

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ЭКОНОМИЧЕСКИЙ ФАКУЛЬТЕТ

АЛГОРИТМИЗАЦИЯ

Методические указания для лабораторно-практических занятий и
самостоятельной работы студентов

Новосибирск 2017

УДК 004.42 (07)

ББК 32.973, я 7

А 456

Кафедра бухгалтерского учета и автоматизированной обработки информации

Составители: *Л.Г. Шишина, старший преподаватель кафедры БУ и АОИ*

Рецензенты: *О.Б. Кравченко доцент кафедры гражданского и гражданского процессуального права*

Алгоритмизация: методические указания для лабораторно-практических занятий и самостоятельной работы студентов / Новосиб. гос. аграр. ун-т; сост.: Л.Г. Шишина – Новосибирск, 2017. – 18 с.

Методические указания по теме «Алгоритмизация» предназначены для аудиторной и самостоятельной работы студентов, изучающих информатику и информационные технологии по всем направлениям подготовки всех форм обучения, а так же дисциплину «Проектирование программных средств сферы таможенных услуг» (направление подготовки Таможенное дело).

Методические указания обсуждены и одобрены на заседании кафедры бухгалтерского учета и автоматизированной обработки информации (протокол № 8 от «12» мая 2017 г.).

Методические указания утверждены и рекомендованы к изданию методической комиссией юридического факультета (протокол № 7 от «15» июня 2017 г.).

Оглавление

Введение	4
Раздел 1. Краткие сведения.....	5
Раздел 2. Практическая часть	14
Библиографический список	17

Введение

Основной целью дисциплин, изучающих информатику, информационные технологии и математические основы информационных технологий является формирование у студентов устойчивых знаний о способах и методах обработки информации за счет автоматизации решения практических задач.

В ходе изучения дисциплины «Проектирование программных средств сферы таможенных услуг» у студентов формируются умения систематизировать, обобщать информацию, представлять внутренние и внешние данные, разрабатывать алгоритмы обработки информации для решения задач и навыки обработки, сбора, хранения, анализа, преобразования данных с использованием программных средств. Полученные знания и умения необходимы для свободной ориентировки в информационной среде и дальнейшего профессионального самообразования в области компьютерной подготовки. Тема «Алгоритмизация» является основополагающей для дисциплины.

Знания и навыки, приобретенные при изучении темы, соответствуют общим требованиям подготовки студентов высшего учебного заведения.

Методические указания состоят из двух разделов:

1. Краткие сведения
2. Практическая часть

В разделе «Краткие сведения» разобраны этапы решения задач, основные структуры и виды алгоритмов. Во втором разделе предложены разноуровневые задания.

Методические указания могут быть использованы как для аудиторной, так и для самостоятельной работы студентов.

Раздел 1. Краткие сведения

Этапы решения задач на ЭВМ

- ▶ Постановка задачи. Четкое выделение требуемых результатов и исходных данных в задаче. На этом этапе всем величинам (известным и неизвестным) необходимо присвоить обозначения и идентификаторы. Определить условия решения задачи.
- ▶ Метод решения. Это связь между исходными данными и результатом. На этом этапе составляется математическая модель задачи, т.е. осуществляется описание задачи с помощью математических формул и уравнений.
- ▶ Сценарий. Это список сообщений, который будет выдавать на экран программный продукт, реализующий решаемую задачу.
- ▶ Разработка алгоритма.
- ▶ Реализация алгоритма в программном коде.
- ▶ Загрузка программного кода и исходных данных.
- ▶ Отладка.
- ▶ Решение задачи на компьютере.
- ▶ Анализ результатов решения.

Определение алгоритма

Алгоритм – точное предписание исполнителю совершить определенную последовательность действий для достижения поставленной цели за конечное число шагов, обеспечивающую получение требуемого результата из исходных данных.

Свойства алгоритма

Дискретность (прерывность, раздельность, конечность) – алгоритм должен представлять процесс решения задачи как последовательное выполнение простых (или ранее определенных) шагов. Каждое действие,

предусмотренное алгоритмом, выполняется только после того, как закончилось исполнение предыдущего.

• **Определенность** (детерминированность) – каждое правило алгоритма должно исключать произвольность толкования любого предписания. Благодаря этому свойству выполнение алгоритма носит механический характер и не требует никаких дополнительных указаний или сведений о решаемой задаче.

