Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Южно-Уральский государственный университет»
(Национальный исследовательский университет)
Институт естественных и точных наук
Факультет математики, механики и компьютерных технологий
Кафедра математического и компьютерного моделирования

ОТЧЕТ ПО ПРЕДДИПЛОМНОЙ ПРАКТИКЕ
ЮУрГУ-010400.68.2017.049.001 КР

Руководитель от предприятия,
к.ф.-м.н., доцент
______________ Т.А. Макаровских
«   » ____________ 2017 г.

Автор проекта
студент группы ЕТ-224
______________ В.А. Безбородов
«   » ____________ 2017 г.

Проект защищен
с оценкой
____
«   » ____________ 2017 г.

Челябинск, 2017

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Южно-Уральский государственный университет»
(Национальный исследовательский университет)
Институт естественных и точных наук
Факультет математики, механики и компьютерных технологий
Кафедра математического и компьютерного моделирования

УТВЕРЖДАЮ
Заведующий кафедрой, д.ф.-м.н.,
доцент
_____________ Загребина С.А.
«   » __________ 2017 г.

З А Д А Н И Е

на педагогическую практику студента

Безбородова Вячеслава Александровича

Группа ЕТ-224

1. Тема работы: Практические занятия по дискретной математике.
2. Срок сдачи студентом законченной работы «   » __________ 2017 г.
3. Исходные данные к работе
   3.1. Учебное пособие по комбинаторике и теории графов [22];
   3.2. Издательская система компьютерной верстки LaTeX.
4. Перечень вопросов, подлежащих разработке
   4.1. Изучить педагогический опыт преподавания дискретной математики;
   4.2. Изучить опыт и систему воспитательной работы преподавателя;
   4.3. Овладеть методикой подготовки и проведения практического занятия;

4.4. Обучить студентов методам мышления, характерным для дискретной математики, основным понятиям комбинаторики и теории графов;

4.5. Развить навыки алгоритмического мышления;

4.6. Разработка отчетной документации (дневника), в котором отражены основные этапы работы (по дням).

5. Перечень графического материала

5.1. Бинарное дерево – 1 л.

5.2. Граф для нахождения минимального остовного дерева – 1 л.

5.3. Граф для решения задачи единого среднего – 1 л.

5.4. Граф для нахождения кратчайшего пути – 1 л.

5.5. Граф для нахождения максимального потока в сети – 1 л.

5.6. Начальный поток – 1 л.

5.7. Поток 1 – 1 л.

5.8. Поток 2 – 1 л.

5.9. Поток 3 – 1 л.

5.10. Максимальный поток – 1 л.

6. Календарный план

Наименование этапов педагогической практики	Срок выполнения этапов	Отметка о выполнении
1. Сбор материалов и литературы по теме педагогической практики	01.02.2017 г.	
2. Проведение практических занятий		
Занятие 1	11.09.2016 г.	
Занятие 2	18.09.2016 г.	
Занятие 3	25.09.2016 г.	
Занятие 4	02.10.2016 г.	
Занятие 5	09.10.2016 г.	
Занятие 6	23.10.2016 г.	
Занятие 7	30.10.2016 г.	
Занятие 8	06.11.2016 г.	
Занятие 9	13.11.2016 г.	
Занятие 10	20.11.2016 г.	
Занятие 11	27.11.2016 г.	
Занятие 12	04.12.2016 г.	
Занятие 13	11.12.2016 г.	
Занятие 14	18.12.2016 г.	
3. Оформление дневника практики	19.12.2016 г.	
4. Проверка дневника практики руководителем, исправление замечаний	20.12.2016 г.	
5. Подготовка графического материала и доклада	21.12.2016 г.	
6. Защита педагогической практики	26.02.2017 г.	

7. Дата выдачи задания «   »__________2017 г.

Заведующий кафедрой______________________________/Загребина С.А./

Руководитель работы______________________________/Т.А. Макаровских/

Студент____/В.А. Безбородов/

ОГЛАВЛЕНИЕ

Министерство образования и науки Российской Федерации
ФГАОУ ВО "Южно-Уральский государственный университет" (НИУ)
Институт "Математика, механика и компьютерные технологии"
Кафедра "Математическое и компьютерное моделирование"

ДНЕВНИК

прохождения практики (преддипломная практика)

Студент: Безбородов Вячеслав Александрович, группа ЕТ-224

Специальность: Прикладная математика и информатика (01.04.02)

Предприятие: Кафедра Математического и компьютерного моделирования ЮУрГУ

Дата прибытия на практику: 30.01.2017

Назначен: ____

(рабочее место, должность)

Дата окончания практики: 26.02.2017

Руководитель практики от предприятия:

____

(должность, Ф.И.О.)

____________________ ____________________

"расшифровка подписи"

М.П.

1. Задание на практику

Цель практики: ____

____

____

____

Задачи практики:____

____

____

____

____

____

____

Индивидуальное задание:

____

____

____

____

____

____

____

____

2. Календарный график прохождения практики

Дата	Наименование подразделения предприятия (учреждения)	Выполняемый вид работ	Рекомендации и замечания руководителя практикой от предприятия

3. Помощь производству, научно-исследовательская или рационализаторская работа студента

Содержание выполненной работы	Итог, полученный эффект

4. Производственные экскурсии

Дата	Краткое содержание, выводы

5. Характеристика работы практиканта предприятием (организацией)

Студент: Безбородов Вячеслав Александрович, группа ЕТ-224

Специальность: Прикладная математика и информатика (01.04.02)

№ п/п	Оценка важности данной компетенции обвести кружком*	Компетенция	Оценка исполнения практикантом данной компетенции обвести кружком*
1	1 2 3 4 5	Владение статистическими методами и средствами обработки экспериментальных данных	1 2 3 4 5
2	1 2 3 4 5	Способность готовить презентации, оформлять научно-технические отчеты по результатам выполненной работы, публиковать результаты исследований в виде статей и докладов на научно-технических конференциях	1 2 3 4 5
3	1 2 3 4 5	Способность находить, обобщать, анализировать, синтезировать и критически переосмысливать полученную научную, справочную, статистическую и иную информацию (продолжение формирования компетенции)	1 2 3 4 5
4	1 2 3 4 5	Способность обрабатывать и анализировать результаты научно-исследовательской работы, находить элементы новизны в разработке, представлять материалы для оформления патентов на полезные модели, готовить к публикации научные статьи и оформлять технич. отчеты	1 2 3 4 5
5	1 2 3 4 5	Способность представлять результаты проведенного исследования научному сообществу в виде статьи или доклада	1 2 3 4 5
Укажите, какими еще компетенциями, не вошедшими в список, обладает студент-практикант и оцените их			
1	1 2 3 4 5		1 2 3 4 5
2	1 2 3 4 5		1 2 3 4 5
3	1 2 3 4 5		1 2 3 4 5
Укажите, какие ещё компетенции Вы хотели бы включить в список и оцените их в отношении данного практиканта:			
1			
2			
3			

* необходимо обвести кружком только одну оценку от "1" - совершенно не важно или совершенно не удовлетворен …… до "5" - очень важно или полностью удовлетворен.

Сведения о рецензенте:

Ф.И.О. ______________________________

Должность ______________________________

Уч. звание ______________________________

Уч. степень ______________________________

______________ ______________

"расшифровка подписи"

Введение

Практика (преддипломная) предусмотрена как один из компонентов основной образовательной программы подготовки магистров.

Преддипломная практика в системе высшего образования является обязательным компонентом профессиональной подготовки. Практика предшествует написанию магистерской диссертации и является логической завершающей ступенью обучения после прохождения основных теоретических дисциплин.

Практика проводилась на базе Южно-Уральского государственного университета, руководитель практики: к.ф.-м.н., доцент Т.А. Макаровских.

Цели преддипломной практики:

- осмысление сущности и целостности педагогического процесса,
- подготовка к преподавательской деятельности в ВУЗе,
- овладение основами учебно-методической и воспитательной работы,
- формирование педагогических навыков и умений ведения практических занятий по дискретной математике.

Задачи практики:

- изучить педагогический опыт преподавания дискретной математики;
- изучить опыт и систему воспитательной работы преподавателя;
- овладеть методикой подготовки и проведения практического занятия.

Практика проводилась в период с 30.01.2017 по 26.02.2017.

Работа состоит из введения, 4 глав, заключения, 1 приложения и списка литературы. Объем работы составляет 52 страниц. Список литературы содержит 22 наименование.

В первой главе даются общие положения по педагогической практике магистров.

Во второй главе описывается план практических занятий.

В третьей главе приводится содержание всех проведенных практических занятий с краткими комментариями, самостоятельных работ и примеров решения типичных задач.

В заключении перечислены основные результаты прохождения педагогической практики.

