

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по учебной практике
Тема: Разработка визуализатора алгоритма Борувки

Студент гр. 7383	_____	Зуев Д.В.
Студентка гр. 7383	_____	Маркова А.В.
Студент гр. 7383	_____	Кирсанов А.Я.
Руководитель	_____	Жангиров Т.Р.

Санкт-Петербург
2019

ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ

Студент Зуев Д.В. группы 7383

Студентка Маркова А.В. группы 7383

Студент Кирсанов А.Я. группы 7383

Тема практики: Разработка визуализатора алгоритма Борувки

Задание на практику:

Командная итеративная разработка визуализатора алгоритма(ов) на Java с графическим интерфейсом.

Алгоритм: Алгоритм Борувки.

Сроки прохождения практики: 01.07.2019 – 14.07.2019

Дата сдачи отчета: 00.07.2019

Дата защиты отчета: 00.07.2019

Студент	_____	Зуев Д.В.
Студентка	_____	Маркова А.В.
Студент	_____	Кирсанов А.Я.
Руководитель	_____	Жангиров Т.Р.

АННОТАЦИЯ

Целью данной учебной практики является освоение навыков разработки графического пользовательского интерфейса, разработки алгоритма Борувки построения минимального остовного дерева графа и реализации связи между алгоритмом и интерфейсом на языке программирования Java. Для выполнения задания внутри команды происходит распределение ролей. Каждый член команды выполняет свое задание, потом разработчик ответственный за реализацию связи между графическим интерфейсом и алгоритмом собирает проект полностью.

SUMMARY

The purpose of this training practice is to master the skills of developing a graphical user interface, the development of Borowka algorithm for building a minimum spanning tree for the graph and the implementation of the connection between the algorithm and the interface in the Java programming language. To perform the task is the distribution of roles within the team. Each team member performs his task, then the developer responsible for the implementation of the connection between the graphical interface and the algorithm builds the project completely.

СОДЕРЖАНИЕ

	Введение	5
1.	Требования к программе	6
1.1.	Исходные требования к программе*	6
1.2.	Уточнение требований после сдачи прототипа	0
1.3.	Уточнение требований после сдачи 1-ой версии	0
1.4.	Уточнение требований после сдачи 2-ой версии	0
2.	План разработки и распределение ролей в бригаде	9
2.1.	План разработки	9
2.2.	Распределение ролей в бригаде	9
3.	Особенности реализации	0
3.1.	Использованные структуры данных	0
3.2.	Основные методы	0
3.3.		0
4.	Тестирование	0
4.1.	Тестирование графического интерфейса	0
4.2.	Тестирование кода алгоритма	0
4.3.	...	0
	Заключение	0
	Список использованных источников	0
	Приложение А. Исходный код – только в электронном виде	0

ВВЕДЕНИЕ

Целью данной учебной практики является освоение навыков разработки графического пользовательского интерфейса, разработки алгоритма Борувки построения минимального остовного дерева графа и реализации связи между алгоритмом и интерфейсом на языке программирования Java.

Задача состоит в написании программы, состоящей из классов реализующих граф, алгоритм, графический интерфейс и взаимодействие между ними.

Алгоритм Борувки (Алгоритм Борувки-Соллина) — это алгоритм нахождения минимального остовного дерева в графе.

Работа алгоритма состоит из нескольких итераций, каждая из которых состоит в последовательном добавлении рёбер к остовному лесу графа, до тех пор, пока лес не превратится в дерево, то есть, лес, состоящий из одной компоненты связности.

