

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«КАЛИНИНГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»**

Л.Г. Высоцкий

ПРОГРАММНАЯ ИНЖЕНЕРИЯ (ПИ)

Учебно-методическое пособие по выполнению курсовой работы
для, обучающихся в бакалавриате по направлению
подготовки 09.03.03 Прикладная информатика (заочное отделение)

Калининград

Издательство ФГБОУ ВО «КГТУ»

2022

УДК 004.9(075)

Рецензент:

кандидат педагогических наук, доцент кафедры прикладной информатики
ФГБОУ ВО «Калининградский государственный технический
университет» Е. Ю. Заболотнова

Высоцкий, Л. Г.

Программная инженерия (ПИ): учебно-методическое пособие по выполнению курсовой работы для студентов заочного отделения, обучающихся в бакалавриате по направлению подготовки 09.03.03 Прикладная информатика /Л. Г. Высоцкий. – Калининград: Изд-во ФГБОУ ВО «КГТУ», 2022. – 41 с.

Учебно-методическое пособие включает подробные указания по выполнению курсовой работы по дисциплине «Программная инженерия» для студентов заочного отделения. Рассмотрены порядок выполнения и содержание каждого этапа разработки программного приложения. Приведен список вариантов заданий для курсовой работы. В пособие также включены подробные указания по оформлению пояснительной записки.

Учебно-методическое пособие рассмотрено и одобрено в качестве локального электронного методического материала кафедрой прикладной информатики института цифровых технологий ФГБОУ ВО «Калининградский государственный технический университет» 19 сентября 2022 г., протокол № 3

Учебно-методическое пособие рекомендовано к использованию в качестве локального электронного методического материала в учебном процессе методической комиссией ИЦТ 20 сентября 2022 г., протокол № 6

УДК 004.9(075)

© Федеральное государственное бюджетное образовательное учреждение высшего образования «Калининградский государственный технический университет», 2022 г.
© Высоцкий Л. Г., 2022 г.

СОДЕРЖАНИЕ

ЦЕЛИ И ЗАДАЧИ КУРСОВОЙ РАБОТЫ	4
ОБЩИЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ	5
УСЛОВИЯ ВЫБОРА ТЕМЫ И ПОРЯДОК ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ.....	8
ПРАВИЛА ОФОРМЛЕНИЯ ПРОГРАММНОЙ ДОКУМЕНТАЦИИ.....	25
ОРГАНИЗАЦИЯ ЗАЩИТЫ КУРСОВОЙ РАБОТЫ.....	27
Список рекомендуемой литературы.....	31
ПРИЛОЖЕНИЕ 1	33
ПРИЛОЖЕНИЕ 2	35
ПРИЛОЖЕНИЕ 3	36
ПРИЛОЖЕНИЕ 4	37

ЦЕЛИ И ЗАДАЧИ КУРСОВОЙ РАБОТЫ

Дисциплина «Программная инженерия» является завершающей для программистского цикла направления «Прикладная информатика» обучающихся в бакалавриате студентов. Поэтому основной целью выполнения данной курсовой работы является закрепление, углубление и обобщение знаний, полученных во время изучения всего программистского цикла данных направлений, т.е. дисциплин «Программирование», «Высокоуровневые технологии программирования» и «Программированная инженерия», а также двух учебных практик, ориентированных на расширение и закрепление знаний по программированию, полученных в ходе учебных занятий в виде лекций и лабораторных работ.

Курсовая работа позволяет:

- систематизировать, закрепить и расширить теоретические знания и практические навыки, применить их при решении конкретной задачи, связанной с будущей специальностью;
- развить навыки системного мышления через определение целей и постановку задачи курсовой работы;
- привить навыки самостоятельного решения вопросов, возникающих в ходе создания сложных программных систем;
- привить навыки самостоятельного освоения дополнительного технического материала по теме работы.
- повысить уровень подготовки по отдельным разделам дисциплины «Программная инженерия».

Курсовая работа по данной дисциплине должна подготовить студентов как к самостоятельному, так и к коллективному применению практических методик и инструментария, лежащих в основе промышленного создания программных продуктов, и, в дальнейшем, использовать их при подготовке выпускной квалификационной работы. Дополнительными задачами являются:

- приобретение навыков оформления печатной работы в соответствии с предъявленными требованиями;
- использование средств электронной презентации для доклада при защите работы.

Данная курсовая работа ориентирована, прежде всего, на освоение студентами практических навыков решения проблем, возникающих на этапе алгоритмизации стадии проектирования жизненного цикла программных систем (ЖЦ ПС). Очень многие задачи, решаемые в ходе работы над проектом ПС, могут реализовываться на базе разных алгоритмов, характеризующихся разным временем выполнения, разной точностью и другими параметрами. Следовательно, студент должен знать как можно больше алгоритмов, на основе которых можно решить каждую подлежащую автоматизации задачу, их характеристики и критерии, на основе которых происходит выбор. Поэтому в основу курсовой работы положено решение «задачи коммивояжера», которая широко применяется для выполнения различных оптимизационных расчетов, и для решения которой создано около десятка алгоритмов, характеризующихся разной трудоемкостью и временем решения.

В ходе формирования пояснительной записки студент должен дополнить и закрепить знания по подготовке технической документации, которые он получил к этому времени в ходе выполнения лабораторных и курсовых работ за предшествующий период обучения.

ОБЩИЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ

Курсовая работа является одним из видов научно-исследовательской деятельности студента, как одного из основных элементов процесса обучения, и должна соответствовать современному состоянию отечественной и зарубежной науки, а также быть направленной на достижение определенной цели.

При работе над ней следует:

- изучить и проанализировать литературу и другие источники (конспекты лекций по дисциплинам, отчеты по лабораторным и предшествующим курсовым работам программистского курса) по теме работы;
- четко уяснить суть индивидуального задания, специфику каждого используемого алгоритма, условия его реализации, характеристики получаемого результата;
- подготовить исходные данные для применяемых алгоритмов;
- выполнить расчет оптимального маршрута на основе каждого предложенного алгоритма;
- оформить пояснительную записку к курсовой работе в соответствии со стандартными требованиями;
- обобщить результаты и сформулировать выводы по сравнительным характеристикам использованных алгоритмов.

По объему курсовая работа должна занимать не менее 20—25 страниц печатного текста. Содержание курсовой работы должно свидетельствовать о достаточно высокой теоретической подготовке студента, которую он должен иметь на данном курсе, и о наличии у автора необходимых знаний по теме работы. Текст пояснительной записки должен быть написан грамотно, хорошим литературным и профессиональным языком, Работа должна иметь правильно составленную библиографию, логичную структуру, обеспечивающую наиболее эффективно решение проблемы. Пояснительная записка, как и любая техническая документация, пишется в обезличенной форме, то есть вместо «я считаю» - «считается» и так далее. В курсовой работе должны присутствовать: титульный лист, задание на курсовую работу, содержание, аннотация (реферат), основная часть, заключение, список использованных источников, список приложений.

Титульный лист оформляется в соответствии с примером, приведенным в Приложении 1.

Содержание – содержание включает наименование всех разделов и подразделов, пунктов, списка литературы и приложений с указанием номеров страниц. Аннотация и содержание в оглавлении не указываются. Подразделы и пункты указываются с отступами вправо. Названия всех разделов должны точно соответствовать тексту курсовой работы. Обязательно указываются страницы, с которых начинаются все пункты. Образец листа содержания приведен в Приложении 2.

Аннотация (реферат) – сокращенное описание работы, оформляется в соответствии с примером, приведенным в Приложении 3.

Постановка задачи – подробное описание задания с указанием требуемых манипуляций с информацией и получаемых конечных результатов.

Представление всех промежуточных и конечных результатов выполнения курсовой работы. Результаты представляются в табличном или графическом виде в соответствии с ходом методических указаний по выполнению курсовой работы, изложенных ниже. Все графические построения выполняются с соблюдением одинаковых масштабов по осям X и Y. Все принятые решения обосновываются.

Заключение содержит четко сформулированные итоги проделанной работы и основные выводы, к которым пришел автор. Заключение должно быть кратким, обстоятельным и соответствовать цели и поставленным в начале задачам, т.е. выявлению основных достоинств и недостатков использованных в курсовой работе алгоритмов.

Список использованных источников представляет собой перечень использованных при написании курсовой работы разнообразных источников. Объем списка должен быть достаточным и обычно составляет не менее 10 единиц. Работа с литературой не ограничивается чтением учебников и учебных пособий. В настоящее время Интернет-ресурсы дают возможность познакомиться с самыми новыми тенденциями и процессами. В тексте работы должны быть обязательно

приведены ссылки на использованные источники с указанием даты последнего обращения.

Список использованной литературы должен соответствовать современным требованиям по ГОСТу 2019 г.

Приложения к курсовой работе оформляются на отдельных листах, причем каждое приложение должно иметь свой тематический заголовок и номер, который пишется в правом верхнем углу, например: «Приложение I». В приложениях представлены формы, которые содержат большой цифровой и фактический материал, как правило, первичный, но дополняющий основной текст работы. Например, громоздкие подробные таблицы. Каждое приложение должно начинаться с новой страницы и иметь заголовок. Приложения нумеруются арабскими цифрами порядковой нумерации. На все Приложения в основной части курсовой работы должны быть ссылки.