1 Симплекс-метод

1.1 Введение

Линейное программирование (ЛП) является широко распространенной техникой решения оптимизационных задач различных областей науки. Сегодня используются два основных подхода к решению задач линейного программирования (ЗЛП) – симплекс-метод и методы внутренней точки. В случаях, когда необходимо решать семейства взаимосвязанных ЗЛП (целочисленное программирование, методы разложения, некоторые классы задач ЛП), симплекс-метод обычно более эффективен [10].

Возможность распараллелить симплекс-метод для решения ЗЛП рассматривалась с 1970-х гг., хотя первые попытки разработать практические реализации предпринимались только с начала 1980-х гг. Наиболее плодотворным для решения этой проблемы оказался период с конца 1980-х до конца 1990-х гг. Также было несколько экспериментов использования векторной обработки и ЭВМ с общей разделяемой памятью; подавляющее большинство реализаций строилось на мультипроцессорах с распределенной памятью и сетевых кластерах [10].

1.2 Сиплекс-метод

Симплекс-метод и его вычислительные требования удобнее обсуждать в контексте стандартной формы ЗЛП

$$
\begin{aligned}
&c^T x \to \min \\
&Ax = b \\
&x \geq 0,
\end{aligned}
\tag{1.1}
$$

где $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$. Матрица A в (1.1) обычно содержит столбцы с единицами, соответствующими фиктивным переменным, возникающим при переводе ограничений-неравенств в равенства. Оставшиеся столбцы A соответствуют обычным переменным.

В симплекс–методе индексы переменных подразделяются на два подмножества: подмножество $\mathcal{B}$, соответствующее m базисным переменным x_B, и подмножество $\mathcal{N}$, соответствующее $n-m$ небазисным переменным x_N. При этом базисная матрица B, составленная из соответствующих $\mathcal{B}$ столбцов A, является невырожденной. Множество $\mathcal{B}$ условно называют базисом. Соответствующие $\mathcal{N}$ столбцы A формируют матрицу N. Компоненты c, соответствующие $\mathcal{B}$ и $\mathcal{N}$, называют базисными c_B и небазисными c_N издержками соответственно [10].

Когда небазисные переменные нулевые, значения $\hat{b} = B^{-1}b$ базисных переменных соответствуют вершинам допустимого региона при условии, что они неотрицательны. Выражение $x_B + B^{-1}N = \hat{b}$, следующее из (1.1), позволяет убрать базисные переменные из целевой функции, которая становится равной $(c_N^T - c_B^T B^{-1}N)x_N + c_B^T\hat{b}$. Если все компоненты вектора альтернативных издержек $\hat{c}_N = c_N^T - c_B^T B^{-1}N$ неотрицательны, тогда текущий базис оптимален.

Если текущий базис неоптимален, на каждой итерации симплекс-метода для ввода в базис выбирается имеющая отрицательное значение альтернативной издержки небазисная переменная x_q. Увеличение этой переменной от нуля при выполнении условий (1.1) соответствует перемещению вдоль ребра допустимого региона в направлении уменьшения значения целевой функции. Направление этого ребра определяется соответствующим x_q столбцом $\hat{a}_q$ при $\hat{N} = B^{-1}N$. При просмотре отношений компонентов вектора $\hat{b}$ к соответствующим положительным компонентам $\hat{a}_q$ находится первая базисная переменная, которая обнулится при росте x_q и, следовательно, шаг к следующей точке допустимого региона вдоль этого ребра.

Существует много стратегий выбора переменной x_q для ввода в базис. Первоначальное правило выбора переменной с наименьшей альтернативной издержкой известно как критерий Данцига [4]. Хотя, если компоненты $\hat{a}_j$ намного превосходят компоненты $\hat{c}_j$, то только небольшое увеличение x_j возможно до обращения одной из базисных переменных в ноль. Альтернативные ценовые стратегии взвешивают альтернативную издержку путем деления на длину $\hat{a}_j$. Точная стратегия наиболее крутого ребра [9] вводит понятие весов $s_j = 1 + ||\hat{\alpha}_j||^2$, соответствующих длине шага при единичном изменении x_j. Практический (приближенный) метод наиболее крутого ребра [7] и Devex стратегия [11] вычисляют приближенное значение весов. При использовании этих подходов количество итераций, необходимых для решения ЗЛП на практике может быть оценено как $O(m+n)$, и теоретически нет препятствий для достижения сложности $O(2^n)$.

Популярной техникой выбора выводимой из базиса переменной является процедура EXPAND [14]. Посредством небольшого расширения ограничений, эта стратегия часто позволяет выбрать выводимую переменную из числа возможных на основании численной стабильности.

Два главных варианта симплекс-метода соответствуют различным пониманиям того, какие данные требуются для определения шага к новой точке. Первый вариант – *стандартный симплекс-метод*, в котором альтернативные издержки и направления всех ребер в текущей точке определяются прямоугольной таблицей. В *обратном симплекс-методе* альтернативные издержки и направление выбранного ребра определяются путем решения систем с базисной матрицей B.

1.3 Обычный симплекс-метод

В обычном симплекс-методе матрица $\hat{N}$, правый вектор-столбец $\hat{b}$, альтернативные издержки $\hat{c_N}$ и текущее значение целевой функции $f = c_B^T\hat{b}$ располагаются в виде таблицы следующей формы.

	$\mathcal{N}$	RHS
$\mathcal{B}$	$\hat{N}$	$\hat{b}$
	$\hat{c}_N^T$	$-\hat{f}$

На каждой итерации обычного симплекс-метода для перехода к новому базису к колонкам этой таблицы применяется процедура преобразований Жордана-Гаусса.

Выполнение симплекс-метода начинается с базиса $B = E$, в то время как в таблице симплекса записана матрица N. Это означает, что таблица является разреженной. Принято считать, что степень заполненности матрицы в процессе выполняемых преобразований такова, что нет необходимости использовать разреженные структуры данных, поэтому обычный симплекс-метод часто реализуют без их использования.

Обычный симплекс-метод по природе своей численно нестабилен, поскольку использует длинную цепочку операций последовательного исключения переменных, выбирая колонку поворота согласно алгоритму, а не из соображений стабильности вычислений. Если алгоритм получает плохо обусловленные базисные матрицы, любая подпоследовательность таблицы, соответствующая хорошо обусловленному базису, скорее всего будет содержать численные ошибки, вызванные плохой обусловленностью на ранних этапах. Это может привести к такому выбору вводимой или выводимой переменной, что при точных вычислениях целевая функция не будет убывать монотонно, ограничения будут нарушены либо базисная матрица станет вырожденной. Для большей надежности необходимо отслеживать возникающие в таблице ошибки и, при необходимости, выполнять полный ее пересчет численно стабильным способом. Проверка ошибок может осуществляться сравнением обновленных альтернативных издержек со значением, полученным напрямую с использованием колонки поворота и базисных издержек. Поскольку операции с матрицей, обратной базисной, могут быть выполнены посредством использования соответствующих ячеек таблицы, вычисление колонки поворота напрямую и сравнение ее с ячейками таблицы может предоставить более надежный (но и более затратный) механизм проверки ошибок.

1.4 Обратный сиплекс-метод

Вычислительные этапы обратного симплекс-метода представлены в таблице 1. Вначале каждой итерации полагается, что вектор альтернативных издержек $\hat{c}_N$ и вектор $\hat{b}$ текущих значений базисных переменных известны, и представление B^{-1} доступно. Первым шагом алгоритма является `CHUZC`, который сканирует (взвешенные) альтернативные издержки, чтобы определить хорошего кандидата q для ввода в базис. *Колонка поворота* $\hat{a}_q$ формируется на шаге `FTRAN`, используя представление B^{-1}.

CHUZC: Выбрать из $\hat{c}_N$ хорошего кандидата q для ввода в базис.
FTRAN: Сформировать колонку поворота $\hat{a}_q = B^{-1}a_q$, где a_q – колонка q матрицы A.
CHUZR: Из отношений $\hat{b}_i/\hat{a}_{iq}$ определить номер p строки хорошего кандидата для вывода из базиса.
Положить $\alpha = \hat{b}_p/\hat{a}_{pq}$.
Обновить $\hat{b} := \hat{b} - \alpha\hat{a}_q$.
BTRAN: Сформировать $\pi_p^T = e_p^T B^{-1}$.
PRICE: Сформировать строку поворота $\hat{a}_p^T = \pi_p^T N$.
Обновить альтернативные издержки $\hat{c}_N^T := \hat{c}_N^T - \hat{c}_q\hat{\alpha}_p^T$.
Если {рост в представлении B^{-1}}, **тогда**
INVERT: Сформировать новое представление B^{-1}.
иначе
UPDATE: Обновить представление B^{-1} в соответствии с изменением базиса.
конец если

Таблица 1 — Итерация обратного симплекс-метода

На шаге CHUZR определяется выводимая из базиса переменная. Индекс p показывает, в какой строке расположена выводимая переменная, а сама строка именуется *строкой поворота*. Индекс самой переменной обозначается как p'. Как только индексы q и p' меняются местами между множествами $\mathcal{B}$ и $\mathcal{N}$, говорят, что произошло *изменение базиса*. После этого, правый вектор-столбец $\hat{b}$ обновляется в соответствии с увеличением $\alpha = \hat{b}_p/\hat{a}_{pq}$ в x_q.

