

**ГОСУДАРСТВЕННОЕ ВЫСШЕЕ УЧЕБНОЕ ЗАВЕДЕНИЕ
ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
К ИНДИВИДУАЛЬНЫМ ЗАДАНИЯМ
ПО КУРСУ “КОМПЬЮТЕРНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА”**

**ГОСУДАРСТВЕННОЕ ВЫСШЕЕ УЧЕБНОЕ ЗАВЕДЕНИЕ
ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
К ИНДИВИДУАЛЬНЫМ ЗАДАНИЯМ
ПО КУРСУ “КОМПЬЮТЕРНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА”**

**для студентов, обучающихся по направлению
“Программная инженерия”**

Рассмотрено на заседании кафедры
Программная инженерия
Протокол №1 от 30.08.2016 г.

Утверждено на заседании
учебно-издательского совета ДонНТУ
Протокол № от ____ . ____ . ____ г.

Донецк 2016

УДК 004.021

Методические указания к индивидуальным заданиям по курсу "Компьютерная дискретной математики" (для студентов, обучающихся по направлениям подготовки «Программная инженерия») / Сост.: Л.В. Незамова - Донецк: ДонНТУ, 2016. – 16с.

Методические указания к индивидуальным заданиям по курсу "Компьютерная дискретная математика" включает задания по самостоятельной работе студента по основным темам раздела "Теория графов".

Составители:

Незамова Л.В., асс.

Рецензент:

Скворцов А.Е., к.т.н., доцент

ВВЕДЕНИЕ

Среди дисциплин и методов дискретной математики теория графов и особенно алгоритмы на графах находят наиболее широкое применение в программировании.

Теория графов многократно переоткрывалась разными авторами при решении различных прикладных задач. Теория графов предоставляет очень удобный язык для описания программных моделей. Система специальных терминов и обозначений теории графов позволяют просто и доступно описывать сложные объекты.

Следует подчеркнуть, что конструирование структур данных для представления в программе объектов математической модели - это основа искусства практического программирования. Выбор наилучшего представления определяется требованиями конкретной задачи. Более того, при решении конкретных задач используются, как правило, некоторые комбинации или модификации указанных представлений, общее число которых очень велико.

В качестве моделей графы удобно использовать в тех случаях, когда рассматриваются системы каких-либо объектов, между которыми существуют определенные связи, а также в тех случаях, когда изучается структура системы, возможности ее функционирования.

1 ФОРМУЛИРОВКА СОДЕРЖАНИЯ ЗАДАНИЯ

В индивидуальном задании в соответствии с вариантом задания должен быть запрограммирован алгоритм решения одной из задач дискретной математике, по разделу “Теория графов”. Индивидуальное задание состоит в написании программы и отчёта.

1.1 Требования к программе

Программа может быть написана на одном из двух языков программирования – Си (допускается C++) или Object Pascale (для Delphi), в соответствии с вариантом задания. Использование других языков программирования (например, Java или C#) – не допускается. Реализация программы должна в точности соответствовать алгоритму, приведённому в книжке по варианту задания. Собственные изменения в алгоритме допускаются, но при этом они в обязательном порядке должны быть описаны в отчёте отдельным подразделом, вместе с обоснованием, почему эти изменения нужно было внести.

Обязательно наличие проблемно-ориентированного пользовательского интерфейса, соответствующего существу решаемой задачи, обеспечивающего однозначное и ясное понимание пользователем значения вводимых им данных и команд.

В программе должна быть возможность выбора между вводом исходных данных из файла или с клавиатуры. Во время работы программа должна записывать в файл все промежуточные переменные, информацию о том, какие из переменных поменяли своё состояние, информацию о том, какие проверки алгоритма были выполнены и какой шаг алгоритма выполняется. Результат работы программы должен выводиться на экран и дублироваться в файле листинга решения задачи.

Учебный характер индивидуальной работы и ограниченность времени самостоятельной работы, выделяемого на её выполнение, допускает условность

её результатов. По окончании работы должен быть предоставлен электронный носитель данных с программным продуктом, прилагаемым к выполненной работе отчет с результатами работы.

1.2 Варианты заданий

Задание студент выбирает в соответствии с номером своего варианта. Список книг с алгоритмами приводится в перечне списка литературы данного методического указания. Перечень вариантов заданий приведен в таблице 1.