УСЛОВИЯ ВЫБОРА ТЕМЫ И ПОРЯДОК ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Выбор конкретного алгоритма для решения программируемой задачи в рамках создаваемой программной системы (ПС) обусловлен многими факторами, этот выбор, в общем случае, невозможно формализовать, сделать однозначным, поскольку для решения одной и той же задачи часто можно использовать разные алгоритмы, которые отличаются своими характеристиками. К настоящему времени программная инженерия накопила сотни типовых алгоритмов для решения того или иного класса задач, многие из которых описаны в известном каждому практикующему программисту четырехтомнике Кнута (Кнут Д.Е. Искусство программирования для ЭВМ. Т. 1-4. – М.: Мир, 1976-1978). Вот почему на этапе алгоритмизации жизненного цикла программной системы решающую роль продолжают играть предшествующий опыт и интуиция разработчиков. Однако, используются и численные критерии при выборе конкретного алгоритма. К ним относят-

ся, прежде всего, время выполнения и точность решения задачи. Специальная нотация « O -большое» описывает скорость работы алгоритма. При этом « O -большое» не оценивает скорость в секундах, а позволяет сравнить количество операций у разных алгоритмов на выполнения одной и той же задачи. Причём для худшего возможного случая. Например:

Простой поиск (перебор) в списке из n неотсортированных записей характеризуется временем $O(n)$. Двоичный (бинарный) поиск в этом же отсортированном списке определяется уже временем $O(\log n)$.

К настоящему времени сформирован целый набор обобщенных типов алгоритмов (методов), способов их реализации и методов построения, которые помогают разработчику в его работе, но требует обоснованного выбора в конкретной ситуации для каждой задачи. В качестве примера именно такой ситуации в курсовой работе предлагается рассмотреть возможные варианты решения «задачи коммивояжера».

Задача имеет *недетерминированное* решение в том случае, если оно выбирается из заранее определенного множества вариантов. По сути, недетерминированный алгоритм описывает систематическую процедуру поиска нужного решения среди всех возможных. В свою очередь, этот алгоритм насчитывает к настоящему времени достаточно много конкретных реализаций: метод простого перебора, метод ветвей и границ, метод ближайшей вершины и т.д. Все эти методы применимы к решению «задачи коммивояжера», которая заключается в поиске оптимального (выгодного по определенному критерию) маршрута в некоторой сети. Задача стала актуальной в тридцатые годы XX века в США (время «великой депрессии»), когда тысячи коммивояжеров колесили по стране с товаром и перед ними возникала проблема оптимизации маршрута (минимизации стоимости, расстояния или времени), при условии, что каждую вершину транспортной сети надо посетить один и только один раз. В настоящее время методы ее решения активно используются при нахождении оптимальных вариантов в различных сферах чело-

веческой деятельности: транспортировке грузов, прокладке межсоединений в ЛВС, определении траектории движения режущего инструмента станка с ЧПУ и др. Т.е. они являются базовыми алгоритмами функционирования многих ПС.

Процесс выполнения курсовой работы предполагает следующую очередность этапов:

- Ознакомится с теоретическим разделом (выделен более тёмным фоном ниже), объясняющим ход решения задачи курсовой работы.

Простейший вариант решения указанной задачи – алгоритм на основе простого перебора, т.е. последовательного просмотра всех возможных маршрутов и выбора из них оптимального. Для упрощения будем считать, что ищется маршрут

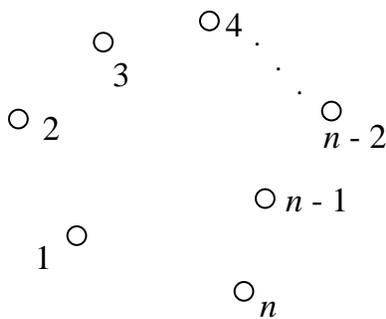


Рис. 1

минимальной длины. Пусть дано n узлов сети (рис. 1). Анализ рисунка показывает, что существует $(n-1)!$ возможных реализаций межсоединений, среди которых одно или несколько - оптимальные. Общая постановка задачи может быть сформулирована следующим образом.

Расстояния между узлами сети можно представить в виде матрицы D размерностью $n \times n$ с неотрицательными элементами d_{ij} . Элемент d_{ij} , расположенный в i -й строке и j -м столбце, соответствует расстоянию от узла i до узла j (т.е. матрица D , в общем случае, симметрична относительно главной диагонали). Порядок соединения узлов представляется маршрутом $T_r \in T$, где T - множество допустимых маршрутов мощностью $(n-1)!$, а T_r – конкретный кортеж из n номеров узлов.

$$T_r = \langle q_1, q_2, \dots, q_k, \dots, q_n, q_1 \rangle, \text{ где } q_k = \overline{1, n}; q_k \neq q_l.$$

Каждый допустимый маршрут представляет цикл в графе полной связности из n вершин, включающий все узлы только один раз (рис. 2). Каждая упорядоченная пара $\langle q_k, q_l \rangle$ является дугой (звеном) маршрута. Длина $Z(T_r)$ маршрута T_r равна сумме соответствующих элементов матрицы расстояний

$$Z(T_r) = \sum_{q_i q_j \in T_r} d_{ij} .$$

Задача состоит в определении такого $T^0 \in T$, при котором $Z(T^0)$ принимает минимальное значение.

Проиллюстрируем метод полного перебора на примере из пяти вершин, координаты которых: $q_1 - (61, 96)$; $q_2 - (48, 45)$; $q_3 - (91, 95)$; $q_4 - (80, 16)$; $q_5 - (13, 16)$.

Табл. 1 представляет матрицу расстояний между узлами.

Таблица 1

В таблице 2 представлены все 24 маршрута, существующие в данной сети. Из таблицы следует, что самым коротким является маршрут $T_6 = \langle \mathbf{125431} \rangle$ длиной $Z(T_6) = 274.85$.

	q_2	q_3	q_4	q_5
q_1	52.63	30.01	82.23	93.3
q_2		65.95	43.19	45.45
q_3			79.76	111.02
q_4				67

Из таблицы следует, что для полного перебора маршрутов необходимо сделать $K = \mathbf{84}$ вычислений (сложений).

Из этой же таблицы видно, что маршрут $T_{10} = \langle \mathbf{134521} \rangle$ также характеризуется минимальной длительностью, что естественно, так как он является обратным для маршрута $T_6 = \langle \mathbf{125431} \rangle$. Очевидно, что для полносвязанного графа, у которого $\langle q_i q_j \rangle = \langle q_j q_i \rangle$, количество анализируемых маршрутов уменьшается в два раза, т.е. определяется как $(n - 1)!/2$. С учетом данного обстоятельства табл. 2 преобразуется в таблицу 3, в которой число маршрутов уменьшено в два раза. Граф маршрутов для данной таблицы представлен на рис. 2. Теперь уже количество вычислений уменьшилось в два раза и составляет $K = \mathbf{42}$.

Даже при относительно небольшом количестве вершин общее число маршрутов, а следовательно, и объем вычислений, может быть значительным. Так, например, для только 10 вершин полносвязанного графа число маршрутов равняется

$$N = 1 * 3 * 4 * 5 * 6 * 7 * 8 * 9 = 181440.$$

Таблица 2

Маршруты	Уровни				
	I	II	III	IV	V
123451	52.63	118.58	198.34	265.34	358.64
123541	52.63	118.58	229.6	296.6	378.83
124351	52.63	95.82	175.58	286.6	379.9
124531	52.63	95.82	162.82	273.84	303.85
125341	52.63	98.08	209.1	288.86	371.09
125431	52.63	98.08	165.08	244.84	274.85
132451	30.01	95.96	139.15	206.15	299.45
132541	30.01	95.96	141.41	208.41	290.64
134251	30.01	109.77	152.96	198.41	291.71
134521	30.01	109.77	176.77	222.22	274.85
135241	30.01	141.03	186.48	229.67	311.9
135421	30.01	141.03	208.03	251.22	303.85
142351	82.23	125.42	191.37	302.39	395.69
142531	82.23	125.42	170.87	281.89	311.9
143251	82.23	161.99	227.94	273.39	366.69
143521	82.23	161.99	273.01	318.46	371.09
145231	82.23	149.23	194.68	260.63	290.64
145321	82.23	149.23	260.25	326.2	378.83
152341	93.3	138.75	204.7	284.46	366.69
152431	93.3	138.75	181.94	261.7	291.71
153241	93.3	204.32	270.27	313.46	395.69
153421	93.3	204.32	284.08	327.27	379.9
154231	93.3	160.3	203.49	269.44	299.45
154321	93.3	160.3	240.06	306.01	358.64

Если трудоемкость поиска оптимального решения слишком велика или для текущей задачи достаточно квазиоптимального решения, то используют методы уменьшения вычислений, например, нахождение квазиоптимального маршрута на основе *ближайшей вершины*. Его суть – от текущей вершины путь прокладывается через ближайшую свободную вершину сети. Для текущего примера, начиная с вершины 1 имеем: к узлу 1 ближе остальных узел 3 ($d_{13}=30.01$), из оставшихся вершин к узлу 3 ближе всех узел 2 ($d_{32}=65.95$), а к узлу 2 узел 4 ближе, чем узел 5 ($d_{24}=43.19$, $d_{25}=45.45$). Далее следуют узлы 5 и 1. Длина $Z\langle\mathbf{132451}\rangle = 30.01 + 65.95 + 43.19 + 67 + 93.3 = 299.45$.

Очевидно, что выбор начальной вершины для алгоритма на основе ближайшей влияет на последовательность самого маршрута и его длину.

Таблица 3

Маршруты	Уровни				
	I	II	III	IV	V
123451	52.63	118.58	198.34	265.34	358.64
123541	52.63	118.58	229.6	296.6	378.83
124351	52.63	95.82	175.58	286.6	379.9
124531	52.63	95.82	162.82	273.84	303.85
125341	52.63	98.08	209.1	288.86	371.09
125431	52.63	98.08	165.08	244.84	274.85
132451	30.01	95.96	139.15	206.15	299.45
132541	30.01	95.96	141.41	208.41	290.64
134251	30.01	109.77	152.96	198.41	291.71
135241	30.01	141.03	186.48	229.67	311.9
142351	82.23	125.42	191.37	302.39	303.85
143251	82.23	161.99	227.94	273.39	366.69

Дальнейшим развитием метода перебора явился метод *поиска с возвратом* (метод *ветвей и границ*), который, в свою очередь, является частным случаем более общего метода проб и ошибок. Его можно проиллюстрировать в виде уже рассмотренного ранее дерева множества возможных вариантов (рис. 2), на котором осуществляется нахождение кратчайшего пути поиском в ширину с отбрасыванием лишних вариантов, т.е. они до конца не просматриваются, если превышают самый короткий из уже пройденных маршрутов. При этом объем вычислений может быть значительно снижен эффективным подбором начального для сравнения маршрута, например, методом ближайшей вершины. Математически метод формулируется таким образом:

Вначале по некоторому правилу выбирается какой-либо вариант, соответствующий полному пути на древовидном графе решений от корневой до висячей вершины. При этом оценивается соответствующая ему длина, которая становится текущей оптимальной $T^{0г}$.