Перед выполнением следующей операции необходимо получить значения альтернативных издержек и представление новой матрицы B^{-1}. Хотя альтернативные издержки могут быть вычислены и напрямую, используя выражения

$$\pi_B^T = c_B^T B^{-1};\ \hat{c}_N^T = c_N^T - \pi_B^T N,$$

в вычислительном смысле гораздо эффективнее обновлять их с помощью строки поворота $\hat{a}_p^T = e_p^T B^{-1} N$ из таблицы стандартного симплекса. Это выполняется в два шага. Сначала, используя представление B^{-1}, на шаге BTRAN формируется вектор $\pi_p^T = e_p^T B^{-1}$, а затем строится вектор $\hat{a}_p^T = \pi_p^T N$ значений строки поворота (шаг PRICE). Как только были получены значения альтернативных издержек, шак UPDATE изменяет представление B^{-1} в соответствии с изменением базиса. Необходимо отметить, что из соображений эффективности и численной стабильности, периодически необходимо находить новое представление B^{-1} с помощью операции INVERT.

При применении стратегии Devex [11], строка поворота, вычисленная для обновления альтернативных издержек, используется также для обновления Devex весов при незначительных вычислительных затратах. Для обновления точных весов в методе наиболее крутого ребра в дополнение к строке поворота требуется дополнительный шаг BTRAN для вычисления $\hat{a}_q^T B^{-1}$ и шаг PRICE для получения результата матричного умножения этого вектора и матрицы N. Также вычислительно неэффективно инициализировать значения весов наиболее крутого ребра, если начальная базисная матрица не единичная. Как следствие этих дополнительных затрат и поскольку стратегия Devex работает хорошо в плане уменьшения количества итераций, необходимых для решения ЗЛП, эта стратегия обычно используется в эффективных последовательных реализациях обратного симплекс-метода.

2 Параллельный симплекс-метод

2.1 Параллельные вычисления

Прежде чем переходить к вопросу распараллеливания симплекс-метода, необходимо рассмотреть некоторые термины и концепции из области параллельного программирования. В этом разделе представлен краткий обзор необходимых понятий. Полное и более общее введение в параллельные вычисления можно найти в [12].

Классифицируя архитектуры параллельных мультипроцессоров, необходимо понимать важное отличие между *распределенной памятью*, когда каждый процессор имеет свою собственную локальную память, и *общей памятью*, когда все процессоры имеют доступ к общей разделяемой памяти. Современные мощные ЭВМ могут состоять из множества распределенных кластеров, каждый из которых может иметь множество процессоров с общей памятью. На более простых мультипроцессорах память может быть либо общей, либо разделяемой.

Обычно успешность распараллеливания измеряется в терминах *ускорения* – отношения времени, необходимого для решения задачи с использованием более одного процессора, ко времени решения задачи на одном процессоре. Традиционной является цель достичь фактор ускорения, равный количеству подключаемых процессоров. Такой фактор называется *линейным ускорением* и соответствует 100% *параллельной эффективности*. Увеличение доступной кэш-памяти и оперативной памяти одновременно с количеством процессоров иногда приводит к феномену *сверхлинейного ускорения*. Схемы распараллеливания, для которых (по крайней мере в теории) производительность растет линейно без ограничений с ростом количества подключаемых процессоров, называются *масштабируемыми* схемами. Если параллелизм не используется во всех главных операциях алгоритма, то ускорение, в соответствии с законом Амдала [1], ограничено долей времени выполнения непараллельных операций.

Существуют две основных парадигмы параллельного программирования. Если работа большинства операций алгоритма может быть распределена среди множества процессоров, тогда говорят о *параллелизме по данным*. В противоположность этому, если возможно выполнять несколько главных операций алгоритма одновременно, тогда имеет место *параллелизм по задачам*. На практике возможно применять одновременно оба подхода для определенного набора главных операций алгоритма.

Есть два фундаментальных способа реализации алгоритмов на параллельных ЭВМ. На машинах с распределенной памятью передача данных между процессорами осуществляется посредством инструкций, порожденных явными вызовами методов *передачи сообщений*. На машинах с общей разделяемой памятью применяется *параллелизм по данным*, когда инструкции записываются как для последовательного исполнения, но транслируются специальным компилятором в параллельный код. Большинство протоколов передачи сообщений также поддерживаются на ЭВМ с общей памятью, равно как и распараллеливание по данным возможно на ЭВМ с распределенной памятью.

На машинах с распределенной памятью, накладные расходы на передачу сообщений между процессорами определяются *задержкой* и *пропускной способностью канала*. Первое – это время передачи, не зависящее от размера сообщения, а второе – это скорость связи. Для общих протоколов передачи сообщений задержка и пропускная способность на определенной архитектуре может быть значительно выше, чем в независящей от архитектуры среде, что обычно регулируется и настраивается поставщиком. Если алгоритму для вычислений необходим интенсивный обмен информацией, растущие накладные расходы на связь могут перевесить любые улучшения от использования дополнительных процессоров.

2.2 Распараллеливание симплекс-метода

Существующие подходы к распараллеливанию симплекс-метода и ему подобных удобно классифицировать по виду симплекс-метода и по использованию разреженных типов данных. Такая классификация позитивно коррелирует с практической ценностью реализации в контексте решения ЗЛП и негативно с успешностью этих подходов в достигнутом ускорении.

Некоторые из рассматриваемых ниже схем предлагают неплохое ускорение относительно эффективных последовательных решателей своего времени. Другие только кажутся неэффективными в свете последовательного обратного симплекса, который к тому моменту либо был малоизвестен, либо был разработан впоследствии. Такие случаи определяются ниже, чтобы подчеркнуть, что в результате огромного увеличения эффективности последовательного обратного симплекс-метода (как во время исследований в области распараллеливания симплекса, так и после) проблема разработки практического параллельного симплекс-решателя стала очень актуальной.

2.2.1 Параллельный симплекс без использования разреженных матриц

Стандартный и обратный симплекс-методы без использования алгебры разреженных матриц реализовывались неоднократно. Простота обычных структур данных и потенциал достичь линейного ускорения делают их привлекательными для применения в параллельных вычислениях. Хотя при решении общих разреженных ЗЛП больших размерностей такие реализации малоэффективны, поскольку они могут соперничать с эффективными последовательными реализациями обратного симплекс-метода, использующими разреженные структуры, только при подключении значительного числа процессоров.

Первые работы в этом направлении ограничиваются обсуждением схем распределения данных и коммуникации; реализации ограничиваются небольшим числом процессов на ЭВМ с распределенной памятью (краткие обзоры даются в [17, 20], примеры других ранних работ можно найти в [2, 6, 8, 21]). В одной из относительно ранних работ [16], в которой были реализованы обычный и обратный симплекс-методы на 16-процессорном Intel hypercube, достигнутое ускорение варьируется от 8 до 12 для небольших задач из библиотеки Netlib. В [3] сообщается о 12-кратном ускорении при решении двух небольших ЗЛП с использованием обычного симлекс-метода на 16-процессорной ЭВМ с *общей разделяемой памятью*. Также были случаи получения более чем 12-кратного ускорения [13].

В [5] разработаны параллельный обычный и обратный симплекс-методы с применением метода наиболее крутого ребра [9] и протестированы на машинах Connection Machine CM-2 и CM-5 с массовым параллелизмом. Решая некоторые ЗЛП средней размерности из Netlib и очень плотные задачи машинного обучения, ускорение между 1.6 и 1.8 быо достигнуто только при удвоении число процессоров. В [18] также используется метод наиболее крутого ребра и обычный симплекс-метод на ЭВМ MasPar MP-1 и MP-2. Решая в основном случайно сгенерированные ЗЛП большой размерности, авторы достигали практически троекратного ускорения. Одной из более поздних работ по реализации параллельного обычного симплекс-метода с запуском на небольшом количестве процессоров является [19].

Работы по созданию небольших параллельных реализаций обычного симплекс-метода без применения разреженных структур данных продолжаются. Были представлены результаты реализации на 8 процессорах с 5-кратным ускорением при решении небольших случайных ЗЛП [15].

2.2.2 Параллельный симплекс с использованием разреженных матриц

Особую сложность в разработке действительно хорошего в практическом смысле параллельного симплекс-метода является применение эффективных техник работы с разреженными матрицами. Разработанный параллельный решатель будет конкурентноспособным по отношению к хорошей последовательной реализации только тогда, когда решение общих разреженных ЗЛП большой размерности будет затрагивать разумное число процессоров.