- ▶ **Результативность** (конечность) – алгоритм должен приводить к решению задачи за конечное число шагов.
- ▶ **Массовость** – алгоритм решения задачи разрабатывается в общем виде, то есть, он должен быть применим для некоторого класса задач, различающихся только исходными данными. При этом исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.

Схематично свойства алгоритма представлены на рис.1



Рис. 1. Свойства алгоритма

Способы задания алгоритмов

- ▶ Словесное задание – описывает алгоритм с помощью слов и предложений (пример: кулинарные рецепты).
- ▶ Табличный способ – представляет алгоритм в форме таблицы и расчетных формул (экономические расчеты).
- ▶ Графическое задание (язык блок-схем) – описывает каждый шаг алгоритма с помощью блоков, представленных в виде геометрических фигур, а последовательность выполнения шагов определяется линиями-связями, показывающими направление процесса обработки.

Основные элементы блок-схем представлены на рис. 2.

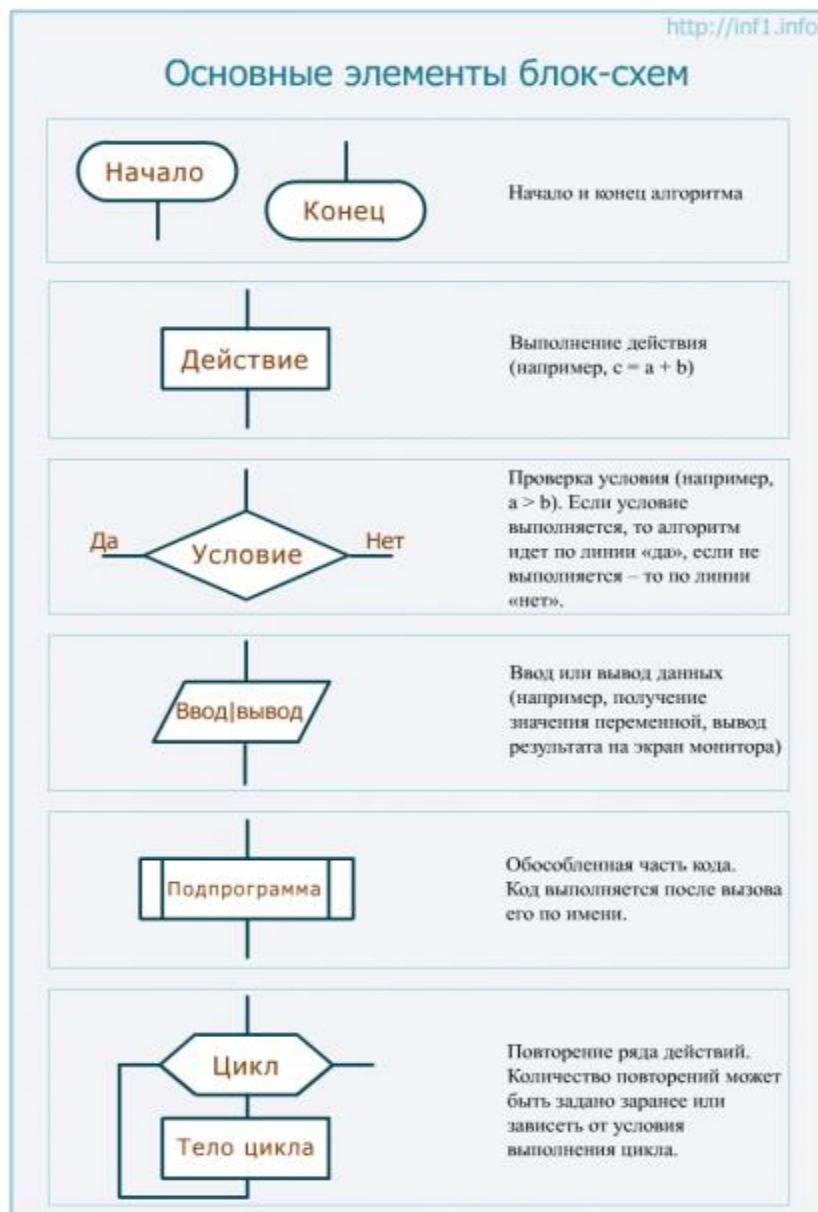


Рис. 2. Элементы блок-схем

Виды алгоритмов

- ▶ **Линейный алгоритм** – набор команд (указаний), выполняемых последовательно во времени друг за другом.
- ▶ **Разветвляющийся** алгоритм – алгоритм, содержащий хотя бы одно условие, в результате проверки которого ЭВМ обеспечивает переход на один из двух возможных вариантов решения шагов.

