**Практическая работа.**

**Создание блок-схемы алгоритмов.**

**Цель работы**: Научиться логически правильно разрабатывать блок-схему алгоритма для решения задач.

**Оборудование**: Персональный компьютер.

**Теоретическая часть.**

**Построение алгоритма.**

**Алгоритм** — строго определенная последовательность действий для некоторого исполнителя, приводящая к поставленной цели или заданному результату за конечное число шагов. Любой алгоритм составляется в расчете на конкретного исполнителя с учетом его возможностей.

**Исполнитель** — субъект, способный исполнять некоторый набор команд. Совокупность команд, которые исполнитель может понять и выполнить, называется системой команд исполнителя. Для выполнения алгоритма исполнителю недостаточно только самого алгоритма.

**Выполнить алгоритм** — значит применить его к решению конкретной задачи, т. е. выполнить запланированные действия по отношению к определенным входным данным. Поэтому исполнителю необходимо иметь исходные (входные) данные — те, что задаются до начала алгоритма.

**Исходные (входные) данные**

**Алгоритм**

**Выходные данные (результат)**

В результате выполнения алгоритма исполнитель должен получить искомый результат — выходные данные, которые исполнитель выдает как результат выполненной работы. В процессе работы исполнитель может создавать и использовать данные, не являющиеся выходными, — промежуточные данные.

**Блок-схема.**

**Блок–схема** — наглядный способ представления алгоритма. Блок–схема отображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий. Определенному типу действия соответствует определенная геометрическая фигура блока. Линии, соединяющие блоки, определяют очередность выполнения действий. По умолчанию блоки соединяются сверху вниз и слева направо. Если последовательность выполнения блоков должна быть иной, используются направленные линии(стрелки).

Предпочтительнее до записи на алгоритмическом языке представить алгоритм в виде блок-схемы. Для построения алгоритма в виде блок-схемы необходимо знать назначение каждого из блоков. В таблице 1. приводятся основные типы блоков и их назначение.

Таблица 1.

|  |  |
| --- | --- |
| Начало/конец | **Блок начала и конца**. Им начинается и заканчивается любая функция. Тип возвращаемого значения и аргументов функции обычно указывается в комментариях к блоку терминатора. |
| A, B, C | **Блок ввода и вывода**. Служит для организации ввода исходных данных и вывода результирующих данных. |
| C = A + B | **Функциональный блок.** Служит для указания действий (шага) алгоритма. Внутри записывается команда, которая должна быть выполнена. |
| Да  Нет  A < B | **Альтернативный блок** (условие, ветвление). Служит для указания выбора одного из двух возможных действий. Внутри ромба размещается условие выбора. |
| Подпрограмма | **Блок предопределённого процесса**. Используется для обращения к отдельным модулям и вспомогательным алгоритмам. |
| Цикл с параметром | **Блок цикла**. Предназначен для организации циклического процесса с параметром. Количество повторений (итераций) и шаг изменения параметра должно быть известно. |
| A | **Соединитель**. В случае, если блок-схема не умещается на лист, используется символ соединителя, отражающий переход потока управления между листами. |
| #ваш комментарий | **Комментарий** может быть соединен как с одним блоком, так и группой. Группа блоков выделяется на схеме пунктирной линией. |

**Пример решения задачи в блок-схеме.**

**Пример №1**

**Задание**: Алгоритм целочисленных преобразований представлен в виде фрагмента блок–схемы. Знаком := в нем обозначен оператор присваивания некоторого значения указанной переменной. Запись X := 1 означает, что переменная Х принимает значение 1.Определить результат работы алгоритма для исходных данных Х = 7, Y = 12.

Ввод X, Y

Нет

X < Y

Нет

Да

X > Y

X:=1

Y:=1

Да

X:=0

Y:=1

X:=1

Y:=0

**Пример №2**

**Задание**: Для приведенного ниже фрагмента блок–схемы алгоритма выбрать соответствующий ему фрагмент из вариантов ответа на алгоритмическом языке.

Да

Нет

Число отрицательное?