В период, когда распараллеливание симплекс-метода только начиналось и широко дискутировалось, практические параллельные методы факторизации и решения разреженных асимметричных СЛАУ были только на стадии становления. Как следствие, несмотря на то, что в симплекс-методе с плотными матрицами внедрение параллелизма проходило успешно, было распространено мнение, что использование разреженных матриц сильно ограничивает возможности распараллеливания (за исключением `PRICE`). Некоторых это наводило на мысль, что разработать хорошую параллельную реализацию невозможно в принципе. Но несмотря на преимущественно последовательную природу компонентов обратного симплекс метода, все еще существуют возможности применения параллелизма по задачам.

2.3 Заключение

Попытки использования параллелизма в симплекс-методе были связаны со многими ведущими членами разработки эффективного последовательного обратного симплекс метода. То, что относительный успех в этой области оказался довольно ограниченным, объясняется трудоемкостью задачи.

Необходимо отметить, что попытки внедрения параллелизма в обычный симплекс-метод для общих разреженных ЗЛП значительно улучшили его производительность по сравнению с хорошей последовательной реализацией обратного симплекс-метода. Параллельные обычный или обратный симплекс-методы без использования разреженных структур данных несостоятельны без привлечения значительного числа процессоров. Внедрение параллелизма по задачам также было ограничено численной нестабильностью и большими накладными расходами на передачу сообщений в ЭВМ с распределенной памятью.

3 План практических занятий

При разработке плана практических занятий необходимо не забывать о разных сторонах одного процесса: что нужно делать, чтобы спокойно преподавать и что нужно сделать, чтобы курсы были хорошие в смысле качества и объема предлагаемого материала.

Федеральный закон "Об образовании в Российской Федерации"№ 273-ФЗ от 29 декабря 2012 года с изменениями 2015-2016 года гласит, что «организация образовательного процесса в образовательном учреждении регламентируется учебным планом (разбивкой содержания образовательной программы по учебным курсам, по дисциплинам и по годам обучения), годовым календарным учебным графиком и расписаниями занятий, разрабатываемыми и утверждаемыми образовательным учреждением самостоятельно».

Это, в свою очередь, означает, что любая учебная программа курса (или учебный план дисциплины) должна иметь некоторый обязательный набор характеристик, который опишем ниже.

Название. Практические занятия по комбинаторике и теории графов.

Выходные данные автора программы. Макаровских Татьяна Анатольевна, доцент, кандидат физико-математических наук (2006).

Требования к слушателям. Студенты специальностей "Прикладная информатика в экономике "Математические методы в экономике"Южно-Уральского государственного университета.

Характеристика изучаемой дисциплины и ее место в системе образования. Дискретная математика — часть математики, изучающая дискретные математические структуры, такие, как графы и утверждения в логике. В рамках учебных программ дискретная математика обычно рассматривается как совокупность разделов, связанных с приложениями к информатике и вычислительной технике: теория функциональных систем, теория графов, теория автоматов, теория кодирования, комбинаторика, целочисленное программирование. Дискретная математика и примыкающие к ней дисциплины изучаются во всех университетах, где осуществляется подготовка специалистов в областях программирования, математики, экономики, а также по техническим и гуманитарным дисциплинам.

Формат занятий и виды контроля. Практические занятия по дисциплине подразумевают повторение и закрепление знаний, полученных в ходе лекций. Каждое занятие должно включать в себя проверку домашнего задания, работу у доски и разбор нового материала. Виды контроля включают в себя *регулярный* (минитесты на каждом практическом занятии по предыдущей теме, 5-7 мин.), *промежуточный* (проведение 3-х контрольных по завершении соответствующих блоков) и *итоговый* (экзамен).

Содержание дисциплины. Программа курса включает в себя решение практических задач по следующим основным разделам.

1) Множества и операции над ними:
   - множества;
   - способы задания множеств;
   - операции над множествами;
   - свойства операций над множествами.
2) Метод математической индукции:
   - решение практических задач.
3) Основные принципы комбинаторики:
   - правило произведения;
   - правило сложения.
4) Размещения, перестановки, сочетания:
   - выборки и размещения;
   - сочетания;
   - перестановки с повторениями;
   - полиномиальная формула.
5) Комбинаторные тождества:
   - решение практических задач.
6) Формирование перестановок и сочетаний:
   - перестановки;
   - сочетания;

- размещения без повторений;
- сочетания с повторениями.

7) Принцип включения-исключения:
   - решение практических задач.
8) Введение в теорию графов, основные понятия и определения:
   - основные определения;
   - лемма о рукопожатиях;
   - вершинная и реберная связность.
9) Способы представления графов и методы просмотра вершин:
   - матрица инцидентности;
   - матрица смежности;
   - списки смежности;
   - поиск в глубину;
   - поиск в ширину.
10) Деревья и леса:
    - числовые параметры, характеризующие ориентированное дерево;
    - бинарные деревья;
    - сортировка;
    - бинарные деревья поиска;
    - остовные деревья;
    - матричная формула Кирхгофа.
11) Эйлеровы и гамильтоновы графы:
    - эйлеровы графы и задача о Кенигсбергских мостах;
    - гамильтоновы графы и задача коммивояжера;
    - связь между эйлеровыми и гамильтоновыми циклами.
12) Двудольные графы и паросочетания:
    - двудольные графы;
    - паросочетания;
    - задача о назначениях.
13) Укладки графов:
    - свойства планарных графов;

- формула Эйлера;
- критерий планарности графа;
- алгоритм укладки графа на плоскости.

14) Нахождение кратчайших путей в графе:
    - решение практических задач.
15) Задачи сетевого планирования:
    - правила построения сетевых графиков;
    - метод критического пути;
    - управление проектами с неопределенным временем выполнения работ;
    - оптимизация сетевого графика.
16) Потоки в сетях:
    - алгоритм Форда и Фалкерсона;
    - метод блокирующих потоков.
17) Раскраска графов:
    - точный алгоритм раскрашивания;
    - приближенный алгоритм последовательного раскрашивания;
    - улучшенный алгоритм последовательного раскрашивания;
    - клики и независимые множества.

Учебно-методическое обеспечение дисциплины. Для лучшего усвоения курса рекомендуется использовать следующую литературу: Панюкова, Т. Комбинаторика и теория графов: Учебное пособие. Изд. 3-е, испр. / Т.А. Панюкова. – М.: ЛЕНАНД, 2014. – 2016 с.

4 Дневник практики

В этом разделе приводится содержание всех проведенных практических занятий с краткими комментариями, самостоятельных работ и примеров решения типичных задач.

Занятие 1

Тема занятия. Введение в дисциплину. Определение понятий: дискретная математика, множество. Таблицы истинности логических операций: конъюнкция, дизъюнкция, исключающее или, отрицание, импликация. Операции над множествами: объединение, объединение семейства, пересечение, пересечение семейства, разность, симметрическая разность, декартово произведение, декартова степень. Свойства операций над множествами: законы идемпотентности, двойное дополнение, законы де Моргана, коммутативность, ассоциативность, дистрибутивность, свойства тождества, свойства дополнения. Алгоритм нахождения всех положительных делителей данного числа.

Внутриклассные задания.[1] 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.10.

Задания на дом. 1.9, 1.11, 1.13, 1.14, 1.15, 1.16.

Примеры задач

Задача. Привести примеры задания множеств с помощью характеристического признака.

Решение.

- $\{(a, b, c) \mid a^2 + b^2 = c^2,\ a, b, c \in \mathbb{N},\ a, b, c > 0\}$ – множество всех Пифагоровых троек.
- $\{a^2 \mid a \in \mathbb{N},\ a \vdots 2\}$ – множество всех точных четных квадратов.

Задача. Пусть $A = \{2, 3, 5, 6, 9\}$, $B = \{10, 5, 2, 6\}$. Определить $A \cup B$, $A \cap B$, $A \backslash B$, $\overline{A}$, $\overline{B}$.

Решение.

[1] Согласно учебнику [22].

- $A \cup B = \{2, 3, 5, 6, 9, 10\}$
- $A \cap B = \{2, 5, 6\}$
- $A \backslash B = \{3, 9\}$
- $\overline{A} = \{1, 4, 7, 8\}$ (при $U = \{1, \ldots, 9\}$)
- $\overline{B} = \{1, 3, 4, 7, 8, 9\}$ (при $U = \{1, \ldots, 10\}$)

Занятие 2

Тема занятия. Метод математической индукции: база индукции, индукционный шаг. Основные принципы комбинаторики: правило произведения, правило сложения.

Внутриклассные задания. 2.1, 2.2, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.12, 3.14, 3.17, 3.18, 3.20, 3.21, 3.23, 3.24.

Задания на дом. 2.3, 2.4, 3.11, 3.19, 3.22, 3.25.

Самостоятельная работа

Вариант 1.

1) Определение дискретной математики.
2) Таблица истинности для исключающего или.
3) Разность множеств.
4) Ассоциативность.

Вариант 2.

1) Определение множества.
2) Таблица истинности для импликации.
3) Симметрическая разность множеств.
4) Дистрибутивность.