Алгоритмы этого класса можно разделить на два вида алгоритмов:

- структура «полное ветвление» задает одно условие (см. рис. 3);
- структура «выбор» позволяет задать несколько условий (см. рис.4);

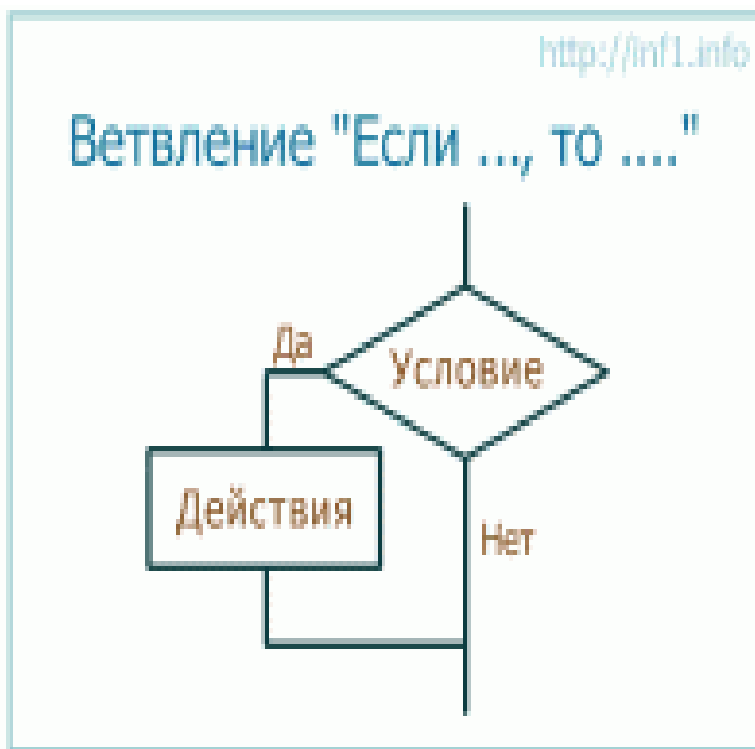


Рис. 3. Структура «Полное ветвление»