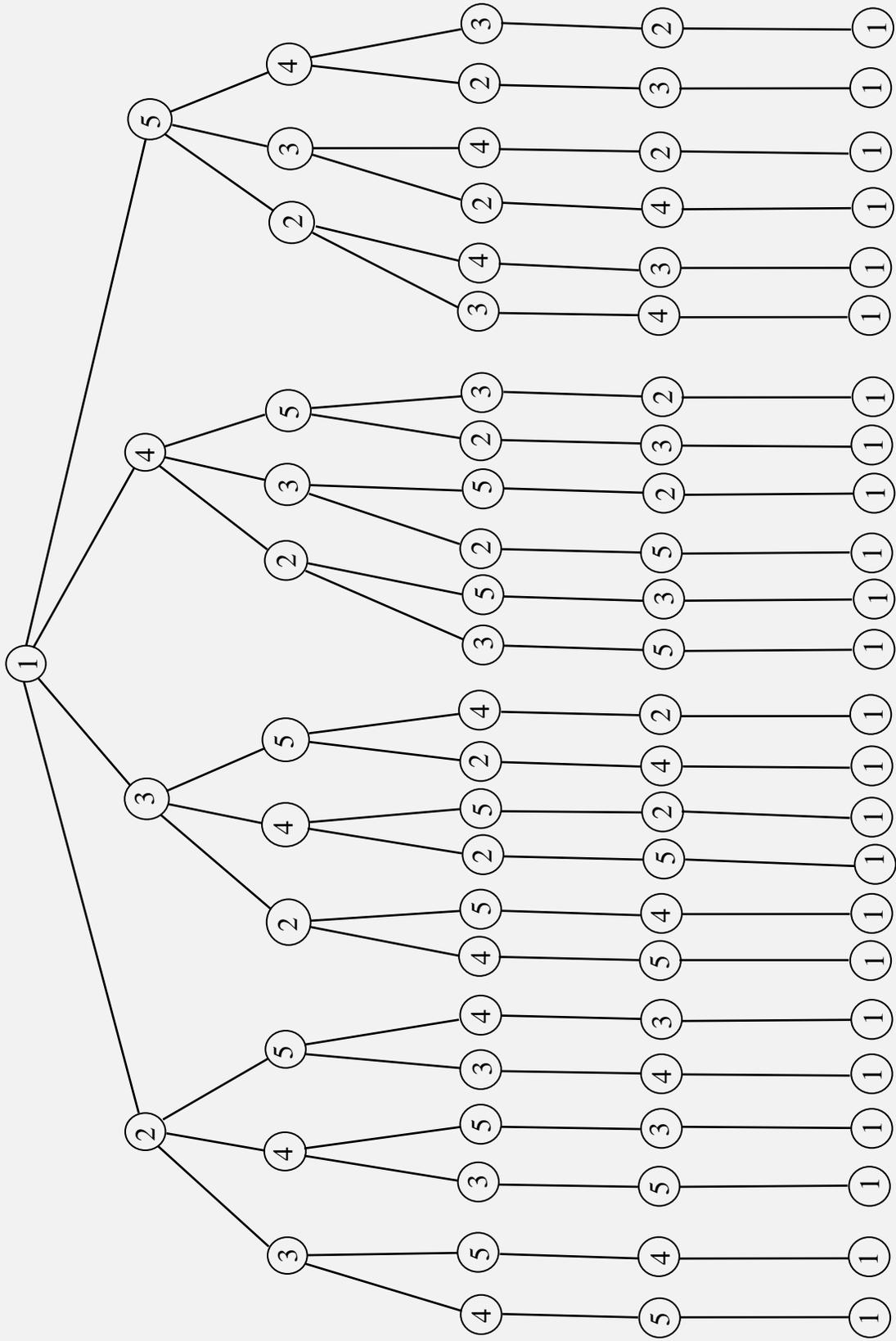


Рис. 2

Далее производится последовательное вычисление длин других маршрутов. Если текущее значение длины очередного маршрута уже превосходит T^{0T} , то движение по нему прекращается и вся соответствующая ему ветвь подмножества маршрутов отсекается. Чем ближе к корневой вершине дерева отсекается ветвь, тем эффективнее уменьшается число просматриваемых вариантов. Если же длина полностью просмотренного маршрута меньше T^{0T} , то она становится текущим значением T^{0T} . Рассмотрим реализацию данного метода на примере табл. 3. Результат работы алгоритма представлен в табл. 4.

Таблица 4

Маршруты	Уровни				
	I	II	III	IV	V
123451	52.63	118.58	198.34	265.34	358.64
123541	52.63	118.58	229.6	296.6	378.83
124351	52.63	95.82	175.58	286.6	379.9
124531	52.63	95.82	162.82	273.84	303.85
125341	52.63	98.08	209.1	288.86	371.09
125431	52.63	98.08	165.08	244.84	274.85
132451	30.01	95.96	139.15	206.15	299.45
132541	30.01	95.96	141.41	208.41	290.64
134251	30.01	109.77	152.96	198.41	291.71
135241	30.01	141.03	186.48	229.67	311.9
142351	82.23	125.42	191.37	302.39	
143251	82.23	161.99	227.94	273.39	366.69

Исходно в качестве текущего оптимального T^{0T} выбирается первый маршрут таблицы маршрута <123451>. Его длина $Z<123451> = 358.64$. Маршрут <124531> характеризуется длиной $303.85 < 358.64$, поэтому он становится текущим оптимальным до маршрута <125431>, который имеет ещё меньшую длину 274.85. В дальнейшем значение T^{0T} не меняется, т.е. именно этот маршрут является самым коротким. Длина маршрута <142351> после четвертого звена становится равной $302.39 > 274.85$, поэтому не имеет смысла в дальнейшем просчете длины. Таким образом, хоть и незначительно, число вычислений сократится. Если бы поиск производился по полной таблице перебора (табл. 5), то результат был бы еще бо-

лее значимым, поскольку аналогичная ситуация имеет место уже для восьми маршрутов. Итоговое дерево анализа для этого случая представлено на рис. 3.

Таблица 5

Маршруты	Уровни				
	I	II	III	IV	V
123451	52.63	118.58	198.34	265.34	358.64
123541	52.63	118.58	229.6	296.6	378.83
124351	52.63	95.82	175.58	286.6	379.9
124531	52.63	95.82	162.82	273.84	303.85
125341	52.63	98.08	209.1	288.86	371.09
125431	52.63	98.08	165.08	244.84	274.85
132451	30.01	95.96	139.15	206.15	299.45
132541	30.01	95.96	141.41	208.41	290.64
134251	30.01	109.77	152.96	198.41	291.71
134521	30.01	109.77	176.77	222.22	274.85
135241	30.01	141.03	186.48	229.67	311.9
135421	30.01	141.03	208.03	251.22	303.85
142351	82.23	125.42	191.37	302.39	
142531	82.23	125.42	170.87	281.89	
143251	82.23	161.99	227.94	273.39	366.69
143521	82.23	161.99	273.01	318.46	
145231	82.23	149.23	194.68	260.63	290.64
145321	82.23	149.23	260.25	326.2	
152341	93.3	138.75	204.7	284.46	
152431	93.3	138.75	181.94	261.7	291.71
153241	93.3	204.32	270.27	313.46	
153421	93.3	204.32	284.08		
154231	93.3	160.3	203.49	269.44	299.45
154321	93.3	160.3	240.06	306.01	

Эффективность метода ветвей и границ можно с помощью коэффициента уменьшения вычислений (КУВ)

$$K_{\text{ув}} = \frac{\text{число проведенных вычислений}}{\text{полное число вычислений}}. \quad (1)$$

Для рассматриваемого случая (с полным перебором) данный коэффициент определится как

$$K_{\text{уб}} = \frac{75}{84} = 0.89$$

Чем меньше значение данного коэффициента, тем успешней был использован метод ветвей и границ.