Примеры задач

Задача. С помощью метода математической индукции доказать, что

$$1 + 2 + 2^2 + \ldots + 2^{n-1} = 2^n - 1.$$

Решение. Пусть утверждение $P(k)$ истинно, т.е.

$$1 + 2 + 2^2 + \ldots + 2^{k-1} = 2^k - 1 \qquad \text{(верно)}.$$

Докажем, что $P(k+1)$ истинно.

$$P(k+1) = 1 + 2 + 2^2 + \ldots + 2^{k-1} + 2^k = 2^k - 1 + 2^k = 2 \cdot 2^k - 1 = 2^{k+1} - 1.$$

Что и требовалось доказать.

Задача. В меню столовой 3 первых блюда, 5 вторых и 3 третьих. Сколькими способами можно выбрать обед из трех блюд (первое, второе и третье)?

Решение. По правилу произведения имеем

$$3 \cdot 5 \cdot 3 = 45.$$

Обед из трех блюд можно выбрать 45 способами.

Занятие 3

Тема занятия. Размещения, перестановки, сочетания, полиномиальная формула, бином Ньютона.

Внутриклассные задания. 4.1, 4.2, 4.4, 4.7, 4.8, 4.10, 4.11, 4.12, 4.14, 4.15, 4.16, 4.17, 4.19, 4.21, 4.22, 4.23, 4.24.

Задания на дом. 4.3, 4.5, 4.6, 4.9, 4.13, 4.18.

Самостоятельная работа

Вариант 1.

1) С помощью метода математической индукции доказать, что

$$\frac{1}{1 \cdot 3} + \frac{1}{3 \cdot 5} + \ldots + \frac{1}{(2n-1)(2n+1)} = \frac{n}{2n+1}.$$

2) Сколько положительных чисел, меньших 700, делятся на 5?

Вариант 2.

1) С помощью метода математической индукции доказать, что

$$1 + r + r^2 + \ldots + r^{n-1} = \frac{1 - r^n}{1 - r}.$$

2) Сколькими способами можно 8 человек поставить в очередь, если на первых 3-х местах могут стоять только 3 VIP-клиента в произвольном порядка?

Примеры задач

Задача. В однокруговом турнире по футболу участвует 8 команд. Сколько всего матчей будет сыграно?

Решение. Вычислим число сочетаний из 8 по 2.

$$C_8^2 = \frac{8!}{2!(8-2)!} = \frac{8!}{2! \cdot 6!} = \frac{8 \cdot 7 \cdot 6!}{2 \cdot 6!} = 28.$$

Всего будет сыграно 28 матчей.

Задача. Найти коэффициент при $x^k y^m$ в разложении $(1+x+y)^n$, $k+m \leq n$.

Решение. Используя полиномиальную формулу, имеем

$$(1 + x + y)^n = \sum_{n_1+n_2+n_3=n} \frac{n!}{n_1! n_2! n_3!} x^{n_2} y^{n_3},$$

$$n_2 = k, \ \ n_3 = m, \ \ \frac{n!}{k! m! (n-k-m)!}.$$

Занятие 4

Тема занятия. Комбинаторные тождества.

Внутриклассные задания. 5.2, 5.4, 5.5.

Задания на дом. 5.6.

Самостоятельная работа

Вариант 1.

1) Формула числа сочетаний без повторений.

2) Полиномиальная формула.

3) В разложении $(3x-4y)^{13}$, найти коэффициент при множителе x^8y^5.

Вариант 2.

1) Формула числа перестановок с повторениями.

2) Формула бинома Ньютона.

3) В разложении $(2x+3y)^{10}$, найти коэффициент при множителе x^6y^4.

Примеры задач

Задача. Доказать равенство

$$\sum_{k=0}^{n}\frac{1}{k+1}C_n^k=\frac{2^{n+1}-1}{n+1}.$$

Решение.

$$\sum_{k=0}^{n}\frac{1}{k+1}C_n^k=\left[k\cdot C_n^k=n\cdot C_{n+1}^{k-1}\right]=$$

$$=\sum_{k=0}^{n}\frac{1}{n+1}C_{n+1}^{k+1}=\frac{1}{n+1}\sum_{k=0}^{n}C_{n+1}^{k+1}=$$

$$=\left[\sum_{k=0}^{n}C_n^k=2^n\right]=\frac{1}{n+1}\left(2^{n+1}-1\right).$$

Что и требовалось доказать.

Задача. Доказать равенство

$$\sum_{k=0}^{n}k\cdot C_n^k=n\cdot 2^{n-1}.$$

Решение.

$$\sum_{k=0}^{n}k\cdot C_n^k=0\cdot C_n^0+\sum_{k=1}^{n}k\cdot C_n^k=$$

$$=\left[k\cdot C_n^k=n\cdot C_{n+1}^{k-1}\right]=\sum_{k=1}^{n}n\cdot C_{n-1}^{k-1}=$$

$$= n\sum_{k=1}^{n} C_{n-1}^{k-1} = [t = k-1] =$$

$$= n\sum_{t=0}^{n-1} C_{n-1}^{t} = \left[\sum_{k=0}^{n} C_n^k = 2^n\right] = n\cdot 2^{n-1}.$$

Что и требовалось доказать.

Занятие 5

Тема занятия. Формирование перестановок и сочетаний.

Внутриклассные задания. 6.1, 6.2, 6.5, 6.6, 6.7, 6.8.

Задания на дом. 6.3, 6.4.

Самостоятельная работа

Вариант 1.

1) Доказать, что

$$\sum_{k=0}^{n} (-1)^k \frac{C_n^k}{k+1} = \frac{1}{n+1}.$$

Вариант 2.

1) Доказать, что

$$\sum_{i=0}^{k} C_{n+i-1}^{i} = C_{n+k}^{k}.$$

Примеры задач

Задача. Определить перестановку, следующую за 21435.

Решение. Согласно алгоритму ПЕРЕСТАНОВКА имеем:

$$\begin{array}{ccccccc} & & 1 & 2 & 3 & \boxed{4} & 5 & \\ P & (& 2 & 1 & 4 & \underline{3} & 5 &) \end{array}$$

1) $m = 4$

2) $a_i = 5$

3) $214\boxed{5}3$

4) 21453

Следующая перестановка 21453.

Задача. Дано множество $S = \{1, 2, 3, 4, 5\}$. Определить сочетание, следующее за $S_1 = \{1, 3, 4\}$.

Решение. Согласно алгоритму СОЧЕТАНИЕ имеем:

1) $S = \{1, 2, 3, 4, 5\}$, $S_1 = \{1, 3, 4\}$
2) 10110
3) 10111
4) $S_2 = \{1, 3, 4, 5\}$

Следующее сочетание $\{1, 3, 4, 5\}$.

Занятие 6

Тема занятия. Принцип включения-исключения. Введение в теорию графов. Основные понятия и определения. Принцип Дирихле.

Внутриклассные задания. 7.3, 7.4, 7.5, 7.6.

Задания на дом. 7.7, 7.8.

Самостоятельная работа

Вариант 1.

1) Определить перестановку, следующую за 13765.
2) Дано множество $S = \{1, 2, 3, 4, 5, 6\}$. Определить сочетание, следующее за $S_1 = \{1, 4, 5, 6\}$.

Вариант 2.

1) Определить перестановку, следующую за 42018.
2) Дано множество $S = \{1, 2, 3, 4, 5, 6\}$. Определить сочетание, следующее за $S_1 = \{1, 3, 4, 5\}$.

Примеры задач

Задача. Сколько существует четырехзначных чисел, которые не делятся ни на 3, ни на 5, ни на 7?

Решение. Четырехзначных чисел, которые делятся на 3 – 2999, на 5 – 1799, на 7 – 1285, на 3 и 5 – 599, на 3 и 7 – 428, на 5 и 7 – 257, на 3, 5 и 7 – 85. Тогда по принципу включения-исключения

$$X = 8999 - (2999 + 1799 + 1285) + (599 + 428 + 257) - 85 =$$

$$= 8999 - 6083 + 1284 - 85 = 4115.$$

Существует 4115 четырехзначных чисел, которые не делятся ни на 3, ни на 5, ни на 7.

Задача. Пусть в простом графе никакие две вершины одинаковой степени не соединены путем длины 2. Доказать, что в графе имеется висячая вершина.

Решение. Докажем от противного. Пусть граф является простым, и в нем нет ни изолированных, ни висячих вершин. Рассмотрим вершину V максимальной степени k. Она соединена ребрами с k попарно различными вершинами. Их степени могут принимать значения от 2 до k. По принципу Дирихле, среди этих значений есть одинаковые. Значит, существует 2 вершины равной степени, соединенной путем длины 2 (через V). Получили противоречие. Что и требовалось доказать.

Занятие 7

Тема занятия. Способы представления графов и методы просмотра вершин. Матрица инцидентности, матрица смежности, список смежности. Поиск в глубину, поиск в ширину. Деревья и леса.

Внутриклассные задания. 8.1, 8.2, 8.3, 8.4, 8.5, 8.8, 8.9, 8.13.

Задания на дом. 8.6, 8.7, 8.10, 8.11, 8.12.