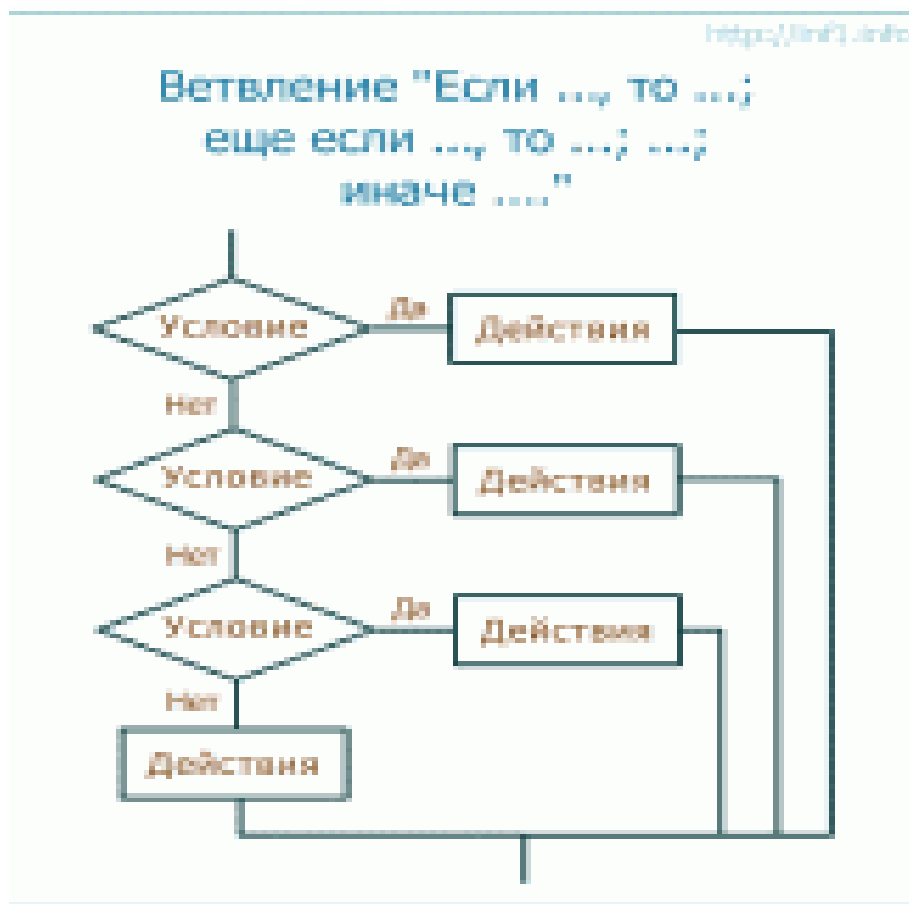


Рис. 4. Структура «Выбор»

Если условие выполняется (ИСТИНА), то выполнение алгоритма идет по ветке «Да». Если условие не выполняется (ЛОЖЬ), то выполнение идет по ветке «Нет».

Количество условий может быть различно. Если выполняется первое, то после выполнения действий, программа переходит к основной ветке, не проверяя дальнейшие условия. Если первое условие возвращает ложь, то проверяется второе условие. Если второе условие возвращает правду, то выполняются действия, включенные в вторую ветку конструкции. Последнее условие проверяется лишь в том случае, если ни одно из предыдущих не дало в результате ЛОЖЬ

- **Циклический алгоритм** — алгоритм, предусматривающий многократное повторение одного и того же действия (одних и тех же операций) над новыми исходными данными. К циклическим

алгоритмам сводится большинство методов вычислений, перебора вариантов. В циклическом алгоритме присутствуют следующие элементы:

- Телом цикла называется повторяемый участок программы.
- Параметром цикла называется переменная, которая при каждом новом входе в цикл принимает новое значение и следит за количеством повторений цикла.

Виды циклического алгоритма:

- Цикл с предусловием (см. рис. 5);

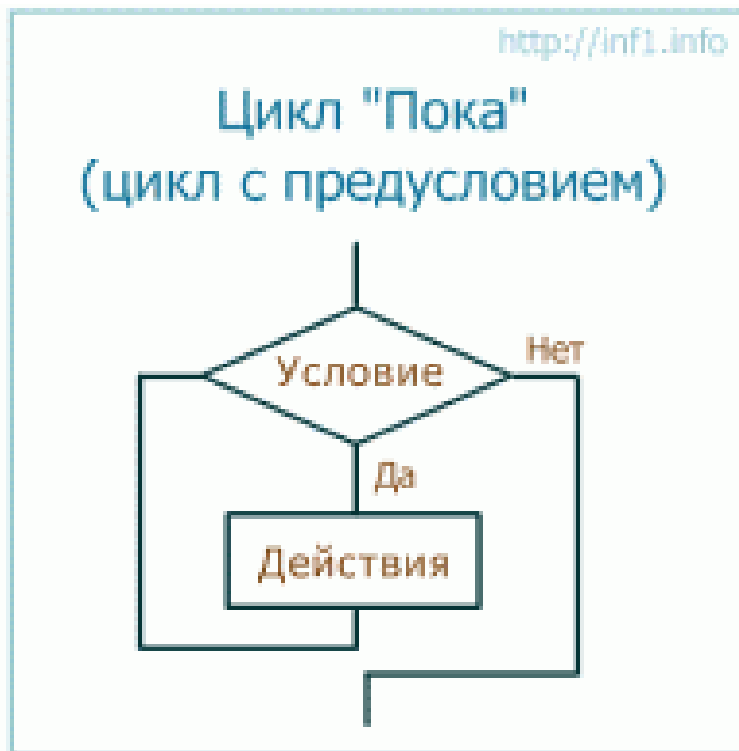


Рис. 5. Структура цикла с предусловием

В данном виде циклического алгоритма пока условие выполняется (результат логического выражения дает ИСТИНУ), будут выполняться действия тела цикла. В данном виде цикла не известно количество повторений. Выход из цикла осуществляется как только условие перестаёт выполняться (возвращает ЛОЖЬ).