Таблица 1. Варианты заданий

Вариант	Задание	Стр.	Источник
1	Поиск в глубину	142	Окулов
2	Поиск в ширину	143	Окулов
3	Порождение всех каркасов	145	Окулов
4	Каркас мин.веса (Краскал)	148	Окулов
5	Каркас мин. веса (Прим)	150	Окулов
6	Матрица достижимости	151	Окулов
7	Двусвязность	154	Окулов
8	Эйлеров цикл	158	Окулов
9	Гамильтонов цикл	157	Окулов
10	Фундаментальное множество циклов	160	Окулов
11	Пути – алгоритм Дейкстры	163	Окулов
12	Все пути в бесконтурном графе	164	Окулов
13	Пути – алгоритм Флойда	166	Окулов
14	Генерация независимых множеств	169	Окулов
15	Наименьшее разбиение	175	Окулов
16	Правильная раскраска	180	Окулов
17	Минимальная раскраска	181	Окулов
18	Максимальный поток в сети	187	Окулов

Продолжение таблицы 1

Вариант	Задание	Стр.	Источник
20	Поиск в глубину	121	Иванов
21	Выделение компонент связности	128	Иванов
23	Поиск остовного дерева (жадный алгоритм)	142	Иванов
24	Поиск остовного дерева (ближайший сосед)	147	Иванов
25	Кратчайший путь в орграфе	154	Иванов
26	Порождение клик графа	166	Иванов
27	Поиск фундаментальных циклов графа	174	Иванов
29	Поиск маршрутов по треугольнику	35	Иванов
33	Поиск точек выпуклой оболочки	82	Иванов
35	Нахождение максимального потока	224	Новиков
36	Выделение компонент сильной связности	227	Новиков
37	Алгоритм Флойда	229	Новиков
38	Алгоритм Дейкстры	230	Новиков
39	Алгоритм построения эйлера цикла	264	Новиков
40	Поиск независимых множеств с возвратами	274	Новиков
41	Алгоритм построения независимых множеств	276	Новиков
42	Последовательное раскрашивание графа	287	Новиков
43	Улучшенное последовательное раскрашивание графа	288	Новиков
44	Бинарный код Грея	28	Новиков
50	Алгоритм Дейкстры	395	Хэзфилд
51	Алгоритм Дейкстры для разреженных графов	396	Хэзфилд
52	Алгоритм Беллмана – Форда	397	Хэзфилд
53	Алгоритм Флойда	401	Хэзфилд
54	Алгоритм Крускала	403	Хэзфилд
55	Алгоритм Прима	405	Хэзфилд

1.3 Рекомендации по выполнению индивидуального задания

Контрольный пример выбирается произвольно. Он не должен быть крайним случаем и должен демонстрировать работу алгоритма на обычных данных. Для проверки корректности функционирования программы он должен быть решен вручную и при помощи программы. При решении вручную должны быть приведены значения всех промежуточных переменных и всех действий алгоритма.

Дополнительные контрольные примеры должны отражать поведение алгоритма на специфических входных данных. Например, пустой граф, тривиальный граф, полный граф или граф, в котором есть только вершины (без дуг или ребер); графы, содержащие в себе изолированные компоненты и т.д. Выбор дополнительных контрольных примеров может быть осуществлен самостоятельно или по рекомендации преподавателя. При выборе дополнительных контрольных примеров рекомендуется для каждого условия выбора в алгоритме сделать два контрольных примера, что бы на одном примере это условие не сработало, а на другом – сработало; возможно, объединение нескольких примеров в один, в котором срабатывает одна или другая группа условий.

Примеры, которые отсекаются специальными проверками, не являются дополнительными контрольными примерами. Например, у Вас может стоять отдельная специальная проверка, что на вход алгоритму нельзя подавать несвязанный граф. В таком случае несвязанный граф не может являться дополнительным примером – это просто пример недопустимого исходного данного. Дополнительный пример – это пример исходного данного, которое подаётся на вход алгоритма и алгоритм выдаёт результат.

В описании программы должна быть приведена таблица, в которой для каждой функции Вашей программы записано: назначение функции, список и назначение аргументов, смысл возвращаемого значения, откуда вызывается, какие функции вызывает. Также должен быть приведён граф, в котором вершины – это функции Вашей программы, а дуги – это передача управления между функциями.

2 СОДЕРЖАНИЕ ОТЧЕТА

По результатам работы составляется отчет. Ниже приводится перечень пунктов отчета.