Самостоятельная работа

Вариант 1.

1) Степень вершины.
2) Цикл.
3) Простой путь.
4) Связный неориентированных граф.

Вариант 2.

1) Простая цепь.
2) Петля.
3) Контур.
4) Связный ориентированный граф.

Примеры задач

Задача. Дана симметричная матрица размера $n \times n$, в каждой строке которой располагается нечетное число ненулевых элементов. Показать, что n является четным. Диагональные элементы матрицы равны нулю.

Решение. Построим пример матрицы, удовлетворяющей условиям задачи.

$$M = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 4 & 5 \\ 2 & 4 & 0 & 6 \\ 3 & 5 & 6 & 0 \end{pmatrix}.$$

Видно, что матрица M, удовлетворяющая всем условиям задачи, имеет размерность 4×4, т.е. $n = 4$ является четным, и построить матрицу с нечетным n невозможно.

Занятие 8

Тема занятия. Бинарные деревья. Сортировка. Бинарные деревья поиска. Остовные деревья. Алгоритм Краскала. Алгоритм Прима. Матричная формула Кирхгофа.

Внутриклассные задания. 9.4, 9.8.

Задания на дом. 8.6, 8.7, 8.10, 8.11, 8.12.

Самостоятельная работа

Вариант 1.

1) Лес.
2) Корень дерева.
3) Лист.
4) Высота дерева.
5) Высота вершины.

Вариант 2.

1) Дерево.
2) Мост.
3) Куст.
4) Глубина вершины.
5) Уровень вершины.

Примеры задач

Задача. Для приведенного ниже графа определить следующие параметры:

1) высоту дерева;
2) уровень вершины e;
3) уровень вершины g;
4) уровень вершины a;
5) какая вершина является родителем i;
6) какие веришны являются сыновьями b?

Решение. 1) 4; 2) 2; 3) 1; 4) 4; 5) e; 6) c, d, e.

Задача. Сколько остовных деревьев имеет граф K_4?

Решение. По формуле Кирхгофа для матрицы смежности A и матрицы степеней D имеем:

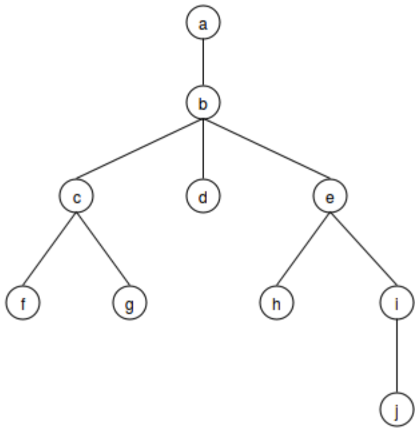


Рисунок 1 — Бинарное дерево.

$$K = D - A = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}.$$

$$K_{11} = \begin{vmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{vmatrix} = 3^3 + (-1)^3 + (-1)^3 - 3 - 3 - 3 = 27 - 11 = 16.$$

Граф K_4 имеет 16 остовных деревьев.

Занятие 9

Тема занятия. Эйлеровы и гамильтоновы графы. Алгоритм Флери. Двудольные графы и паросочетания.

Внутриклассные задания. 10.1, 10.2, 10.4, 10.7, 10.10, 10.12, 10.13, 10.18.

Задания на дом. 10.5, 10.6, 10.8, 10.11, 10.19.

Самостоятельная работа

Вариант 1.

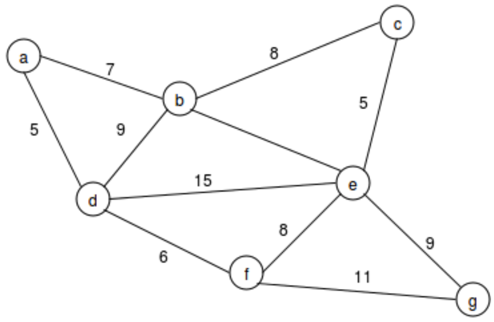


Рисунок 2 — Граф для нахождения минимального остовного дерева.

1) Для графа построить минимальное остовное дерево, используя алгоритм Прима.

Вариант 2.

1) Для графа построить минимальное остовное дерево, используя алгоритм Краскала.

Примеры задач

Задача. Могут ли существовать двудольные графы со следующими степенями вершин:

1) в доле A – 2,3,4,5; в доле B – 3,3,4,4;

2) в доле A – 2,3,3,4; в доле B – 3,3,3,4;

3) в доле A – 3,4,5,5,5,6,6,6; в доле B – 3,5,5,5,5,5,5,5?

Решение. 1) да; 2) нет; 3) нет.

Задача. В классе 12 мальчиков и 16 девочек. Каждая девочка дружит ровно с 3 мальчиками. Количество девочек, с которыми дружат мальчики, одинаково. Со сколькими девочками дружит каждый мальчик?

Решение. Построив двудольный граф, удовлетворяющий всем условиям задачи, убедимся, что каждый мальчик дружит с 4 девочками.

Занятие 10

Тема занятия. Задача о назначениях.

Внутриклассные задания. 11.1, 11.5, 11.6, 12.1, 12.2, 12.4, 12.5, 12.7, 12.8, 12.9.

Задания на дом. 11.7, 12.10, 12.11, 12.12.

Примеры задач

Задача. Четверо вкладчиков решили открыть счета в четырех банках. Их годовая прибыль указана в матрице

$$\begin{pmatrix} 4 & 6 & 9 & 7 \\ 13 & 10 & 14 & 14 \\ 9 & 9 & 16 & 13 \\ 12 & 10 & 12 & 10 \end{pmatrix}.$$

Определить оптимальное распределение суммарной прибыли вкладчиков.

Решение. Согласно алгоритму решения задачи о назначениях:

$$\begin{pmatrix} -4 & -6 & \boxed{-9} & -7 \\ -13 & -10 & \boxed{-14} & -14 \\ -9 & -9 & \boxed{-16} & -13 \\ \boxed{-12} & -10 & -12 & -10 \end{pmatrix} \Rightarrow \begin{pmatrix} 5 & 3 & 0 & 2 \\ \boxed{1} & 4 & 0 & 0 \\ 7 & 7 & 0 & 3 \\ 0 & \boxed{2} & 0 & 2 \end{pmatrix} \Rightarrow$$

$$\Rightarrow \begin{pmatrix} 4 & \boxed{0} & 0 & 2 \\ \boxed{0} & 1 & 0 & 0 \\ 6 & 4 & \boxed{0} & 3 \\ \boxed{0} & 0 & 1 & 3 \end{pmatrix}.$$

Таким образом, максимальная прибыль составит $S = 6 + 14 + 16 + 12 = 48$.

Занятие 11

Тема занятия. Укладки графов. Формула Эйлера. Критерий планарности графа. Алгоритм укладки графа на плоскости.

Внутриклассные задания. 13.1, 13.2, 13.3, 13.4, 13.6.

Задания на дом. 13.5, 13.7, 13.9.

Самостоятельная работа

$$M = \begin{pmatrix} 1 & 7 & 1 & 3 \\ 1 & 6 & 4 & 6 \\ 17 & 1 & 5 & 1 \\ 1 & 6 & 10 & 4 \end{pmatrix}.$$

Вариант 1.

1) Для матрицы M решить задачу о назначениях на минимум.

Вариант 2.

1) Для матрицы M решить задачу о назначениях на максимум.

Примеры задач

Задача. Планарный граф содержит 12 вершин степени 3. Сколько у этого графа ребер и граней?

Решение. По формуле Эйлера:

$$\begin{cases} n - m + f = 2, \\ m \le 3n - 6, \end{cases} \Rightarrow \begin{cases} f = 2 + m - n, \\ m \le 30. \end{cases}$$

Сумма степеней всех вершин графа $12 \cdot 3 = 36$.

$$36 = 2m \Rightarrow m = 18 \Rightarrow f = 2 + 18 - 12 = 8.$$

Таким образом, у графа 18 ребер и 8 граней.

Задача. Проверить граф K_6 на планарность.

Решение. Для графа K_6 $n = 6$, $m = 15$. Тогда по формуле Эйлера

$$15 \le 3 \cdot 6 - 6,$$

$$15 \le 12. \text{ (не верно)}$$

Значит, граф K_6 не планарный.

Занятие 12

Тема занятия. Нахождение кратчайших путей в графе. Алгоритм Дейкстры.

Внутриклассные задания. 14.3, 14.4.

Задания на дом. 14.5, 14.6.

Примеры задач

Задача. Для схемы городов решить задачу единого среднего.

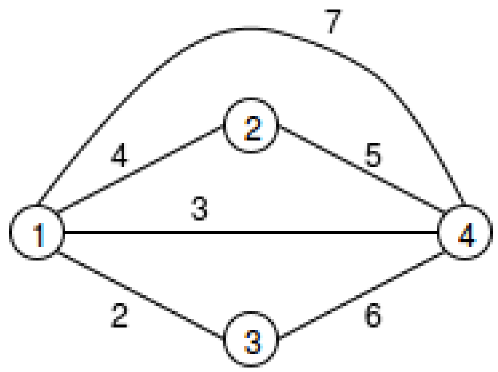


Рисунок 3 — Граф для решения задачи единого среднего.