- Цикл с постусловием (см. рис.6);

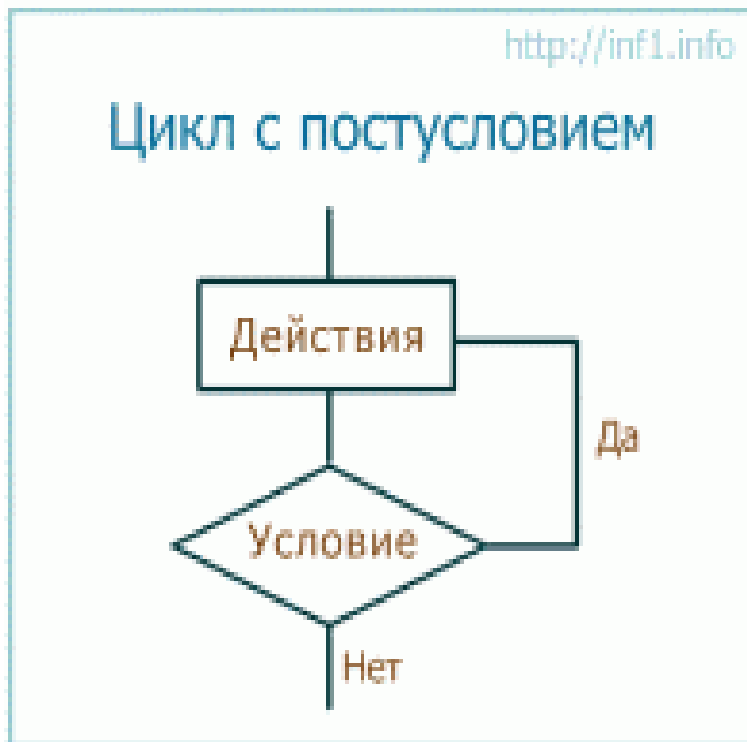


Рис.6. Структура цикла с постусловием

В этом цикле первый раз условие проверяется лишь после выполнения действий тела цикла. Если условие возвращает ИСТИНУ, то выражения-действия повторяются снова. Каким бы ни было условие, тело данного цикла хотя бы раз, но выполнится. Выход из цикла – когда условие возвращает ЛОЖЬ.

- Арифметический цикл (см. рис 7);



Рис. 7. Структура арифметического цикла

Данный цикл также называют циклом «Для» (for). По условию задачи известно, сколько раз нужно выполнить одну и ту же операцию. В его заголовке указывается три параметра: начальное значение переменной (от), конечно значение (до) и ее изменение с помощью арифметической операции на каждом «обороте» цикла (шаг).

- ▶ Алгоритм **со структурой вложенных циклов**, когда телом цикла является цикл. Глубина вложенности циклов ограничивается возможностями компьютера. Для подсчета количества повторений существуют: параметр внешнего цикла и параметр внутреннего цикла. При каждом изменении параметра внешнего цикла, значение параметра внутреннего цикла меняется от начального значения до конечного.
- ▶ **Смешанный** (комбинированный) алгоритм. Это алгоритм, в котором могут присутствовать все вышерассмотренные виды.

- ▶ **Вспомогательный** (подчиненный) алгоритм (процедура, подпрограмма) – алгоритм, отделенный от основной программы, выполняется лишь в случае их вызова из основной программы (из любого ее места). Одна и та же функция может вызываться из основной программы сколь угодно раз.

Раздел 2. Практическая часть

Задание 1. Сделать постановку задачи, определить структуру алгоритма и составить блок-схему для следующих задач:

1.1. $Y = A + BX^2$

1.2. $Y = \frac{A+X^2}{B}$

1.3. $Y = \frac{X^2}{AB}$

1.4. $Y = \lg X$

1.5. $AX^2 + BX + C = 0$

1.6. Даны два числа, определить наибольшее из них.

1.7. Даны координаты точки (X, Y), определить в какой четверти она лежит.

