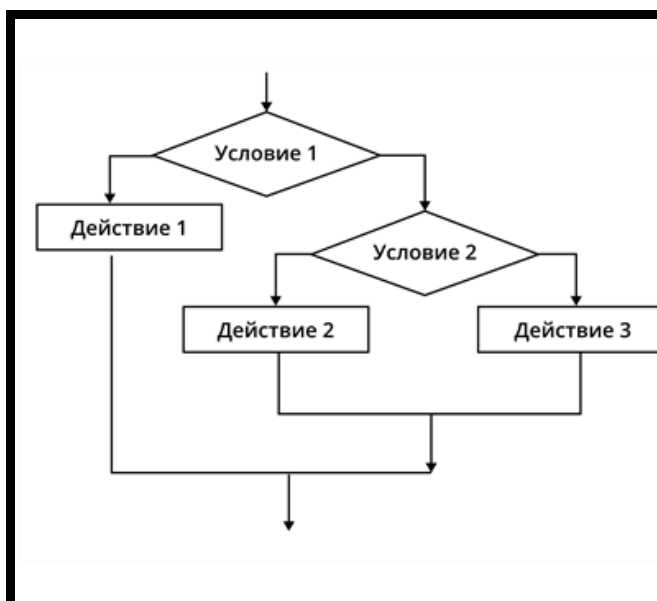


Рис. 3. Структура «Выбор»

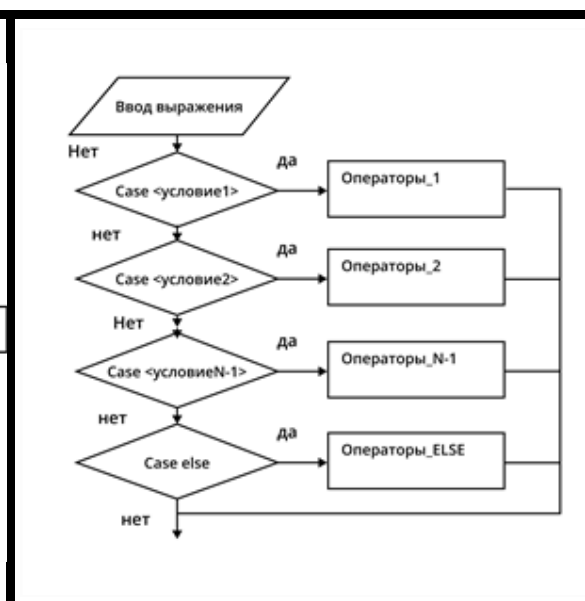


Рис. 4. Структура Case

Программный код в Pascal и Python

Pascal

Структура «Выбор»	Структура Case
if <условие 1> then <действие 1>	case <параметр> of вариант_1: <действие 1>

<pre> else if <условие 1> then <действие 2> else <действие N> </pre>	<pre> вариант_2: <действие 2> else: <действие N> end; </pre>
--	--

Python

Структура «Выбор» (каскадное ветвление)
<pre> if <условие 1>: <действие 1> elif <условие 2>: <действие 2> else: <действие 2> </pre>

Условия бывают **простые** и **составные** (сложные).

Простые условия используют один знак сравнения.

Pascal: >, <, =, >=, <=.

Python: >, <, =, >=, <=.

Составные (сложные) условия — это несколько простых, соединённых «логическими связками» **or**, **and**, **not** и др.

Пример на Pascal	Пример на Python
<pre> (a>10) and (a<=20) (a<20) or (b>10) </pre>	<pre> a>10 and a<=20 a<20 or b>10 </pre>

ПРОЦЕДУРЫ и ФУНКЦИИ

Процедура — это вспомогательный алгоритм, который исполняет некоторые действия.

Процедура записывается до основной программы и вызывается внутри неё.

Синтаксис процедуры

Процедура в Pascal	Процедура в Python
<pre> procedure название (параметры); {начало процедуры} var...; {объявление локальных переменных} begin ... {тело процедуры} end; {конец процедуры} </pre>	<pre> def название (параметры): #объявление переменной #тело процедуры </pre>

Обрати внимание!

Имена переменных в процедуре не должны совпадать с именами в основной программе.

Функция — это вспомогательный алгоритм, который, в отличие от процедуры, возвращает значение (результат).

Функции используются для вычисления каких-либо значений. Например, вычислить факториал, перевести число и др.

Синтаксис функции

Функция в Pascal	Функция в Python
function название (параметры): тип результата; {целый, вещ., строк. и т. д.} var ...; {объявление переменных}; begin ... {тело функции} название: = результат; {результат вычислений всегда присваивается функции} end ; {конец функции}	def название (параметры): ...#объявление переменной ...#тело функции