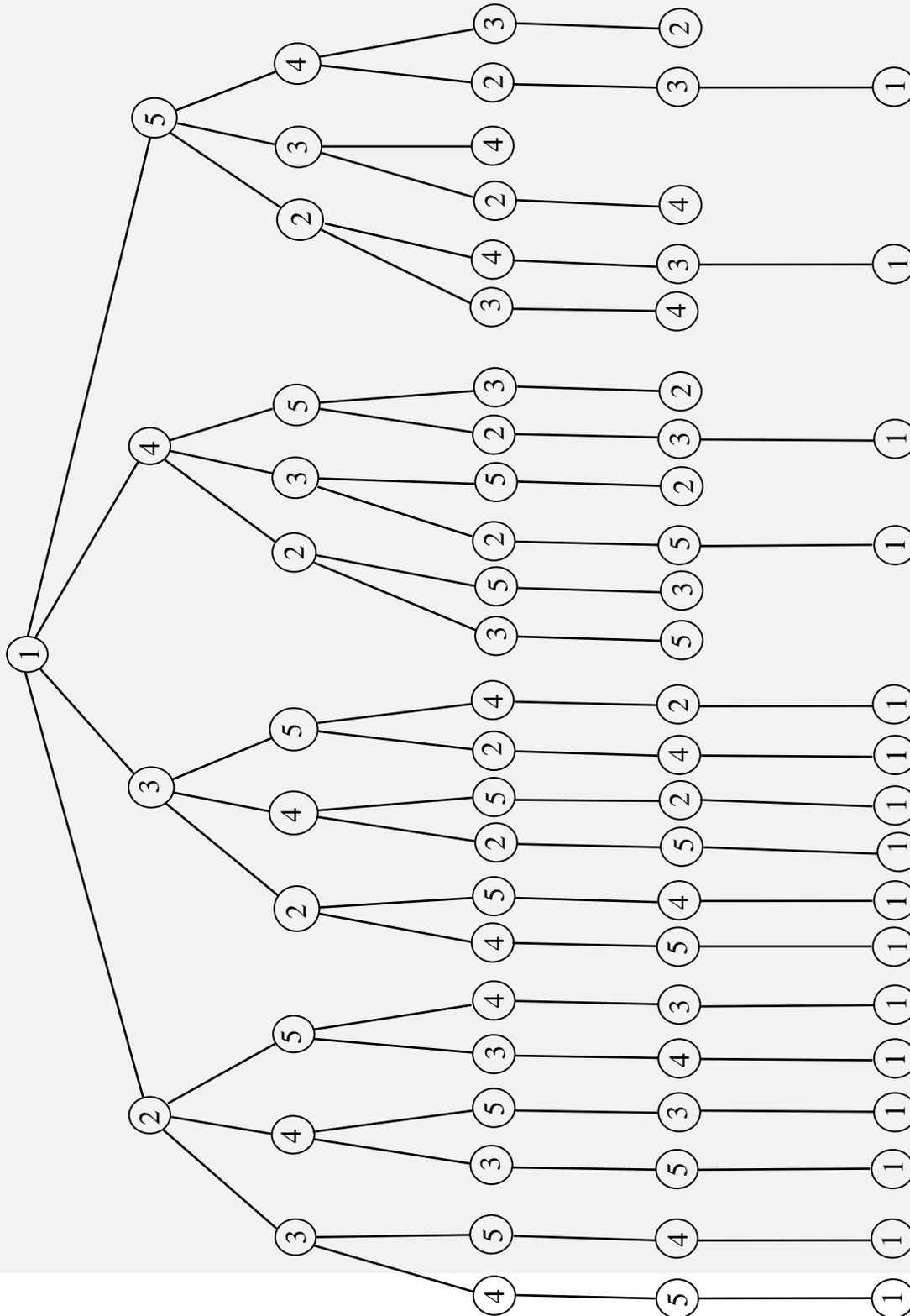


Рис. 3

Естественно, чем меньше начальное значение $T^{0т}$, тем меньше может оказаться $K_{yв}$. Поэтому на практике его обычно выбирают как меньшее из следующих маршрутов: полученного на основе последовательного обхода вершин и маршрутов на основе ближайшей вершины, рассчитанных на основе последовательного перебора начальной вершины. Но в данном случае $T^{0т} = Z\langle 132451 \rangle = 299.45$ не изменило бы конечного результата.

Известен также геометрический метод решения данной оптимизационной задачи, суть которого заключается в следующем. Все узлы отмечаются на плоскости точками в соответствии с их координатами. При этом выделяются граничные узлы, для которых существуют прямые, разделяющие плоскость на полуплоскости, одна из которых не содержит точек данного множества.

На первом этапе процедуры строится многоугольник, вершинами которого являются граничные узлы. На втором этапе производится последовательное включение в контур многоугольника точек, дающих минимальное увеличение длины этого контура. При наличии нескольких точек, дающих одинаковое увеличение периметра многоугольника, они включаются в контур одновременно. Если для некоторой точки имеется несколько претендующих на замену сторон многоугольника, то предпочтение отдается той из них, замена которой на ломаную, проходящую через центр тяжести системы внутренних точек, дает минимальное увеличение длины контура.

Ниже рассмотрен пример, иллюстрирующий данный метод. На рис. 4 показано расположение узлов некоторой ЛВС, а в табл. 6 приведены расстояния между узлами.

Для начала найдем длину одного из маршрутов по методу «ближайшей вершины»:

$$Z = \langle 1-6-7-11-8-10-9-5-3-2-4-1 \rangle = 13+10+10+10+13+8+30+55+40+90+80 = 359.$$

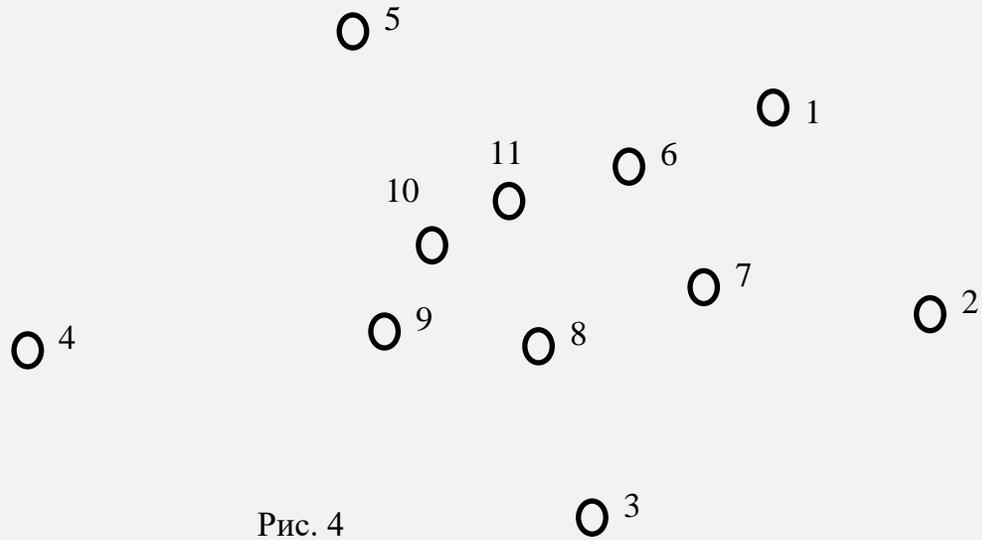


Рис. 4

Таблица 6

	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}
q_1	30	50	80	30	13	22	32	43	35	22
q_2		40	90	56	28	22	33	52	45	53
q_3			60	55	40	30	22	30	30	33
q_4				60	70	70	60	40	48	62
q_5					28	35	35	30	25	25
q_6						10	18	32	23	10
q_7							13	30	23	10
q_8								20	13	10
q_9									8	23
q_{10}										13

Теперь перейдем к геометрическому методу. Очевидно, что граничными точками являются узлы 1, 2, 3, 4, 5. Через них строится граничный многоугольник (см. рис. 5).

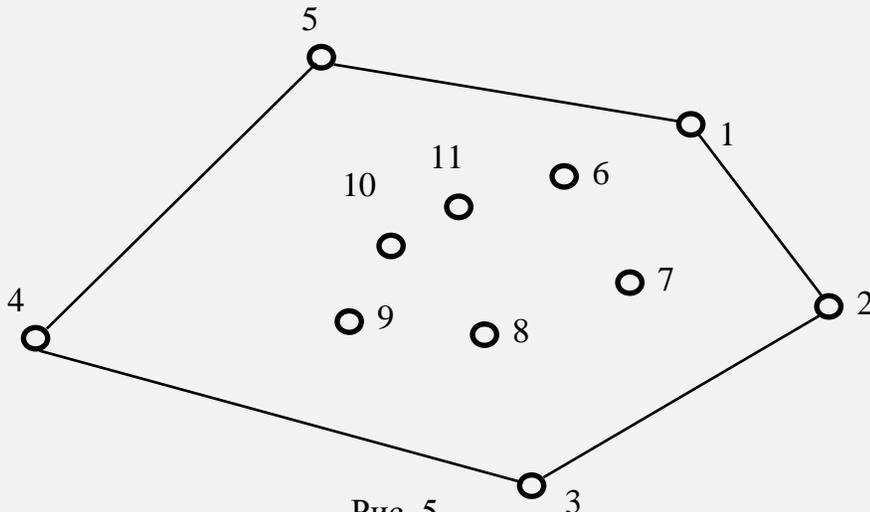


Рис. 5

Каждая из внутренних точек i ($i=6, 7, 8, 9, 10, 11$) образует со стороной многоугольника (j, k) треугольник ($j=k; k=1, 2, 3, 4, 5$). Используя табл. 6, определяется для каждой внутренней точки и каждой стороны многоугольника

$$\Delta_{j,k}^i = d_{i,k} + d_{i,j} - d_{j,k},$$

т.е. удлинение, вносимое точкой при включении её в контур.