В псевдокоде, алгоритм можно описать так:

1. Изначально, пусть T — пустое множество рёбер (представляющее собой остовный лес, в который каждая вершина входит в качестве отдельного дерева).
2. Пока T не является деревом (что эквивалентно условию: пока число рёбер в T меньше, чем $V-1$, где V — число вершин в графе):
 - Для каждой компоненты связности (то есть, дерева в остовном лесе) в подграфе с рёбрами T , найдём самое дешёвое ребро, связывающее эту компоненту с некоторой другой компонентой связности. (Предполагается, что веса рёбер различны, или как-то дополнительно упорядочены так, чтобы всегда можно было найти единственное ребро с минимальным весом).
 - Добавим все найденные рёбра в множество T .
3. Полученное множество T является минимальным остовным деревом входного графа.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

1.1. Исходные Требования к программе

1.1.1 Требования к вводу исходных данных

Граф должен задаваться как набор ребер. Его можно вводить как в редакторе пользовательского интерфейса, так и считывать из файла. Ребро состоит из двух вершин целочисленного типа и веса вещественного типа. Новое ребро должно вводиться с новой строки.

1.1.2 Требования к визуализации

Графический интерфейс должен давать пользователю возможность вводить граф в редакторе или считывать граф из файла. Так же с помощью интерфейса пользователь должен иметь возможность пошагово проходить Алгоритм Борувки, возвращаться к предыдущему шагу выполнения алгоритма и сохранять результат работы алгоритма в файл.

На рисунке 1 представлен эскиз главного окна прототипа, открывающееся при запуске программы.

Главное окно содержит кнопку для открытия окна редактирования графа, кнопки управления работой алгоритма (шаг назад, шаг вперед, выполнить до конца), кнопку для сохранения результата в файл, кнопку для завершения работы (закрытия окна), окно, в котором отображается графическое представление графа и окно логов.

Кнопка «Редактирование графа» открывает окно редактирования графа. Эскиз окна представлен на рис. 2.

Кнопки шаг вперед и выполнить до конца инициализируют выполнение алгоритма. Пока алгоритм не инициализирован кнопки шаг вперед, шаг назад нажать нельзя. При пошаговом выполнении алгоритма редактирование графа будет запрещено. Сохранение результата будет доступно при полном выполнении алгоритма на конкретном графе.

В окне графа будет отображаться введенный граф до начала работы алгоритма, во время работы алгоритма в окне будут отображаться вершины и

ребра графа, которые добавлены в остовное дерево на данном шаге. Ребра и вершины, не добавленные на данном шаге, будут более тусклого цвета, чем добавленные. То же самое будет после окончания работы алгоритма. Это сделано для того чтобы не создавать лишнее окно для хранения начального графа.

В окне логов (окно для записи данных о работе программы) будет выводиться информация о пошаговой работе алгоритма (ребра, добавленные на текущем шаге), о создании и редактировании графа, о сохранении результата в файл, о успешной работе алгоритма.