1.8. Решить систему уравнений:

$$\begin{cases} Y = \frac{1}{X+2}, X > 0 \\ Y = \frac{(X+B^2+C)^4}{C^2}, X \leq 0 \end{cases}$$

1.9. Решить систему уравнений:

$$\begin{cases} Y = 2 + X, X > 3 \\ Y = X^2, X < 0 \\ Y = X, X = 0 \\ Y = X - 2, 0 < X \leq 3 \end{cases}$$

1.10. Решить систему уравнений:

$$\begin{cases} Y = 2X, X > 2 \\ Y = X, X = 0 \\ Y = 2 + X, 1 < X \leq 2 \\ Y = X - 2, X < 0 \end{cases}$$

Задание 2. Сделать постановку задачи и составить блок-схему для циклического процесса:

2.1. Дан массив из N элементов. Необходимо вывести на экран элементы массива:

- a) в исходном порядке;
- b) в обратном порядке;
- c) только четных элементов;
- d) только нечетных элементов;
- e) только положительных элементов;
- f) только отрицательных элементов;
- g) элементов равных нулю (их порядковый номер);
- h) каждый третий элемент.

2.2. Вычислить значение функции: $Z = \frac{\sin X}{X}$, при $X = [0.1; 1]$, $\Delta X = 0.1$

2.3. Вычислить значение функции: $Y = \lg X^2$, для десяти различных значений аргумента.

Задание 3. Сделать постановку задачи, определить тип алгоритма и составить блок-схему:

3.1. Найти количество элементов массива размерностью N:

- a) всех;
- b) положительных;
- c) равных нулю;
- d) нечетных.

3.2. Найти сумму элементов массива размерностью N:

- a) всех;
- b) отрицательных;
- c) не равных нулю;
- d) четных.

3.3. Вычислить произведение элементов массива размерностью N:

- a) всех;

- b) кратных 2;
- c) не превышающих N;
- d) первого и последнего.

3.4. Найти факториал числа N.

3.5. Вычислить значение выражения: a^n , где $n > 0$

3.6. Вычислить значение выражения: $S = \sum_{i=1}^7 X_i * Y_i$

3.7. Вычислить значение выражения: $S = 1^2 + 2^2 + \dots + N^2$

3.8. а) На склад ежедневно привозят товар определенного наименования. График привоза указан в таблице 1. Необходимо составить отчет о наличии товара на складе: 1) на каждый день; 2) на конец недели.

б) На складе хранится товар определенного наименования в количестве 1000 у.е. Ежедневно со склада развозят некое количество товара заказчикам, график отгрузок представлен в таблице.1. Необходимо составить отчет о наличии товара на складе: 1) на каждый день; 2) на конец недели.

Таблица 1. График привозов и отгрузок на складе

День недели	Привоз, у.е.	Отгрузка, у.е.
Понедельник	100	120
Вторник	120	230
Среда	150	50
Четверг	200	180
Пятница	230	250
Суббота	180	70

3.9. Дан массив размерностью N. Найти наибольший элемент.

3.10. Дан массив размерностью N. Найти наименьший элемент.

3.11. Дан массив размерностью N. Расположить элементы массива в возрастающем порядке.

3.12. Дан массив размерностью N. Расположить элементы массива в убывающем порядке.

Библиографический список

Список основной литературы

1. Базовые и прикладные информационные технологии: Учебник / В.А. Гвоздева. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 384 с. (ЭБС)
2. Технология разработки программного обеспечения: Учеб.пос. / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул; Под ред. проф. Л.Г. Гагариной - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 400 с. (ЭБС)

Список дополнительной литературы

1. Введение в программирование на языке Visual C#: Учебное пособие / С.Р. Гуриков. - М.: Форум: НИЦ ИНФРА-М, 2013. - 448 с. (ЭБС)
2. Программирование на языке высокого уровня. Программирование на языке ObjectPascal: Учебное пособие / Т.И. Немцова; Под ред. Л.Г. Гагариной. - М.: ФОРУМ: ИНФРА-М, 2015. - 496 с. (ЭБС)
3. Прикладные информационные технологии: Учебное пособие / Е.Л. Федотова, Е.М. Портнов. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2013. – 336 с. (ЭБС)

Составитель
Шишина Лариса Георгиевна

АЛГОРИТМИЗАЦИЯ

Методические указания для лабораторно-практических занятий и
самостоятельной работы студентов

Авторская редакция
Компьютерная верстка *Л.Г. Шишина*