

Министерство образования Республики Беларусь
«ПОЛОЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет информационных технологий
Кафедра технологий программирования

**Методические указания для выполнения
лабораторной работы №2
по курсу «Конструирование программного
обеспечения»**

«Разработка схем алгоритмов программ»

Полоцк, 2022 г.

ЦЕЛЬ РАБОТЫ

Познакомится с такими понятиями как блок-схема и алгоритм, разобрать такие виды алгоритмов как: линейные, алгоритмы разветвленной структуры и циклические алгоритмы. На основе примеров, приведенных в данной лабораторной работе, спроектировать блок-схему информационной системы.

ТЕОРИТИЧЕСКИЕ СВЕДЕНИЯ

Блок-схема – это схематичное представление процесса, системы или компьютерного алгоритма. Блок-схемы часто применяются в разных сферах деятельности, чтобы документировать, изучать, планировать, совершенствовать и объяснять сложные процессы с помощью простых логичных диаграмм. Для построения блок-схем применяются прямоугольники, овалы, ромбы и некоторые другие фигуры (для обозначения конкретных операций), а также соединительные стрелки, которые указывают последовательность шагов или направление процесса.

Блок-схемы варьируются от незамысловатых, нарисованных вручную до подробных, составленных на компьютере диаграмм со множеством шагов и процессов. Если учесть все возможные вариации, блок-схемы можно признать одним из самых распространенных видов схем во всем мире. Они широко используются в разных сферах как технической, так и нетехнической направленности.

Иногда блок-схемы получают более узкоспециальные названия, например, схема процесса, схема рабочего процесса, функциональная блок-схема, моделирование бизнес-процессов, модель и нотация бизнес-процессов (BPMN) или схема технологического процесса (PFD). Они тесно связаны с другими распространенными видами схем, такими как диаграммы DFD и диаграммы активности на унифицированном языке моделирования (UML).

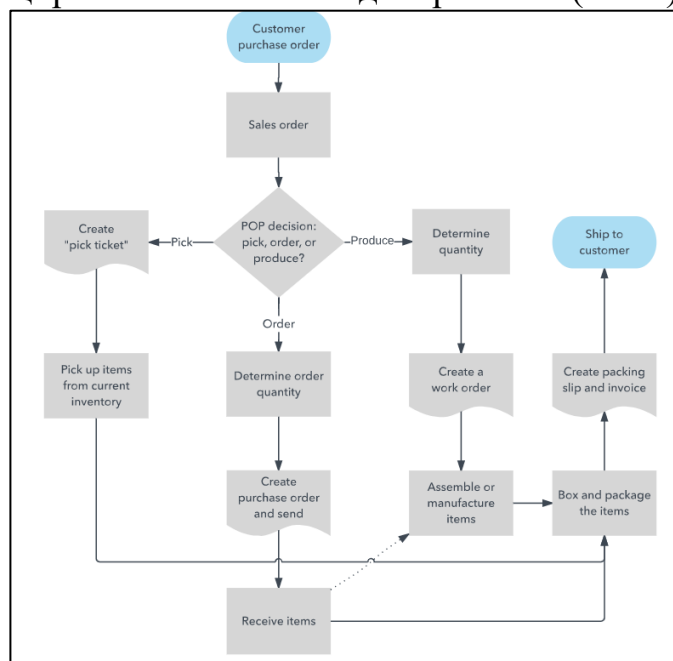


Рисунок 1.1 – Пример блок-схемы заказа товара


Алгоритм – это точная конечная система правил, определяющая содержание и порядок действий исполнителя над некоторыми объектами (исходными и промежуточными данными) для получения после конечного числа шагов искомого результата. Алгоритм может быть описан одним из трех способов:

- словесным;
- графическим (виде специальной блок-схемы);
- с помощью специальных языков программирования.

Алгоритмы делятся на:

1. **Линейный алгоритм** – это такой алгоритм, в котором все операции выполняются последовательно одна за другой.

Пример: составление алгоритма запуска программы Paint в ОС Windows.

Блок-схема	Порядок действий
 <pre> graph TD A[Начало] --> B[Меню Пуск] B --> C[Все программы] C --> D[Стандартные] D --> E[Paint] E --> F[Конец] </pre>	<ol style="list-style-type: none"> 1. Войти в меню «Пуск». 2. Войти в пункт «Все программы». 3. Войти в пункт «Стандартные». 4. Выбрать программу «Paint».

2. **Алгоритмы разветвленной структуры** применяются, когда в зависимости от некоторого условия необходимо выполнить либо одно, либо другое действие.

Пример: составление алгоритма для перехода дороги на светофоре.

Блок-схема	Порядок действий
<pre> graph TD Start([Начало]) --> Approach[Подойти к дороге] Approach --> Decision{Горит зеленый?} Decision -- Да --> Cross[Перейти] Decision -- Нет --> Wait[Дождаться зеленого света] Wait --> Cross Cross --> End([Конец]) </pre>	<ol style="list-style-type: none"> 1. Подойти к светофору. 2. Посмотреть на его свет. 3. Если горит зелёный, то перейти дорогу. 4. Если горит красный, то подождать, пока загорится зелёный, и уже тогда перейти дорогу.