- Титульный лист
- Реферат
- Содержание
- Введение
- Теоретическая часть
- Контрольный пример, решённый вручную
- Листинг решения контрольного примера в программе
- Описания и листинги решения дополнительных контрольных примеров, иллюстрирующих особенности поведения алгоритма на различных входных данных.
- Описание программы
- Выводы
- Список использованной литературы
- Инструкция пользователю
- Листинг программы с подробными комментариями
- Экранные формы

Образец титульного листа и листа реферата приводится в приложении.

Раздел “*Введение*” может содержать следующие пункты:

- краткое обоснование актуальности выполненного исследования (где и как оно может быть применено, в чём состоит его практическая польза);
- формулировку цели исследования;
- перечень задач исследования (3...5 задач);
- при необходимости — дополнительные сведения, например, информацию о практическом применении полученных результатов, в том числе в учебном процессе, об участии в студенческих научных конференциях, конкурсах студенческих научных работ и т.п. с данной темой.

Теоретическая часть включает описание исходных данных с примерами, текст алгоритма из книги, примеры и рисунки, поясняющие ход работы алгоритма, примеры результатов работы алгоритма с описаниями. Если у Вас возникла необходимость внести изменения в алгоритм (по сравнению с тем, как алгоритм приведён в книге), то здесь же должно быть обоснование и описание этих изменений. Теоретическая часть должна называться по названию алгоритма и начинаться со слова «Описание». Например, «Описание алгоритма Флойда для нахождения всех кратчайших путей в графе».

Ниже приведены требования к разделу «*Выводы*». Эта часть отчета должна отражать:

- заключение о достижении цели исследования и результаты решения задач, сформулированных во введении;
- рекомендации (одну-две) по результатам тестовой работы программы с обязательным указанием условностей, допускаемых в связи с учебным характером задачи;
- заключение о степени практической полезности (с указанием возможностей её повышения) изученных методов и приёмов математического моделирования и синтеза систем с заданными свойствами.

Вместе с отчетом предоставляется электронный носитель со следующей информацией. Все исходные коды программы. Должна быть возможность перекомпилировать программу. Откомпилированный вариант программы. При этом откомпилированный вариант не должен требовать для своего запуска установки каких-то специальных программ (например, Visual Studio). Файлы всех примеров, которые есть в отчёте и, возможно, каких-нибудь дополнительных примеров. Создать файл `readme.txt`, в котором описано, какой файл примера, что в себе содержит.

В инструкции пользователю должно быть приведено: описание формата файла исходных данных с примером файла и соответствующего этому файлу графа, описание формата файла листинга решения задачи, с краткими примерами, алгоритм действия пользователя для запуска программы, загрузки файла и интерпретации результатов.

3 ОБЩИЕ ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТА

Для оформления отчета по индивидуальному заданию используйте нормативную документацию: ГОСТ 7.32–2001 и ГОСТ 2.105–95.

Работа печатается на стандартных листах формата А4 (210×297 мм) с соблюдением полей:

- слева — 30 мм;
- справа — 10 мм;
- сверху — 20 мм;
- снизу — 25 мм.

Распечатанный манускрипт либо переплетается, либо надёжно сшивается (скрепляется) и помещается в папку.

Текст работы набирается, как правило, шрифтом *Times New Roman*. Размер шрифта — 14 pt, межстрочный интервал — полуторный. Отступ первой строки абзаца — 12,5 мм. Отступы слева и справа от абзаца, перед абзацем и после него отсутствуют. Висячие строки запретить. При использовании текстового процессора *Microsoft Word* для этой цели следует переопределить стиль *Обычный* в соответствии с вышеуказанными требованиями. Установить автоматический перенос слов с помощью *Разметка страницы -> Расстановка переносов*.

Каждый раздел работы всегда начинать с нового листа, для этого можно подключить режим *Вставка -> Разрыв Страницы*. Названия разделов не подчёркиваются, а записываются заглавными буквами. Нумеруются эти элементы арабскими цифрами, подзаголовки и названия параграфов могут разделяться точками, в конце точка не ставиться, например, 1.1, 1.2, 1.3. 6.2.4 Разделы, подразделы должны иметь заголовки. Заголовки разделов, подразделов следует печатать с абзацного отступа, с прописной буквы без точки в конце, не подчеркивая. Если заголовок состоит из двух предложений, их разделяют точкой.