$$\Delta_{1,2}^6 = d_{6,1} + d_{6,2} - d_{1,2} = 13+28-30 = 11;$$

$$\Delta_{2,3}^6 = d_{6,2} + d_{6,3} - d_{2,3} = 28+40-40 = 28;$$

$$\Delta_{3,4}^6 = d_{6,3} + d_{6,4} - d_{3,4} = 40+70-60 = 50;$$

$$\Delta_{4,5}^6 = d_{6,4} + d_{6,5} - d_{4,5} = 70+28-60 = 38;$$

$$\Delta_{5,1}^6 = d_{6,5} + d_{6,1} - d_{5,1} = 28+13-30 = 11;$$

$$\Delta_{1,2}^7 = 22+22-30 = 14;$$

$$\Delta_{2,3}^7 = 22+30-40 = 12;$$

$$\Delta_{3,4}^7 = 30+70-60 = 40;$$

$$\Delta_{4,5}^7 = 70+35-60 = 45;$$

$$\Delta_{5,1}^7 = 35+22-30 = 27;$$

$$\Delta_{1,2}^9 = 43+52-30 = 65;$$

$$\Delta_{2,3}^9 = 52+30-40 = 42;$$

$$\Delta_{3,4}^9 = 30+40-60 = 10;$$

$$\Delta_{4,5}^9 = 40+30-60 = 10;$$

$$\Delta_{5,1}^9 = 30+43-30 = 43;$$

$$\Delta_{1,2}^8 = 32+33-30 = 35;$$

$$\Delta_{2,3}^8 = 33+22-40 = 15;$$

$$\Delta_{3,4}^8 = 22+60-60 = 22;$$

$$\Delta_{4,5}^8 = 60+35-60 = 35;$$

$$\Delta_{5,1}^8 = 35+32-30 = 37;$$

$$\Delta_{1,2}^{10} = 35+45-30 = 50;$$

$$\Delta_{2,3}^{10} = 45+30-40 = 35;$$

$$\Delta_{3,4}^{10} = 30+48-60 = 18;$$

$$\Delta_{4,5}^{10} = 48+25-60 = 13;$$

$$\Delta_{5,1}^{10} = 25+35-30 = 30;$$

$$\Delta_{1,2}^{11} = 22+53-30 = 45;$$

$$\Delta_{2,3}^{11} = 53+33-40 = 46;$$

$$\Delta_{3,4}^{11} = 33+62-60 = 35;$$

$$\Delta_{4,5}^{11} = 62+25-60 = 27;$$

$$\Delta_{5,1}^{11} = 25+22-30 = 17.$$

Из результатов видно, что $\min\{\Delta_{j,k}^i\} = \Delta_{3,4}^9 = \Delta_{4,5}^9 = 10$, т.е. на замену претендуют стороны (3, 4) и (4, 5). Для корректного выбора определяется центр тяжести системы внутренних точек (табл. 7).

Таблица 7

	q ₆	q ₇	q ₈	q ₉	q ₁₀	q ₁₁
q ₆	103	10	18	32	23	10
q ₇		86	13	30	23	10
q ₈			74	20	13	10
q ₉				113	8	23
q ₁₀					80	13
q ₁₁						66

Табл. 7 формируется на основе табл.

6 путем вычеркивания строк и столбцов уже соединенных узлов. На главной диагонали указываются значения центров тяжести как суммы расстояний от данной точки до всех остальных.

Из анализа табл. 7 видно, что центр тяжести приходится на точку 11. Ввиду того, что $\Delta_{4,5}^{11} < \Delta_{3,4}^{11}$ ($62+25-60=27 < 33+62-60=35$), сторона (4, 5) заменяется на ломаную $\langle(4, 9), (9, 5)\rangle$ (рис. 6).

Повторяем предыдущую операцию для оставшихся узлов с учетом новых рёбер.

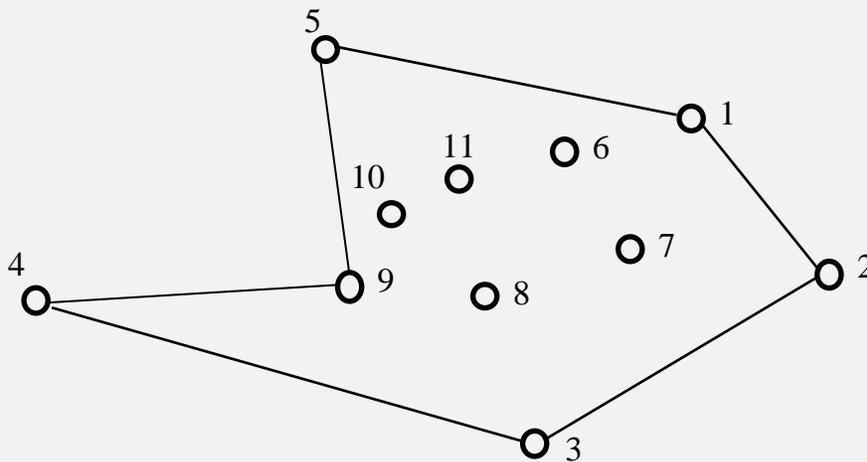


Рис. 6

$$\Delta_{1,2}^7 = 22+22-30 = 14;$$

$$\Delta_{2,3}^7 = 22+30-40 = 12;$$

$$\Delta_{3,4}^7 = 30+70-60 = 40;$$

$$\Delta_{4,9}^7 = 70+30-40 = 60;$$

$$\Delta_{1,2}^8 = 32+33-30 = 35;$$

$$\Delta_{2,3}^8 = 33+22-40 = 15;$$

$$\Delta_{3,4}^8 = 22+60-60 = 22;$$

$$\Delta_{4,9}^8 = 60+20-40 = 40;$$

$$\Delta_{9,5}^7 = 30+35-30 = 35;$$

$$\Delta_{5,1}^7 = 35+22-30 = 27;$$

$$\Delta_{1,2}^{10} = 35+45-30 = 50;$$

$$\Delta_{2,3}^{10} = 45+30-40 = 35;$$

$$\Delta_{3,4}^{10} = 30+48-60 = 18;$$

$$\Delta_{4,9}^{10} = 48+8-40 = 16;$$

$$\Delta_{9,5}^{10} = 8+25-30 = 3;$$

$$\Delta_{5,1}^{10} = 25+35-30 = 30;$$

$$\Delta_{9,5}^8 = 20+35-30 = 25;$$

$$\Delta_{5,1}^8 = 35+32-30 = 37;$$

$$\Delta_{1,2}^{11} = 22+53-30 = 45;$$

$$\Delta_{2,3}^{11} = 53+33-40 = 46;$$

$$\Delta_{3,4}^{11} = 33+62-60 = 35;$$

$$\Delta_{4,9}^{11} = 62+23-40 = 45;$$

$$\Delta_{9,5}^{11} = 23+25-30 = 18.$$

$$\Delta_{5,1}^{11} = 25+22-30 = 17.$$

Проведенный расчет показал, что на включение претендует только узел 10 путем замены ребра (9, 5) на ломаную $\langle(9, 10), (10, 5)\rangle$ (рис. 7).

Аналогичным образом последовательно включаются в контур узлы 6, 11, 7, 8. Окончательный вариант топологии для данного примера приведен на рис. 8.

Теперь осталось просчитать длину полученного маршрута:

$$Z = \langle 1-2-3-4-9-10-5-11-8-7-6-1 \rangle = 30+40+60+40+8+25+25+10+13+10+13 = 274.$$

Это меньше, чем длина маршрута по «ближайшей вершине» $Z = 359$.

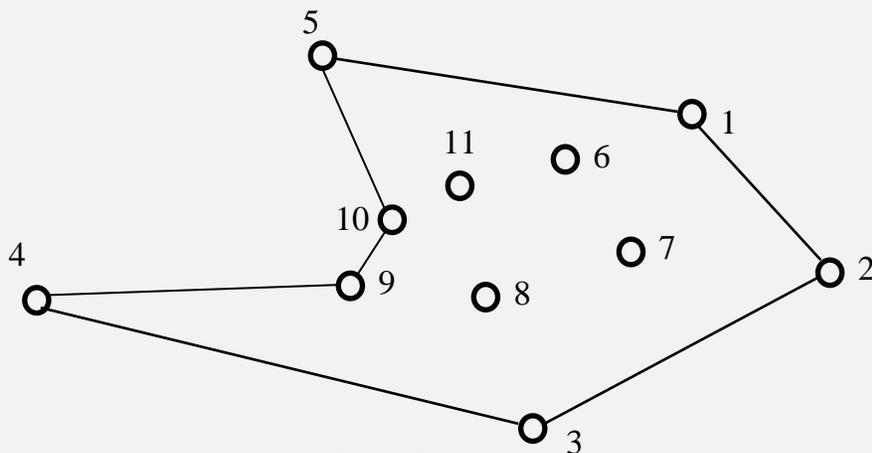


Рис. 7

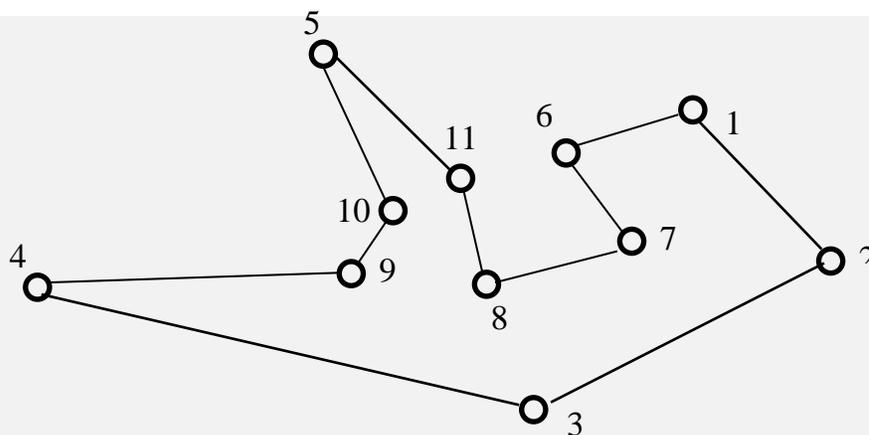


Рис. 8

- В соответствии с вариантом выбрать из табл. 8 координаты узлов анализируемой сети на двумерной плоскости.