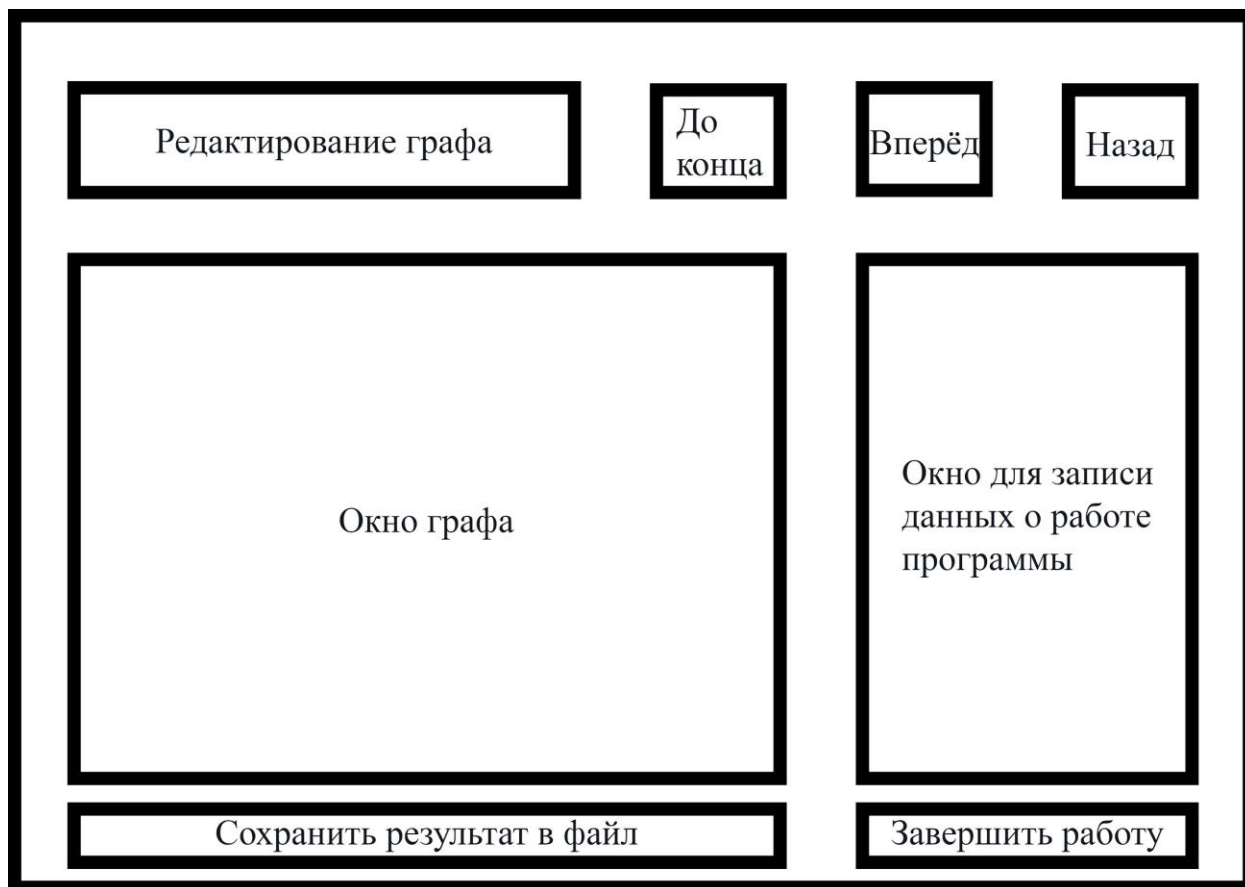


Рисунок 1 - Главное окно

В поле ввода вводится информация необходимая для редактирования файла. При вводе двух вершин и веса и нажатии кнопки “добавить” в граф добавляется ребро, это изменение отображается в списке ребер. При нажатии кнопки “удалить ребро” или кнопки “удалить вершину” появляется окно, где

пользователю предлагается выбрать из списка существующих ребер или вершин то, что нужно удалить. При нажатии кнопки “удалить граф”, граф введенный пользователем удаляется полностью. При нажатии кнопки “считать данные из файла”, анализируется путь до файла, записанный в окне ввода и при существовании файла по данному пути и соблюдении в нем формата ввода графа этот граф считывается. При нажатии кнопки “сохранить граф” граф, введенный пользователем или полученный изменением графа, введенного ранее, сохраняется, и окно редактирования закрывается.

Откуда:	Куда:	Вес ребра:
<input type="text"/>		

Окно для ввода данных

Добавить

Считать данные из файла

Удалить ребро

Удалить вершину

Удалить граф

Сохранить граф

Список добавленных рёбер

Рисунок 2 - Окно редактирования графа

На рис. 3,4,5 представлены UML диаграммы классов, реализующих графический интерфейс, работу алгоритма Борувки и взаимодействие пользователя с программой соответственно.