Масса грузов, которые необходимо привезти, указана в таблице.

Пункт	1	2	3	4
Груз (т)	8	9	7	6

Решение. Составим таблицу.

Пункт	Груз	Склад				Г×C_1	Г×C_2	Г×C_3	Г×C_4
		1	2	3	4				
1	8	0	4	2	3	0	32	16	24
2	9	4	0	6	5	36	0	54	45
3	7	2	6	0	6	14	42	0	42
4	6	3	5	6	0	18	30	36	0
					$\sum$	$\boxed{68}$	104	106	111

Исходя из положения минимума в последней строке, разместим склад в пункте 1.

Занятие 13

Тема занятия. Задачи сетевого планирования. Метод критического пути. Управление проектами с неопределенным временем выполненения работ. Оптимизация сетевого графика.

Внутриклассные задания. 15.1, 15.3, 15.4.

Задания на дом. 15.2, 15.5.

Самостоятельная работа

Для графа найти кратчайшие пути от вершины 1 до всех остальных.

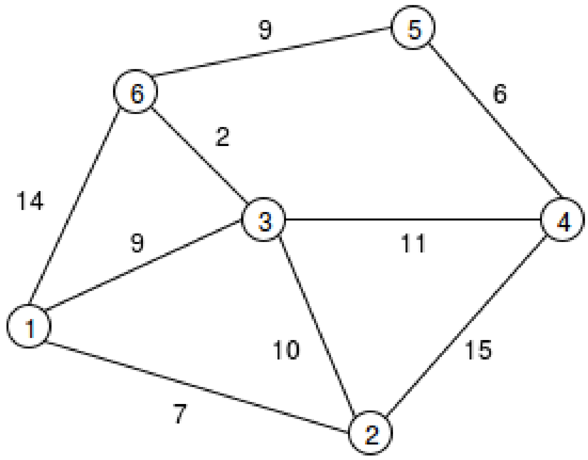


Рисунок 4 — Граф для нахождения кратчайшего пути.

Примеры задач

Задача. Минимизировать общее время выполнения проекта с наименьшими дополнительными затратами. Построить сетевой график.

Решение. Найдем критический путь при условии, что все работы совершаются за минимальное время.

Таким образом, минимальное время, за которое может быть выполнен проект, равно 12 дням. Критический путь проходит по этапам B, D, E, F, H. То есть, работы A, C, G не лежат на критическом пути.

Построим сетевой график, соответствующий условию задачи (рисунок 5).

Рассмотрим эти работы и определим, нельзя ли выполнить их в стандартные сроки без увеличения общего времени выполнения проекта. Выполнение

Работа	Предшественник	Стандартное время, дн.	Минимальное время, дн.	Затраты на работы	
				При стандартном времени, т.р.	При минимальном времени, т.р.
A	—	3	1	900	1700
B	—	6	3	2000	4000
C	A	2	1	500	1000
D	B, C	5	3	1800	2400
E	D	4	3	1500	1850
F	E	3	1	3000	3900
G	B, C	9	4	8000	9800
H	F, G	3	2	1000	2000

i	1	2	3	4	5	6	7
$l(v_i)$	0	1	3	6	9	10	12
$l'(v_i)$	12	10	9	6	3	2	0
$L(v_i)$	0	2	3	6	9	10	12

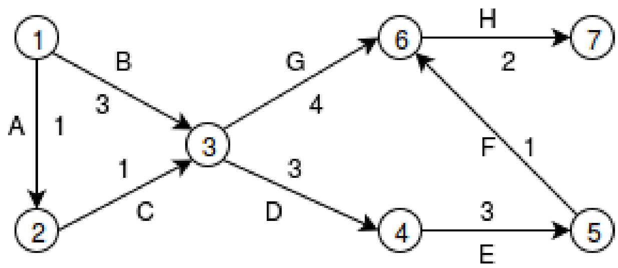


Рисунок 5 — Сетевой график.

этих работ в стандартное время дает следующую экономию: A – 800 т.р., C – 500 т.р., G – 1800 т.р.

Проверим, существует ли реальная возможность сэкономить. Для работы G: $r(3,6) = 3$ дня – невозможно сэкономить; A: $r(1,2) = 1$ день – также невозможно; C: $r(2,3) = 1$ день – экономия возможна, новый критический путь: $A \to C \to D \to E \to F \to H$.

Таким образом, общая стоимость проекта: $1700+500+2400+1850+3900+2000 = 12350$ т.р.

Занятие 14

Тема занятия. Потоки в сетях. Алгоритм Форда и Фалкерсона. Метод блокирующих потоков. Раскраска графов. Клики и независимые множества.

Внутриклассные задания. 16.1, 16.3, 16.5.

Задания на дом. 16.2, 16.4.

Примеры задач

Задача. Найти максимальный поток для сети.

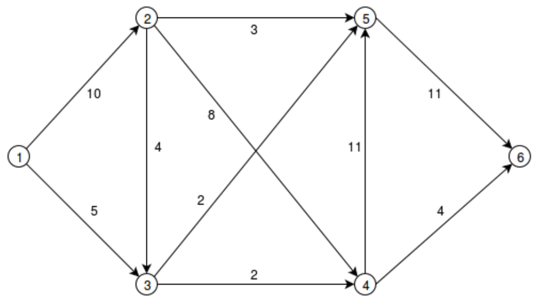


Рисунок 6 — Граф для нахождения максимального потока в сети.

Решение. Согласно алгоритму Форда-Фалкерсона, построим начальный поток в сети. Для примера, это может быть поток $1 \to 2 \to 5 \to 6$. Максимальная пропускная способность сети на этом участке определяется участком $2 \to 5$ и ограничивается 3 единицами траффика. Построив такой поток мы тем самым полностью загрузим участок $2 \to 5$. Отразим это на рисунке путем введения обратного потока указанного объема. Пропускная способность исходных каналов, естественно, сократится.

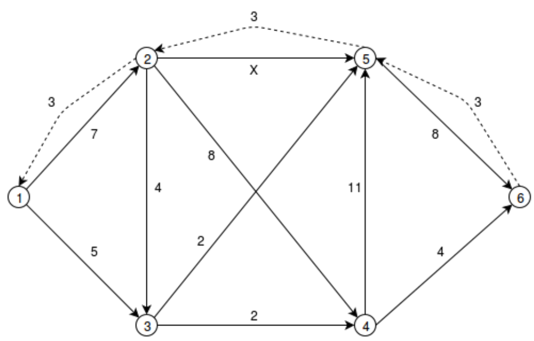


Рисунок 7 — Начальный поток.

Следующий поток $1 \to 3 \to 4 \to 6$ величиной 2 (рисунок 8).

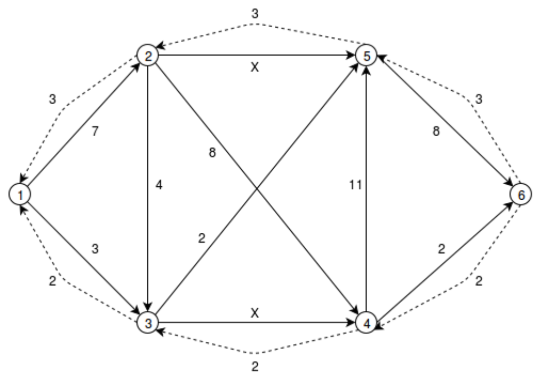


Рисунок 8 — Поток 2.

Следующий поток $1 \to 2 \to 4 \to 5 \to 6$ величиной 7 (рисунок 9).

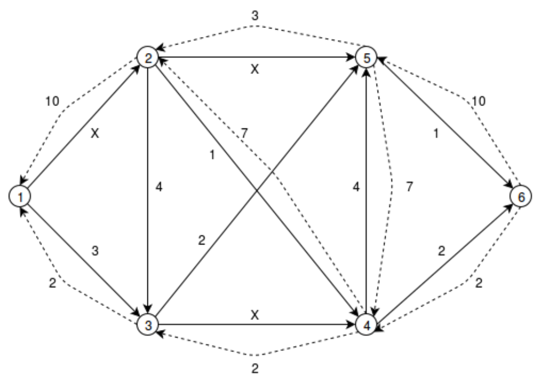


Рисунок 9 — Поток 3.

Следующий поток $1 \to 3 \to 5 \to 6$ величиной 1 (рисунок 10).