Таблица 8

№ п/п	Координаты узлов									
	X ₁ ,Y ₁	X ₂ ,Y ₂	X ₃ ,Y ₃	X ₄ ,Y ₄	X ₅ ,Y ₅	X ₆ ,Y ₆	X ₇ ,Y ₇	X ₈ ,Y ₈	X ₉ ,Y ₉	X ₁₀ ,Y ₁₀
1.1	56,34	9,60	60,89	13,77	52,35	55,70	39,88	90,54	57,21	11,95
1.2	50,3	79,37	10,23	87,23	16,13	90,93	99,94	87,71	23,26	32,94
1.3	90,50	19,53	57,75	71,18	49,15	73,9	53,42	20,72	78,44	78,96
1.4	73,41	85,87	96,61	92,47	40,50	42,94	70,94	48,51	15,14	53,91
1.5	81,48	11,96	3,96	60,63	92,37	78,16	91,37	27,6	21,67	97,68
1.6	23,93	5,3	19,72	54,63	95,61	7,47	97,16	44,27	15,3	38,21
1.7	43,56	0,98	54,85	31,39	34,8	81,76	43,39	83,73	71,22	17,73
1.8	87,60	89,11	3,2	12,37	83,50	69,10	75,93	96,59	27,29	57,62
1.9	2,50	84,88	65,17	45,32	55,72	94,94	50,68	43,84	86,24	74,53
1.10	32,76	25,56	1,60	61,12	50,26	13,46	2,87	35,70	83,13	17,10
1.11	2,11	42,0	38,54	66,15	34,63	84,28	21,41	51,45	26,89	91,45
1.12	84,35	73,54	28,15	13,44	73,72	51,7	15,0	32,63	36,37	81,70
1.13	83,6	24,98	79,9	27,24	90,91	6,16	73,89	17,14	51,7	38,69
1.14	52,70	62,6	25,38	98,8	57,56	92,21	98,1	56,72	21,79	45,65
1.15	86,68	63,93	0,68	77,39	88,90	6,69	60,43	68,72	98,67	22,70
1.16	56,34	9,60	60,89	13,77	52,35	69,93	34,90	71,80	96,32	58,3
1.17	50,3	79,37	10,23	87,23	16,13	19,45	34,73	39,79	14,99	11,95
1.18	90,50	19,53	57,75	71,18	49,15	38,45	71,84	5,28	57,21	32,94
1.19	73,41	85,87	96,61	92,47	40,50	41,41	70,63	90,54	23,26	78,96
1.20	81,48	11,96	3,96	60,63	92,37	95,61	31,39	3,2	84,88	32,76
1.21	23,93	5,3	19,72	54,63	95,61	34,8	12,37	65,17	25,56	2,1

Таблица 6. Продолжение

1.22	43,56	0,98	54,85	31,39	34,8	83,5	45,32	1,6	42,7	84,35
1.23	87,60	89,11	3,2	12,37	83,50	6,16	98,1	68,72	70,28	4,77
1.24	2,50	84,88	65,17	45,32	55,72	92,21	60,43	24,38	65,30	91,82
1.25	32,76	25,56	1,60	61,12	50,26	6,69	18,50	83,85	56,29	57,41
1.26	2,11	42,0	38,54	66,15	34,63	69,93	34,90	71,80	96,32	58,3
1.27	84,35	73,54	28,15	13,44	73,72	19,45	34,73	39,79	14,99	11,95
1.28	83,6	24,98	79,9	27,24	90,91	38,45	71,84	5,28	57,21	32,94
1.29	52,70	62,6	25,38	98,8	57,56	41,41	70,63	90,54	23,26	78,96
1.30	86,68	63,93	0,68	77,39	88,90	95,61	31,39	3,2	84,88	32,76
1.31	32,65	64,89	7,95	79,29	7,41	34,8	12,37	65,17	25,56	2,1
1.32	10,30	25,41	52,55	69,84	34,41	83,5	45,32	1,6	42,12	84,35
1.33	15,17	75,55	78,30	62,85	70,37	69,93	34,90	71,80	96,32	58,3
1.34	59,57	85,6	91,72	90,6	65,36	19,45	34,73	39,79	14,99	11,95
1.35	29,74	95,87	60,41	33,91	97,55	38,45	71,84	5,28	57,21	32,94
2.1	32,65	64,89	7,95	79,29	7,41	69,93	18,50	24,38	70,28	46,32
2.2	10,30	25,41	52,55	69,84	34,41	19,45	34,90	83,85	65,30	4,77
2.3	15,17	75,55	78,30	62,85	70,37	38,45	34,73	71,80	56,29	91,82
2.4	59,57	85,6	91,72	90,6	65,36	41,41	71,84	39,79	96,32	57,41
2.5	29,74	95,87	60,41	33,91	97,55	47,58	70,63	5,28	14,99	58,3
2.6	56,34	79,37	57,75	92,47	92,37	7,47	43,39	96,59	86,24	17,10
2.7	50,3	19,53	96,61	60,63	95,61	81,76	75,93	43,84	83,13	91,45
2.8	90,50	85,87	3,96	54,63	34,8	69,10	50,68	35,70	26,89	81,70
2.9	73,41	11,96	19,72	31,39	83,50	94,94	2,87	51,45	36,37	38,69
2.10	81,48	5,3	54,85	12,37	55,72	13,46	21,41	32,63	51,7	45,65
2.11	23,93	0,98	3,2	45,32	50,26	84,28	15,0	17,14	21,79	22,70
2.12	43,56	89,11	65,17	61,12	34,63	51,7	73,89	56,72	98,67	46,32
2.13	87,60	84,88	1,60	66,15	73,72	6,16	98,1	68,72	70,28	4,77
2.14	2,50	25,56	38,54	13,44	90,91	92,21	60,43	24,38	65,30	91,82
2.15	32,76	42,0	28,15	27,24	57,56	6,69	18,50	83,85	56,29	57,41
2.16	2,11	73,54	79,9	98,8	88,90	69,93	34,90	71,80	96,32	58,3
2.17	84,35	24,98	25,38	77,39	7,41	19,45	34,73	39,79	14,99	11,95
2.18	83,6	62,6	0,68	79,29	34,41	38,45	71,84	5,28	57,21	32,94
2.19	52,70	63,93	7,95	69,84	70,37	41,41	70,63	90,54	23,26	78,96
2.20	11,95	23,26	20,72	70,94	78,16	95,61	31,39	3,2	84,88	32,76
2.21	32,94	78,44	48,51	91,37	7,47	34,8	12,37	65,17	25,56	2,1
2.22	78,96	15,14	27,6	97,16	81,76	83,5	45,32	1,6	42,8	84,35
2.23	56,34	79,37	57,75	92,47	92,37	41,41	70,63	90,54	23,26	78,96
2.24	50,3	19,53	96,6	60,63	95,61	9,61	31,39	3,2	84,88	32,76

Таблица 6. Окончание

2.25	90,50	85,87	3,96	54,63	34,8	34,8	12,37	65,17	25,56	2,1
2.26	73,41	11,96	19,72	31,39	83,50	83,5	45,32	1,6	42,34	84,35
2.27	90,50	85,87	3,96	54,63	34,8	34,8	12,37	65,17	45,56	2,1
2.28	73,41	31,96	19,72	31,39	83,50	83,5	45,32	1,6	42,34	84,35
2.29	2,11	73,54	79,9	98,8	88,90	99,93	34,90	71,80	96,32	58,3
2.30	50,3	79,37	10,23	87,23	26,13	39,45	54,73	39,79	14,99	11,95
2.31	73,41	85,87	96,61	92,47	40,50	42,94	70,94	48,51	45,14	53,91
2.32	78,96	55,14	27,6	97,16	11,76	83,5	45,32	91,6	42,8	84,35
2.33	87,60	89,81	33,2	12,37	83,50	69,10	75,93	96,59	27,29	57, 2
2.34	50,39	79,37	50,23	87,23	16,83	19,45	34,73	39,79	14,99	11,95

- Рассчитать расстояния между узлами и занести их в таблицу, аналогичную табл. 6.
- Рассчитать длину маршрута на основе последовательного обхода вершин.
- Рассчитать длины маршрутов на основе метода ближайшей вершины с перебором всех вершин в качестве начальной в маршруте.
- Из одиннадцати рассчитанных маршрутов выбрать минимальный в качестве текущего оптимального маршрута T^{0T} .
- Методом ветвей и границ проанализировать первых 20 уникальных маршрутов (аналог табл. 4).
- Рассчитать для них по выражению (1) КУВ и построить на основе таблицы 4 итоговое дерево анализа (аналог рис. 3).
- Найти оптимальный вариант топологии геометрическим методом и рассчитать его длину.

ПРАВИЛА ОФОРМЛЕНИЯ ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

Текстовые документы оформляются на листах формата А4, графический материал допускается представлять на листах формата А3.

Поля на листе: левое – не менее 30, правое – не менее 10, верхнее – не менее 15, нижнее – не менее 20 мм.

Нумерация страниц сквозная. Номера проставляются сверху справа арабской цифрой. Первой страницей считается титульный лист, на котором номер страницы не проставляется. Страницами считаются и листы с рисунками и приложениями.

Номера разделов пишутся прописными буквами в середине строки. Расстояние между заголовком и текстом – два интервала.

Номера подразделов и пунктов начинаются с абзацного отступа, вразрядку, с прописной буквы и без точки в конце.

Расстояние между последней строкой текста предыдущего раздела и последующим заголовком при расположении их на одной странице – три интервала.

Разделы и подразделы нумеруются арабскими цифрами с точкой. Ссылки на пункты, разделы и подразделы указываются в виде «в разд. 4», «в п. 3.3.4».

Текст разделов печатается через 1,5 интервала. Высота шрифта не менее 12 пунктов.

Перечисления отображаются арабскими цифрами со скобкой, например: 2), 3) и т.д. Допускается выделять перечисления дефисом перед пунктом текста или символом, его заменяющим.

Все иллюстрации именуется рисунками. Все рисунки, таблицы и формулы нумеруются арабскими цифрами последовательно или в пределах раздела. В приложениях – в пределах приложения. Каждый рисунок имеет подпись – название снизу рисунка, например:

Рис. 23. Главное окно программного комплекса.

На все рисунки, таблицы и формулы в документе должны быть ссылки в виде: «рис. 23» или «блок-схема данного модуля приведена на рис. 31». Если позволяет место, рисунки и таблицы должны размещаться сразу после абзаца, в котором они упоминаются в первый раз, или как можно ближе к этому абзацу на следующей странице. Если рисунок занимает более одной страницы, то на всех страницах, кроме первой, проставляется номер рисунка и слово «Продолжение». Ри-

сунки следует размещать так, чтобы их можно было рассматривать без поворота страницы. В противном случае рисунки надо размещать так, чтобы для просмотра страницу надо было повернуть по часовой стрелке.