Разделить на 2

Умножить на -1

Извлечь корень

**Варианты ответов:**

1. **Если** число отрицательное

**То** разделить на 2

**Иначе** умножить на -1; извлечь корень

**Конец**

1. **Если** отрицательное

**То** умножить на -1

**Иначе** разделить на 2

**Конец**

1. **Если** отрицательное

**То** извлечь корень

**Иначе** разделить на 2

**Конец**

1. **Если** отрицательное

**То** умножить на -1; извлечь корень

**Иначе** разделить на 2

**Конец +**

**Практическая часть.**

**Варианты для выполнения первого задания.**

Номер вашего варианта соответствует номеру компьютера.

**Значения** применяются для первого вводимого числа.

|  |  |
| --- | --- |
| **Вариант:** | **Значения переменной:** |
| **1** | **7** |
| **2** | **-3** |
| **3** | **10** |
| **4** | **-8** |
| **5** | **4** |
| **6** | **11** |
| **7** | **-15** |
| **8** | **-7** |
| **9** | **9** |
| **10** | **-4** |
| **11** | **14** |
| **12** | **-20** |

**Задание №1.**

Найти результат выполнения работы алгоритма.

Где A – значение вашего варианта.

В ответе напишите получившееся число B.

Начало

-20

Да

Нет

A > 0

B = 3 \* 13

B = -2 \*( -20)

40

Конец

**Ответ: 40**

**Варианты для выполнения второго задания.**

|  |  |
| --- | --- |
| **Вариант:** | **Значения переменной:** |
| **1** |  |
| **2** |  |
| **3** |  |
| **4** |  |
| **5** |  |
| **6** |  |
| **7** |  |
| **8** |  |
| **9** |  |
| **10** |  |
| **11** |  |
| **12** |  |

**Задание №2**

Найти результат выполнения работы алгоритма.

Где S – значение вашего варианта.

В ответе напишите получившееся число N.

S1:=0

N:=0

Конец

Вывод N

Да

Нет

S1 < 12

12

Начало

N:=N+1

S1:=S1+1/N

**Ответ: 0**

**Задание №3**

Создать блок-схему исходя из условия: рассчитать площадь и периметр прямоугольника по двум известным сторонам (a,b).

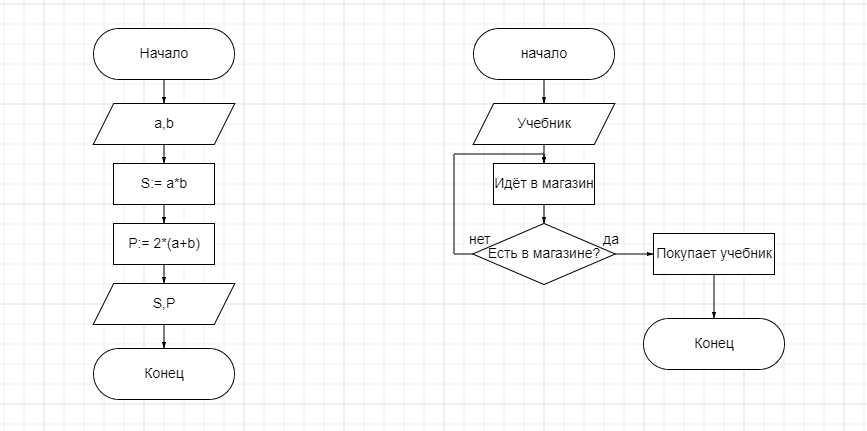
**Задание №4**

Создать блок-схему исходя из условия: студенту требуется купить учебник. Составить блок-схему описывающую действия студента в случае, если учебника нет в ряде магазинов.

**Задание №5**

**Контрольные вопросы.**

1. Что такое блок-схема?  
   Ответ: визуальное представление алгоритма при помощи блоков (геометрических фигур)
2. Как работает альтернативный блок?
3. Что такое входные и исходные данные?  
   Ответ: входные данные – это информация, которая передается в программу или алгоритм в качестве входа для обработки.  
   Исходные данные - это информация, которую следует обработать.
4. Опишите своими словами: Что значит выполнить алгоритм?   
   Ответ: это значит сделать каждый пункт от начала до конца, от первого пункта до последнего по очереди.

**Ответ на задания 3,4  
**