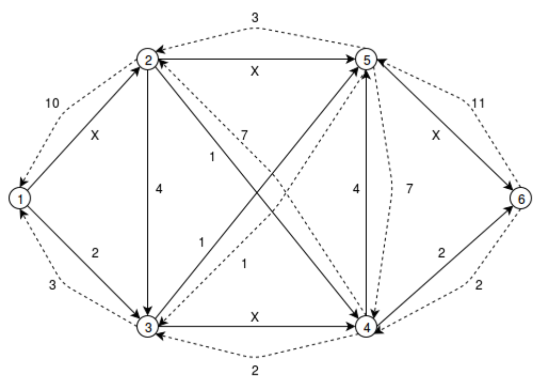


Рисунок 10 — Максимальный поток.

После выполнения этого шага не существует возможности пустить в сети еще поток ненулевого траффика. Таким образом, величина максимального потока известна и равна $3 + 2 + 7 + 1 = 13$.

Заключение

В ходе практики удалось реализовать все поставленные цели и задачи:

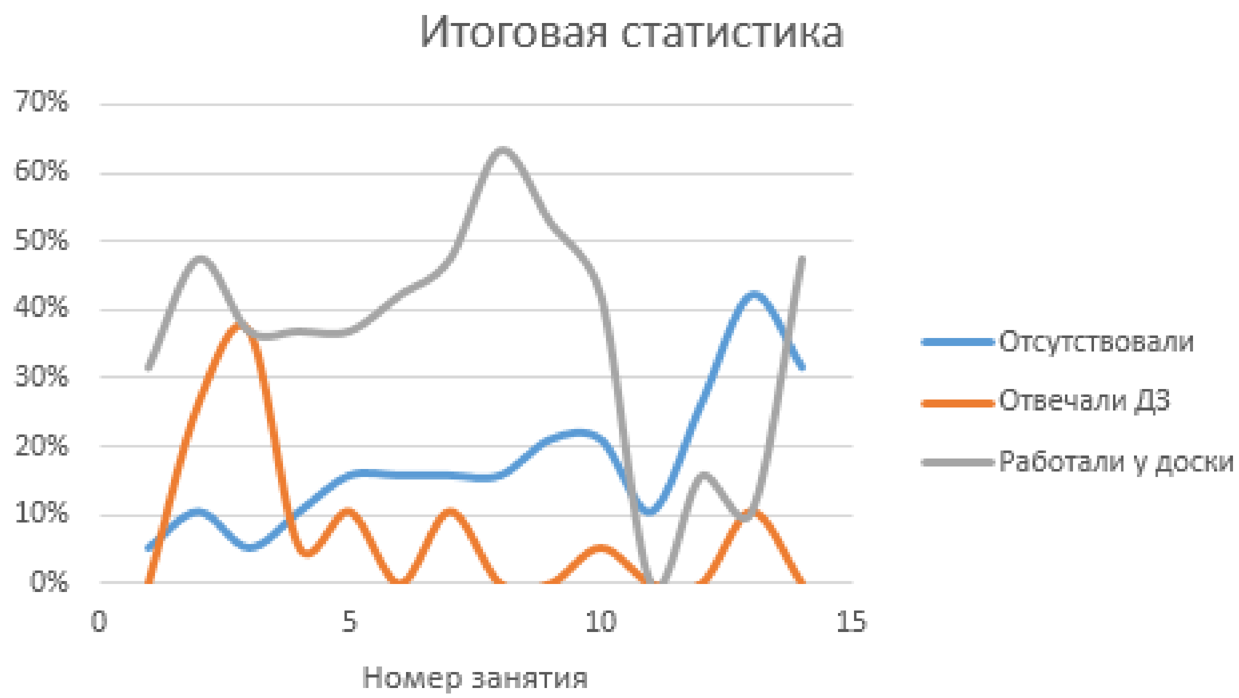
- приобрести бесценный практический опыт и навыки работы с коллективом студентов с учетом его психологической структуры и уровня развития;
- углубить свои знания в дискретной математике и педагогике;
- приобрести опыт индивидуального консультирования по сложным вопросам предмета;
- сформировать умения по организации продуктивного взаимодействия с группой на паре (установление личных контактов, навыки сотрудничества, диалогового общения и т.д.);
- развить умения выявлять, анализировать и учитывать при организации учебно-воспитательного процесса общие психологические закономерности;
- умение помечать и анализировать возникающие в коллективе ситуации, требующие педагогического вмешательства.

Очень важным, если не определяющим фактором, способствующим успешному прохождению педагогической практики, явилось доброжелательное, участвующее отношение преподавателей: оказывалась всякая помощь, давались ценные советы по разработке и проведению занятий. В свою очередь, студенты проявляли дисциплинированность, одобрение и заинтересованность, демонстрировали хорошую посещаемость в семестре и успеваемость на экзамене (некоторая статистика по занятиям приведена в приложении А).

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ А

ИТОГОВАЯ СТАТИСТИКА



БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Amdahl, G. Validity of the single-processor approach to achieving large scale computing capabilities. / G.M. Amdahl. —AFIPS Press, Reston, Va., 1967. —pp. 483-485.
2. Boffley, T. Implementing Parallel simplex algorithms. / T.B. Boffley, R. Hay. —Cambridge University Press, 1989. —pp. 169-176.
3. Cvetanovic, Z. Efficient decomposition and performance of parallel pde, fft, monte-carlo simulations, simplex, and sparse solvers. / Z. Cvetanovic, E.G. Freedman, C. Nofsinger // Journal of Supercomputing. — 1991. — no. 5. —P. 19–38.
4. Dantzig, G. Linear Programming and Extensions / G.B. Dantzig. — Princeton, NJ: Princeton Univ. Press., 1963.
5. Data-parallel implementations of dense simplex mmethod on the connection machine cm-2. / J. Eckstein, I.I. Boduroglu, L. Polymenakos, D. Goldfarb // ORSA Journal on Computing. —1995. —no. 7. —P. 402–416.
6. Finkel, R. Large-grain parallelism – three case studies. / R.A. Finkel ; Ed. by L.H. Jamieson, D. Gannon, R.J. Douglas. —MIT Press, Cambridge, MA, 1987. —pp. 21-63.
7. Forrest, J. Steepest-edge simplex algorithms for linear programming. / J.J. Forrest, D. Goldfarb // Mathematical Programming. — 57:341–374, 1992.
8. Four vector-matrix primitives. / A. Agrawal, G.E. Blelloch, R.L. Krawitz, C.A. Phillips // ACM Symposium on parallel Algorithms and Architectures. —1989. —P. 292–302.
9. Goldfarb, D. A practical steepest-edge simplex algorithm. / D. Goldfarb, J.K. Reid // Mathematical Programming. —12:361–371, 1977.
10. Hall, J. Towards a practical parallelisation of the simplex method / J.A.J. Hall // Computational Management Science. — 7 (2010), pp. 139–170.
11. Harris, P. Pivot selection method of the devex lp code. / P.M.J. Harris // Mathematical Programming. —5:1–28, 1973.

12. Introduction to Parallel Computing: Design and Analysis of Algorithms. / V. Kumar, A. Grama, A. Gupta, G. Karypis. — 2nd edition. — Addison-Wesley, 2003.
13. Luo, J. Linear programming on transputers. / J. Luo, G.L. Reijns // Algorithms, Software, Architecture / Ed. by J. van Leeuwen. —Vol. A-12. —Elsevier, 1992. —P. 525–534.
14. A practical anti-cycling procedure for linear constrained optimization. / P.E. Gill, W. Murray, M.A. Saunders, M.H. Wright // Mathematical Programming. —45:437–474, 1989.
15. Some cocomputation results on mpi parallel implementation of dense simplex method. / E.-S. Badr, M. Moussa, K. Papparrizos [et al.] // Transactions on Engineering, Computiong and Technology. — 2006. — December. —no. 17. —P. 228–231.
16. Stunkel, C. Linear optimization via message-based parallel processing. / C.B. Stunkel // International Conference on Parallel Processing. —Vol. III. —August 1988. —P. 264–271.
17. A survey of Parallel algorithms for linear programming. / J. Luo, G.L. Reijns, F. Bruggeman, G.R. Lindfield ; Ed. by E.F. Deprettere, A.J. van der Veen. —Elsevier, 1991. —Vol. B. —pp. 485-490.
18. Thomadakis, M. An efficient steepest-edge simplex algorithm for simd computers. / M.E. Thomadakis, J.-C. Liu // International Conference on Supercomputing. —1996. —P. 286–293.
19. Yarmish, G. A Distributed Implementation of the Simplex Method. : Ph. D. thesis / G. Yarmish ; Polytechnic University. — Brooklyn, NY : 2001. — March.
20. Zenios, S. Parallel numerical optimization: current status and annotated bibliography. / S.A. Zenios // ORSA Journal on Computing. —1989. —no. 1(1). —P. 20–43.
21. Бабаев, Д. Параллельный алгоритм решения задач линейного программирования. / Д.А. Бабаев, С.С. Марданов // Журнал Вычислительной Математики и Математической Физики. —1991. —№ 31. – С. 86-95.

22. Панюкова, Т. Комбинаторика и теория графов: Учебное пособие. Изд. 3-е, испр. / Т.А. Панюкова. — М.: ЛЕНАНД, 2014. — 216 с.