Схемы алгоритмов выполняются в соответствии с ЕСПД (Приложение 5).

Номер таблицы размещается в правом верхнем углу или перед заголовком таблицы, если он есть. Ссылки на таблицы в тексте документа указываются в виде слов «табл.» и номера таблицы.

Номера формул ставятся с правой стороны в круглых скобках. Ссылка на номер формулы дается в скобках, например: «радиус окружности определяется по формуле (7)».

Каждое приложение должно начинаться с новой страницы с указанием в правом углу слова «ПРИЛОЖЕНИЕ» и иметь тематический заголовок. При наличии более одного приложения все они нумеруются арабскими цифрами. Все рисунки и таблицы в приложении нумеруются арабскими цифрами в пределах каждого приложения с добавлением буквы «П», например:

Рис. П. 9

Дополнительные требования к оформлению:

- использовать автоматическую нумерацию страниц;
- содержание должно быть вставлено автоматически;
- желательно использовать колонтитулы;
- сокращения должны быть расшифрованы.

ОРГАНИЗАЦИЯ ЗАЩИТЫ КУРСОВОЙ РАБОТЫ

После создания программного приложения, реализующего задачу курсовой работы, результат демонстрируется студентом преподавателю, руководящему данным процессом, для чего заранее определяется дата и время встречи или это происходит в заранее установленные часы консультации по курсовой работе. Если преподаватель выражает согласие с представленными результатами, то студент

переходит к написанию пояснительной записке с указанным выше содержанием. В противном случае студент фиксирует все сделанные преподавателем замечания и согласовывается дата и время следующей встречи.

После написания студентом пояснительной записки она, с подписью студента и текущей датой, передается руководителю работы для просмотра, в ходе которого проверяется как собственно содержимое, общая структура которого представлена ранее в данных методических указаниях, так и правила оформления технической документации. При выявлении недочетов, влияющих на качество курсовой работы, записка возвращается студенту с подробным объяснением сути претензий и назначением срока для их устранения.

Пояснительная записка без замечаний подписывается руководителем курсовой работы с визой «Допущена к защите». Все подписи сопровождаются датами. После этого устанавливается преподавателем единолично или по согласованию со студентом дата защиты курсовой работы.

Защита принимается комиссией из двух-трех человек, назначенной заведующим кафедрой. В неё обязательно входит руководитель разработки, ведущий лектор (если он не является руководителем работы) или специалист по данной проблематике, который может быть как работником кафедры, так и представителем сторонней организации, являющийся экспертом по данной проблематике.

В ходе защиты студент демонстрирует работу программного приложения и отвечает на вопросы как по листингу, так и по другим разделам пояснительной записки. На основании защиты определяется окончательная оценка курсовой работы, которая выставляется в ведомость и зачетную книжку, а также проставляется на титульном листе пояснительной записки с датой и подписью всеми членами комиссии.

Курсовая работа оценивается по следующим параметрам:

1. соответствие содержания работы ее теме;

2. тщательность проработки материала (постановки задачи, обосновании модульной структуры программы, выбора алгоритмов);
3. самостоятельность при выполнении работы;
4. стиль изложения работы, ее логика и качество формулировок в ее выводах;
5. соответствие оформления курсовой работы требованиям;
6. соблюдение правил орфографии и пунктуации;
7. качество оформления графического материала;
8. ответы на вопросы в процессе защиты.

Нормы оценки курсовой работы:

1. Оценку «отлично» получают работы, в которых:

- a. используется материал, освоенный студентом дополнительно в русле самостоятельного расширения знаний по изучаемой дисциплине, т.е. очевидна самостоятельность студента в выполнении работы, предлагаются обоснованные методы решения поставленной задачи, дается аргументированный критический анализ используемого теоретического и практического материала на базе глубоких знаний технической информации по данной теме.
- b. материал пояснительной записки излагается грамотно, логично, последовательно.
- c. оформление отвечает требованиям написания курсовой работы.
- d. во время защиты студент показал умение кратко, доступно (ясно) представить результаты работы, адекватно ответить на поставленные вопросы, обосновать (защитить) выбранные методы решения проблемы.

2. Оценка «хорошо» ставится:

- a. в работе, выполненной на достаточном теоретическом и практическом уровне, полно и всесторонне решаются вопросы поставленной проблемы, но нет должной степени творчества.

- b. студент показал знание теоретического материала по рассматриваемой проблеме, однако умение анализировать, аргументировать свою точку зрения, делать обобщения и выводы вызывают у него затруднения.
- c. материал пояснительной записки не всегда излагается логично, последовательно. Имеются недочеты в оформлении курсовой работы.
- d. во время защиты студент показал умение кратко, доступно (ясно) представить результаты исследования, однако затруднялся отвечать на поставленные вопросы.

3. Оценку «удовлетворительно» имеют работы:

- a. правильно, но не в полной мере, используются теоретические и практические знания по изучаемой дисциплине, не проявилось умение логически стройного их использования, самостоятельного анализа источников, предлагаемые решения не содержат новизны, встречаются отдельные ошибки в решении поставленной в ходе проектирования задачи.
- b. студент испытывает затруднения при необходимости анализировать, аргументировать свою точку зрения, делать обобщение и выводы.
- c. материал не всегда излагается логично, последовательно.
- d. имеются недочеты в оформлении курсовой работы.
- e. во время защиты студент затрудняется в представлении результатов работы и ответах на поставленные вопросы.

4. *Оценку «неудовлетворительно» студенты получают в случае, когда не могут ответить на вопросы преподавателей во время защиты, не владеют материалом работы, не в состоянии дать объяснения выводам и практическим результатам по решаемой проблеме.*

Список рекомендуемой литературы

Основная литература:

1. Романов, Е. Л. Программная инженерия: учебное пособие: [16+] / Е. Л. Романов; Новосибирский государственный технический университет. – Новосибирск: Новосибирский государственный технический университет, 2017. – 395 с. : табл., схем., ил. – (Учебники НГТУ). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=573945> (дата обращения: 16.05.2022). – Библиогр. в кн. – ISBN 978-5-7782-3455-0. – Текст: электронный. [74](#)
2. Кайгородцев, Г. И. Введение в курс метрической теории и метрологии программ: учебник/ Г. И. Кайгородцев. – Новосибирск: Новосибирский государственный технический университет, 2011. – 190 с. : табл., схем., ил. – (Учебники НГТУ). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=435984> (дата обращения: 16.05.2022). – Библиогр. в кн. – ISBN 978-5-7782-1648-8. – Текст: электронный.
3. Программирование и основы алгоритмизации: учебное пособие / В. К. Зольников, П. Р. Машевич, В. И. Анциферова, Н. Н. Литвинов; Федеральное агентство по образованию, Воронежская государственная лесотехническая академия. – Воронеж: Воронежская государственная лесотехническая академия, 2011. – 341 с.: ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=142309> (дата обращения: 29.05.2022). – Текст: электронный.

Дополнительная литература:

4. Филлипс Д., Гарсиа-Диас А. Методы анализа сетей / Д. Филлипс - М.: Мир, 1984. - 496 с.
4. Пономарев В.Ф. Основы дискретной математики: Учеб. пособие / В.Ф. Пономарев. – Калининград: КГТУ, 1997. - 165 с.

6. Кириков И.А., Пономарев В.Ф. Основы построения автоматизированных систем управления: Методические указания по выполнению лабораторных работ для студентов высших учебных заведений по специальности «Автоматизированные системы управления» / И.А. Кириков - Калининград: КТИРПиХ, 1986. - 129 с.
7. Локальные вычислительные сети: Справочник: В 3-х кн. – Кн. 3. Организация функционирования, эффективность, оптимизация / С.В. Назаров, Н.В. Ашихмин, А.В. Луговец и др.; под ред. С.В. Назарова. – М.: Финансы и статистика, 1995 - 248 с.
8. Курейчик В.М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР / В.М. Курейчик. - М.: Радио и связь, 1987. - 352 с.

Образец заполнения титульного листа

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО РЫБОЛОВСТВУ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Калининградский государственный технический университет»

Институт цифровых технологий

Кафедра _____
наименование кафедры

Курсовая работа
допущена к защите
Руководитель: _____
(уч. степень, звание, должность*)
_____ И.О. Фамилия
« ___ » _____ 202__ г.

Курсовая работа защищена
с оценкой _____
Руководитель: _____
(уч. степень, звание, должность)
_____ И.О. Фамилия
« ___ » _____ 202__ г.

ТЕМА КУРСОВОЙ РАБОТЫ

Курсовая работа по дисциплине
«Наименование дисциплины»
КР.ХХ¹.ХХ.ХХ.ХХ².Х³.Х⁴

Работу выполнил:
студент гр.____
_____ И.О. Фамилия
« ___ » _____ 20__ г.

Калининград
202__

ПОЯСНЕНИЯ

Обозначения в шифре

КП.ХХ¹.ХХ.ХХ.ХХ².Х³.ХХ⁴.ПЗ

КР – курсовая работа.

КП – курсовой проект.

ХХ¹ – номер кафедры.

ХХ.ХХ.ХХ² – шифр направления подготовки

Х³ – последняя цифра года, когда выполнена работа (например, 2022 год, будет цифра 2).

ХХ⁴ – номер варианта курсовой работы(проекта).