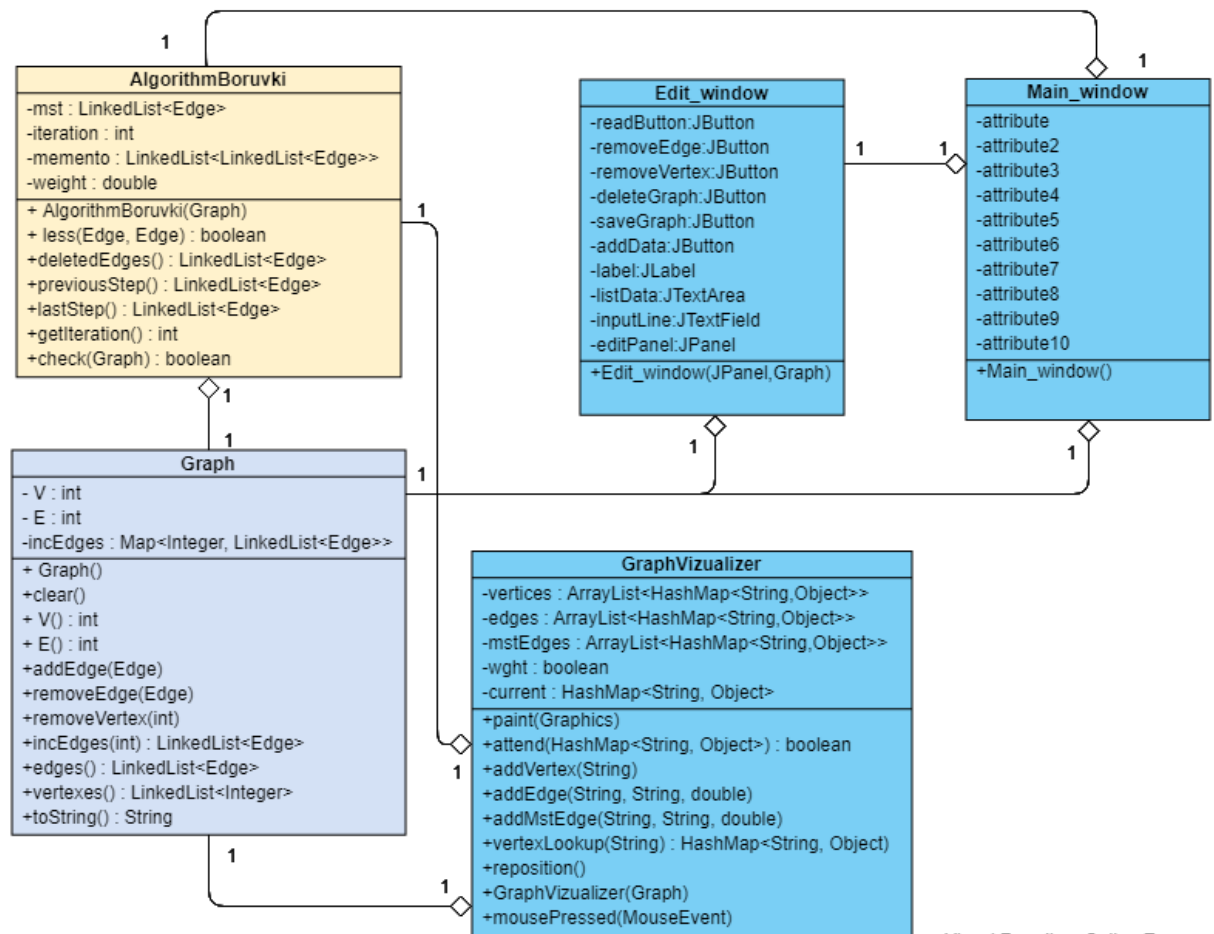


Рисунок 3 – UML диаграмма графического интерфейса

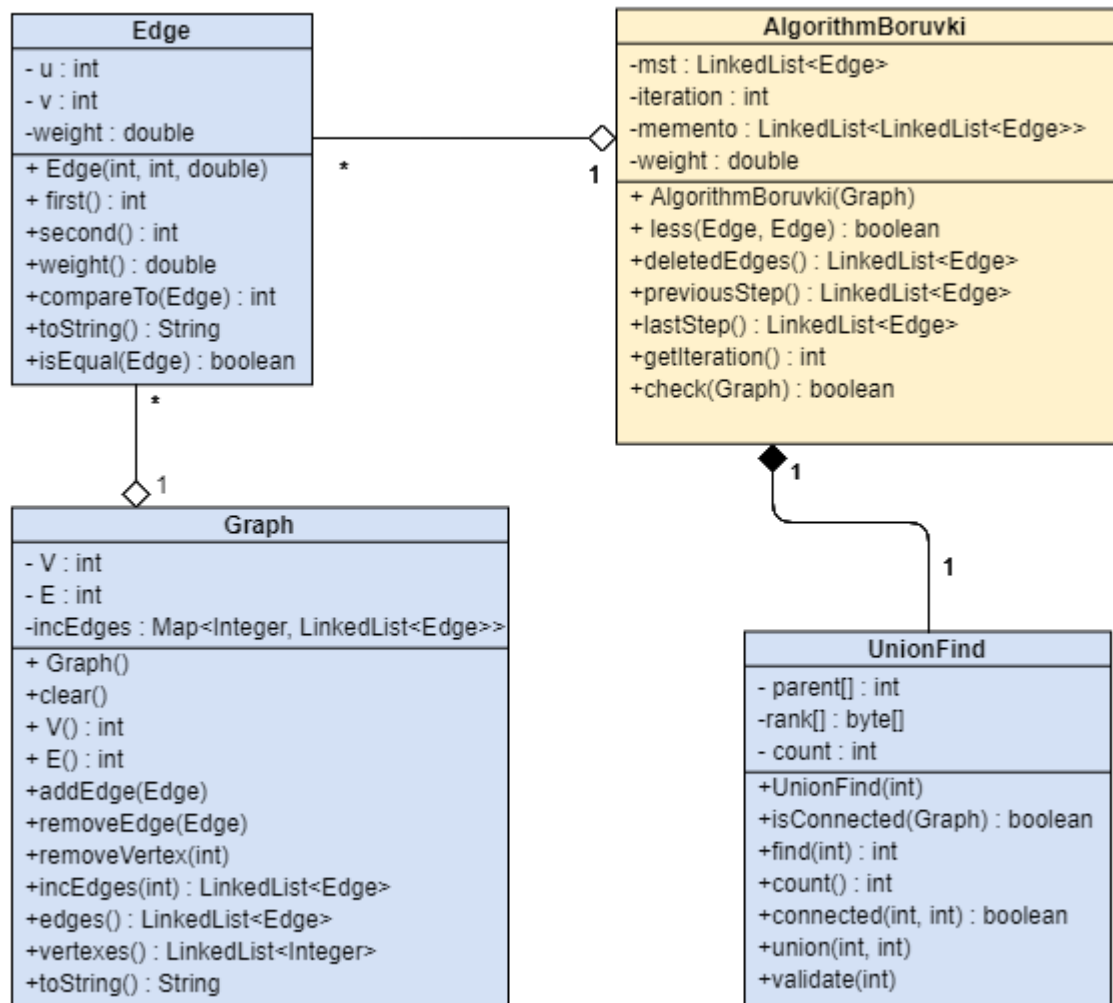


Рисунок 4 – UML диаграмма алгоритма Борувки

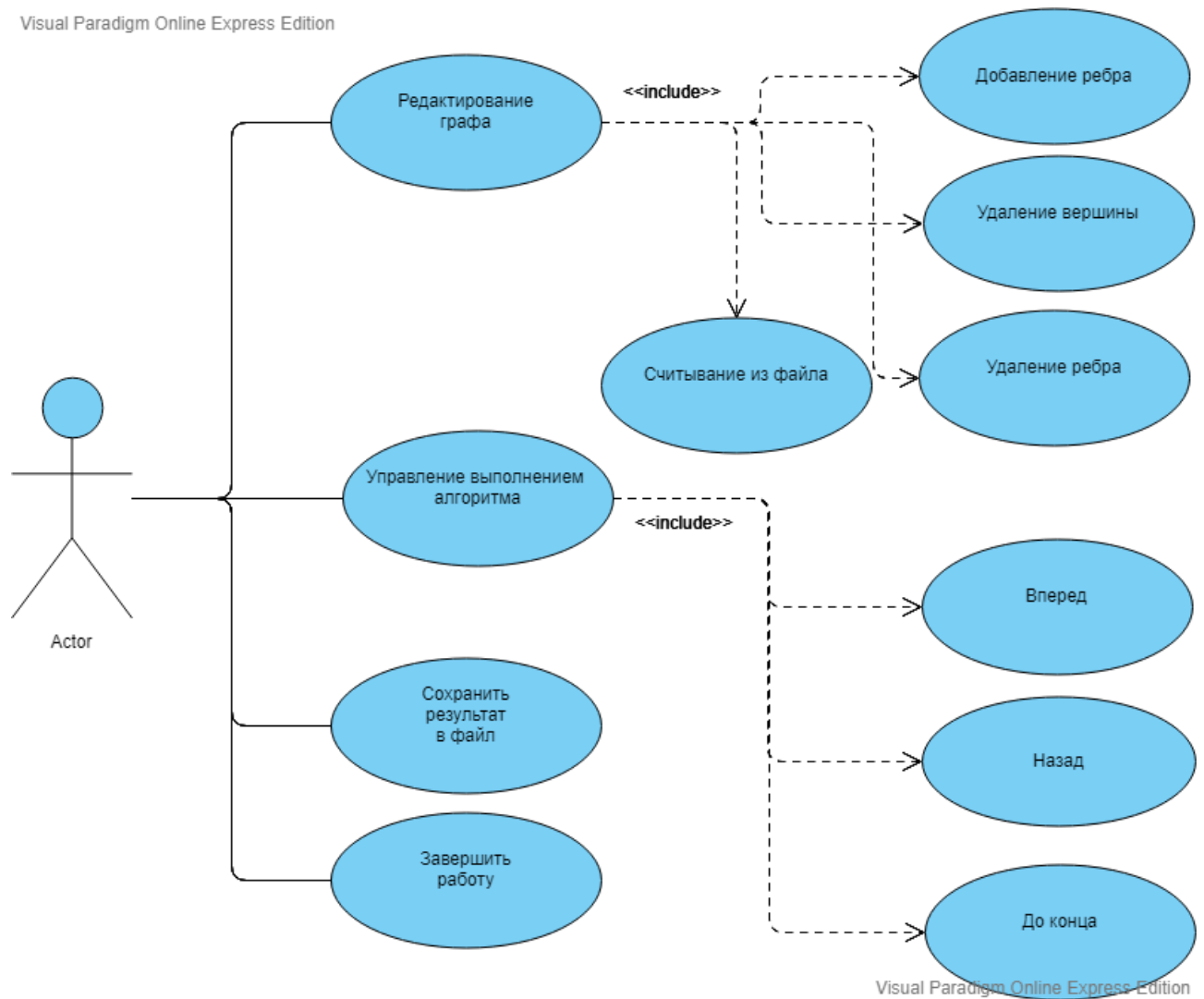


Рисунок 5 – UML диаграмма взаимодействия пользователя с программой

2. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

2.1. План разработки

Прототип:

Будут реализованы все окна программы. Все кнопки интерфейса будут реализовать только открытие или закрытие соответствующих им окон, остальные кнопки будут отключены.

Итерация 1 (08.07.2019):

Будет реализован ввод графа с клавиатуры. Будет реализовано окно логов, отображающее введенные ребра графа. Будет реализовано удаление ребер и вершин графа и удаление графа целиком.

Итерация 2 (10.07.2019):

Будет реализовано отображение графа и окно логов будет отображать данные о работе программы. Будет реализован алгоритм Борувки с пошаговым выполнением. Будет реализован ввод и вывод графа из файла.