Алгоритмы циклической структуры. Циклом называют повторение одних и тех же действий (шагов). Последовательность действий, которые повторяются в цикле, называют телом цикла. Циклические алгоритмы подразделяют на алгоритмы с предусловием, постусловием и алгоритмы с конечным числом повторов.

В алгоритмах с предусловием сначала выполняется проверка условия окончания цикла и затем, в зависимости от результата проверки, выполняется (или не выполняется) так называемое тело цикла.

Пример: составление алгоритма перевода чисел из десятичной системы в двоичную.



Блок-схема	Порядок действий
<pre> graph TD Start([Начало]) --> Decision1{Число a ≥ 2} Decision1 -- Да --> Divide[Делим a на 2] Divide --> Remainder[Остаток пишем в конец числа b] Remainder --> Replace[Число a заменяем на частное] Replace --> Decision2{a ≥ 2} Decision2 -- Да --> Divide Decision2 -- Нет --> Append[Дописываем a в начале числа b] Decision1 -- Нет --> Append Append --> Output[/b/] Output --> End([Конец]) </pre>	<ol style="list-style-type: none"> 1. Если число равно 0 или 1, то это и будет его двоичное представление. 2. Если число больше 1, то мы делим его на 2. 3. Полученный остаток от деления записываем в последний разряд двоичного представления числа. 4. Если полученное частное равно 1, то его дописываем в первый разряд двоичного представления числа и прекращаем вычисления. 5. Если же полученное частное больше 1, то мы заменяем исходное число на него и возвращаемся в пункт 2).

Общепринятые символы и способы нотации блок-схем

Не дайте широкому разнообразию фигур для блок-схем сбить вас с толку. У каждого символа есть свое значение и собственный контекст использования. Если в процессе создания блок-схемы вы вдруг запутаетесь в символах, помните, что в подавляющем большинстве случаев можно обойтись минимальным набором общепринятых символов, перечисленных ниже.

В блок-схемах чаще всего встречаются следующие фигуры и символы.

Символ	Название	Описание
	Процесс	Этот символ, также известный под названием «Действие», используется для обозначения процесса, действия или функции. Это самый распространенный символ в блок-схемах.
	Начало/конец	Данный символ, который иногда также именуют «Терминатором», применяется для обозначения начальной или конечной точки схемы или возможного результата того или иного пути развития процесса. Внутри блока, как правило, располагается слово «Начало» или «Конец».
	Документ	Символизирует ввод или вывод документа. Под вводом документа может подразумеваться поступление отчета, электронного письма или заказа. Примеры вывода документов: создание презентации, рабочего конспекта или письма.
	Решение	Символизирует вопрос, на который требуется ответ (как правило, «да/нет» или «истина/ложь»). На этом этапе блок-схема разветвляется в разных направлениях в зависимости от выбранного ответа и последующих блоков.
	Соединитель	Обычно применяется в более сложных схемах для соединения отдельных блоков в пределах одной страницы.
	Межстраничный соединитель	Часто применяется в сложных схемах для соединения отдельных блоков, расположенных на разных страницах. Для удобства интерпретации внутри фигуры, как правило, указывается номер страницы.

	<p>Ввод/вывод</p>	<p>Эта фигура, также известная под названием «Данные», символизирует данные, доступные для ввода или вывода, а также затраченные или полученные ресурсы. Хотя «Бумажная лента» также означает ввод/вывод данных, на сегодняшний день этот символ считается устаревшим и потому довольно редко используется в блок-схемах.</p>
	<p>Комментарий (примечание)</p>	<p>В сочетании с другими материалами этот символ позволяет добавить необходимый контекст, разъяснение или комментарий к определенному диапазону данных. Комментарий также можно присоединить к необходимому разделу блок-схемы с помощью пунктирной линии.</p>

ПРАКТИЧЕСКОЕ ЗАДАНИЕ

РАЗРАБОТКА СХЕМ АЛГОРИТМОВ ИНФОРМАЦИОННОЙ СИСТЕМЫ И ПРОЕКТИРОВАНИЕ БЛОК-СХЕМ

1. На основе выбранного варианта информационной системы, перечень которых представлен в документе в Google Classroom, разработать в Microsoft Visio блок-схемы работы информационной системы по 3-ём алгоритмам:

- линейному;
- алгоритму разветвленной структуры;
- алгоритму циклической структуры.

2. Оформить полученные блок-схемы в отчет по лабораторной работе и предоставить преподавателю на проверку.

!Содержание отчёта!

1. Ф.И.О., группа, название лабораторной работы.
2. Цель работы.
3. Описание проделанной работы.
4. Результаты выполнения лабораторной работы.
5. Выводы.