ПЗ – пояснительная записка

*Ученую степень и звание следует сокращать в соответствии с рекомендациями Министерства науки РФ, например:

Сокращение	Полное написание
Учёные степени	
д-р биол. наук	доктор биологических наук
д-р с.-х. наук	доктор сельскохозяйственных наук
д-р техн. наук	доктор технических наук
канд. с.-х. наук	кандидат сельскохозяйственных наук
канд. техн. наук	кандидат технических наук
канд. хим. наук	кандидат химических наук
Учёные звания	
доц.	доцент
проф.	профессор

Пример рубрикации

Содержание

Содержание

1. Цели и задачи курсовой работы	2
2. Обоснование выбора языка Python для написания курсовой работы	3
3. Техническое задание	4
3.1 Обоснование для разработки	6
3.2 Описание задания	9
.....	
13. Руководство системного программиста	34
13.1 Требования к информационной и программной совместимости	37
13.2 Требования к составу и параметрам технических средств	39
14. Руководство пользователя	43
.....	
Список использованной литературы	47

Пример аннотации

Аннотация

Представленная курсовая работа состоит из введения, трех глав, заключения, списка литературных источников и приложений.

В первой части работы рассматривается подробная характеристика главных источников права, а также, источников права, используемых в различных правовых семьях. Дана сводная классификация. В следующей главе детально анализируется каждый из видов правовых источников и делается вывод на основании изложенного материала.

Ключевые слова: источники права, правовые семьи.

Работа состоит из 31 страницы, содержит 43 литературных источника, 3 таблицы и 8 приложений.

ПРИЛОЖЕНИЕ 4

Требования к оформлению блок-схем

Схема - это абстракция какого-либо процесса или системы, наглядно отображающая наиболее значимые части. Соглашения выработаны для изображения схем-алгоритмов и закреплены ГОСТами и международными стандартами.

На территории Российской Федерации действует единая система программной документации (ЕСПД), частью которой является Государственный стандарт — ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем» [1].

В приложении приведена только часть по разработки схем алгоритмов программ.

Рассматриваемый ГОСТ практически полностью соответствует международному стандарту **ISO 5807:1985**.

Элементы блок-схем алгоритмов

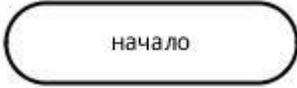
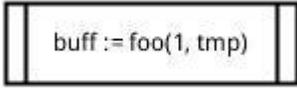
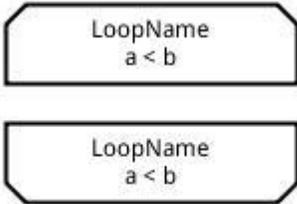
Блок-схема представляет собой совокупность символов, соответствующих этапам работы алгоритма и соединяющих их линий.

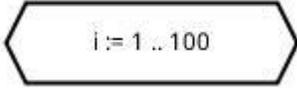
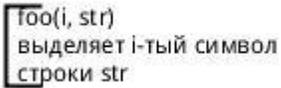
Пунктирная линия используется для соединения символа с комментарием. *Сплошная линия* отражает зависимости по управлению между символами и может снабжаться стрелкой. Стрелку можно не указывать при направлении дуги слева направо и сверху вниз.

Линии должны подходить к символу слева, либо сверху, а исходить снизу, либо справа.

Есть и другие типы линий, используемые, например, для изображения блок-схем параллельных алгоритмов

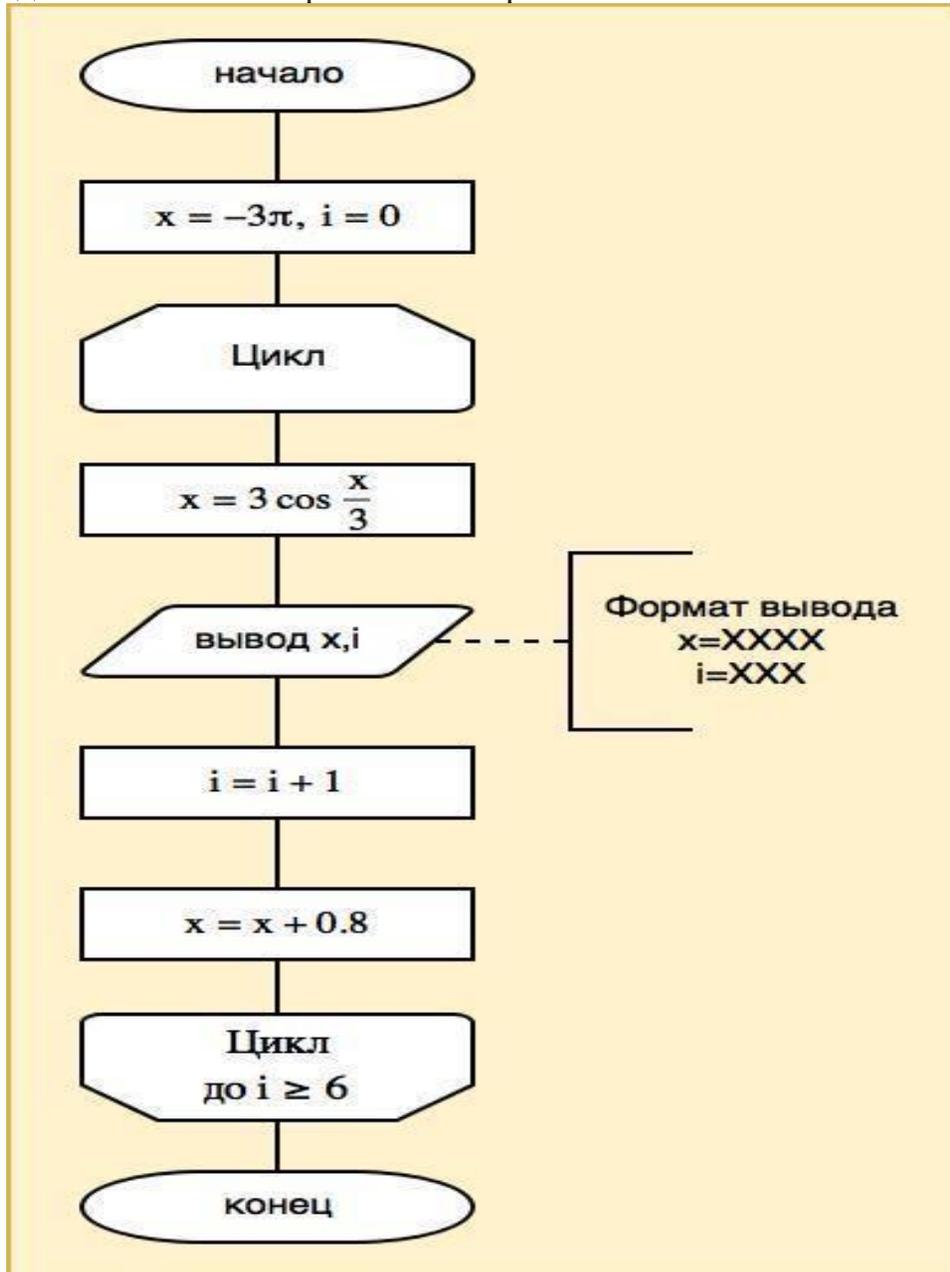
Далее рассмотрены лишь основные символы, которых вполне достаточно для оформления рассматриваемой пояснительной записки.

 <p>Терминатор начала и конца работы функции</p>	<p>Терминатором начинается и заканчивается любая функция. Тип возвращаемого значения и аргументов функции обычно указывается в комментариях к блоку терминатора.</p>
 <p>Операции ввода и вывода данных</p>	<p>В ГОСТ определено множество символов ввода/вывода, например вывод на магнитные ленты, дисплеи и т.п. Если источник данных не принципиален, обычно используется символ параллелограмма. Подробности ввода/вывода могут быть указаны в комментариях.</p>
 <p>Выполнение операций над данными</p>	<p>В блоке операций обычно размещают одно или несколько (ГОСТ не запрещает) операций присваивания, не требующих вызова внешних функций.</p>
 <p>Блок, иллюстрирующий ветвление алгоритма</p>	<p>Блок в виде ромба имеет один вход и несколько подписанных выходов. В случае, если блок имеет 2 выхода (соответствует оператору ветвления), на них подписывается результат сравнения — «да/нет». Если из блока выходит большее число линий (оператор выбора), внутри него записывается имя переменной, а на выходящих дугах — значения этой переменной.</p>
 <p>Вызов внешней процедуры</p>	<p>Вызов внешних процедур и функций помещается в прямоугольник с дополнительными вертикальными линиями.</p>
 <p>Начало и конец цикла</p>	<p>Символы начала и конца цикла содержат имя и условие. Условие может отсутствовать в одном из символов пары. Расположение условия, определяет тип оператора, соответствующего символам на языке высокого уровня — оператор с предусловием (while) или постусловием (do ... while).</p>

 <p>Подготовка данных</p>	<p>Символ «подготовка данных» в произвольной форме (в ГОСТ нет ни пояснений, ни примеров), задает входные значения. Используется обычно для задания циклов со счетчиком.</p>
 <p>Соединитель</p>	<p>В случае, если блок-схема не умещается на лист, используется символ соединителя, отражающий переход потока управления между листами. Символ может использоваться и на одном листе, если по каким-либо причинам тянуть линию не удобно.</p>
 <p>Комментарий</p>	<p>Комментарий может быть соединен как с одним блоком, так и группой. Группа блоков выделяется на схеме пунктирной линией.</p>

Примеры блок-схем

Ниже приведена блок-схема простого алгоритма



Часто для оформления блок-схем используются возможности *Microsoft Word*, но это мало удобно, например в *MS Word* нет стандартного блока для терминатора начала и конца алгоритма (прямоугольник со скругленными краями, а не овал). Более удобными являются утилиты *MS Visio* и *yEd* [5],

Локальный электронный методический материал

Леонид Григорьевич Высоцкий

ПРОГРАММНАЯ ИНЖЕНЕРИЯ (ПИ)

Редактор Г.А. Смирнова

Уч.-изд. л. 2,1. Печ. л. 2,6.

Издательство федерального государственного бюджетного образовательного
учреждения высшего образования
«Калининградский государственный технический университет».
236022, Калининград, Советский проспект, 1