2.2. Распределение ролей в бригаде

Зуев Даниил - ответственный за архитектуру проекта, реализация связи между алгоритмом и интерфейсом.

Маркова Ангелина - реализация графического интерфейса, визуализация.

Кирсанов Артём - разработка и реализация алгоритма и его последующее тестирование.

3. ТРЕТИЙ РАЗДЕЛ

3.1. Первый подраздел третьего раздела

3.2. Второй подраздел третьего раздела

ЗАКЛЮЧЕНИЕ

Кратко подвести итоги, проанализировать соответствие поставленной цели и полученного результата.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Ниже представлены примеры библиографического описания, В КАЧЕСТВЕ НАЗВАНИЯ ИСТОЧНИКА в примерах приводится вариант, в котором применяется то или иное библиографическое описание.

1. Иванов И. И. Книга одного-трех авторов. М.: Издательство, 2010. 000 с.
2. Книга четырех авторов / И. И. Иванов, П. П. Петров, С. С. Сидоров, В. В. Васильев. СПб.: Издательство, 2010. 000 с.
3. Книга пяти и более авторов / И. И. Иванов, П. П. Петров, С. С. Сидоров и др.. СПб.: Издательство, 2010. 000 с.
4. Описание книги под редакцией / под ред. И.И. Иванова СПб., Издательство, 2010. 000 с.
5. Иванов И.И. Описание учебного пособия и текста лекций: учеб. пособие. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2010. 000 с.
6. Описание методических указаний / сост.: И.И. Иванов, П.П. Петров. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2010. 000 с.
7. Иванов И.И. Описание статьи с одним-тремя авторами из журнала // Название журнала. 2010, вып. (№) 00. С. 000–000.
8. Описание статьи с четырьмя и более авторами из журнала / И. И. Иванов, П. П. Петров, С. С. Сидоров и др. // Название журнала. 2010, вып. (№) 00. С. 000–000.
9. Иванов И.И. Описание тезисов доклада с одним-тремя авторами / Название конференции: тез. докл. III международной науч.-техн. конф., СПб, 00–00 янв. 2000 г. / СПбГЭТУ «ЛЭТИ», СПб, 2010, С. 000–000.
10. Описание тезисов доклада с четырьмя и более авторами / И. И. Иванов, П. П. Петров, С. С. Сидоров и др. // Название конференции: тез. докл. III международной науч.-техн. конф., СПб, 00–00 янв. 2000 г. / СПбГЭТУ «ЛЭТИ», СПб, 2010, С. 000–000.
11. Описание электронного ресурса // Наименование сайта. URL: <http://east-front.narod.ru/memo/latchford.htm> (дата обращения: 00.00.2010).

12. ГОСТ 0.0–00. Описание стандартов. М.: Изд-во стандартов, 2010.
13. Пат. RU 000000000. Описание патентных документов / И. И. Иванов, П. П. Петров, С. С. Сидоров. Опубл. 00.00.2010. Бюл. № 00.
14. Иванов И.И. Описание авторефератов диссертаций: автореф. дисс. канд. техн. наук / СПбГЭТУ «ЛЭТИ», СПб, 2010.
15. Описание федерального закона: Федер. закон [принят Гос. Думой 00.00.2010] // Собрание законодательств РФ. 2010. № 00. Ст. 00. С. 000–000.
16. Описание федерального постановления: постановление Правительства Рос. Федерации от 00.00.2010 № 00000 // Опубликовавшее издание. 2010. № 0. С. 000–000.
17. Описание указа: указ Президента РФ от 00.00.2010 № 00 // Опубликовавшее издание. 2010. № 0. С. 000–000.

ПРИЛОЖЕНИЕ А
НАЗВАНИЕ ПРИЛОЖЕНИЯ

полный код программы должен быть в приложении, печатать его не надо