Нумерация страниц должна быть сквозной, включая приложения. Первым считается титульный лист, но на нём номер не ставится. Нумерация остальных страниц проставляется на расстоянии 15 мм от нижнего края листа *по центру* относительно полей.

Иллюстрации (графики, схемы, компьютерные распечатки, диаграммы) следует располагать в отчете непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. На все иллюстрации должны быть даны ссылки в отчете. Иллюстрации, за исключением иллюстрации приложений, следует нумеровать арабскими цифрами сквозной нумерацией. Слово "рисунок" и его наименование располагают посередине строки. Допускается нумеровать иллюстрации в пределах раздела. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой. Например, Рисунок 1.1 - Детали прибора. При ссылках на иллюстрации при сквозной нумерации следует писать "... в соответствии с рисунком 1.2" при нумерации в пределах раздела.

Таблицы применяют для лучшей наглядности и удобства сравнения показателей. Наименование таблицы должно отражать ее содержание, быть точным, кратким. Наименование таблицы следует помещать над таблицей слева, без абзацного отступа в одну строку с ее номером через тире. Таблицу следует располагать в отчете непосредственно после текста, в котором она упоминается впервые, или на следующей странице. На все таблицы должны быть ссылки в отчете. При ссылке следует писать слово "таблица" с указанием ее номера. Таблицу допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист (страницу) слово "Таблица", ее номер и наименование указывают один раз слева над первой частью таблицы, а над другими частями также слева пишут слова "Продолжение таблицы" и указывают номер таблицы. Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Таблица должна быть обозначена "Таблица В.1",

если она приведена в приложении В. В конце заголовков и подзаголовков таблиц точки не ставят.

Примечания приводят в отчетах, если необходимы пояснения или справочные данные к содержанию текста, таблиц или графического материала. Слово "Примечание" следует печатать с прописной буквы с абзаца и не подчеркивать.

Сведения об источниках следует располагать в порядке появления ссылок на источники в тексте отчета и нумеровать арабскими цифрами без точки и печатать с абзацного отступа, на первом месте всегда стоят нормативно-правовые акты, а в конце - иностранная литература и источники Интернет. Ссылки на использованные источники следует указывать порядковым номером библиографического описания источника в списке использованных источников. Порядковый номер ссылки заключают в квадратные скобки. Нумерация ссылок ведется арабскими цифрами в порядке приведения ссылок в тексте отчета независимо от деления отчета на разделы.

Приложение оформляют после библиографического списка на последующих его листах. В тексте отчета на все приложения должны быть даны ссылки. Приложения располагают в порядке ссылок на них в тексте отчета. Каждое приложение следует начинать с новой страницы с указанием наверху посередине страницы слова "Приложение", его обозначения. Приложение должно иметь заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой. Приложения обозначают заглавными буквами русского алфавита, начиная с А, за исключением букв, З, Й, О, Ч, Ъ, Ы, Ь. После слова "Приложение" следует буква, обозначающая его последовательность.

СПИСОК ЛИТЕРАТУРЫ

1. Окулов С. М. Программирование в алгоритмах — М.: БИНОМ. Лаборатория знаний, 2002. — 341 с.
2. Иванов Б.Н. Дискретная математика. Алгоритмы и программы: Учеб. Пособие — М.: Лаборатория Базовых Знаний, 2003. — 288 с.
3. Новиков Ф.А. Дискретная математика для программиста — СПб.: Питер, 2000. — 304 с.
4. Хэзфилд Ричард, Кирби Лоуренс и др. Искусство программирования на С. Фундаментальные алгоритмы, структуры данных и примеры приложений. Энциклопедия программиста: Пер. с англ. — К.: ДиаСофт, 2001. — 736 с.

Приложение А
Титульный лист отчета

**ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
"ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ"**

Пояснительная записка
к индивидуальному заданию
по курсу "Компьютерная дискретная математика"
тема "Алгоритм Дейкстры"

Руководитель:
асс. каф. ПИ
Иванов Б.Б.
____.____.2017г.

Выполнил:
ст. гр. ПИ-16а
Петров А. И.
____.____.2017г.

Приложение Б

Лист реферат

РЕФЕРАТ

Пояснительная записка содержит: __ с., __ рис., __ приложений, __ источников.

Объектом исследования данной работы является.....

Цель работы –

СПИСОК КЛЮЧЕВЫХ СЛОВ