

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
прикладной математики и
информатики**

А.М. Райгородский

	Рабочая программа дисциплины (модуля)
по дисциплине:	Алгоритмы на графах и динамическое программирование
по направлению:	Информатика и вычислительная техника
профиль подготовки:	Системное программирование и прикладная математика Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
курс:	2
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 3 (осенний) - Экзамен

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 0 час.

лабораторные занятия: 30 час.

Самостоятельная работа: 90 час.

Подготовка к экзамену: 30 час.

Всего часов: 180, всего зач. ед.: 4

Количество контрольных работ, заданий: 3

Программу составил: С.Л. Бабичев, канд. физ.-мат. наук, доцент

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 07.04.2022

Аннотация

Курс представляет собой вторую часть большого курса по разработке и анализу алгоритмов. Она существенно опирается на аппарат, полученный студентами на курсе «Разработка и анализ алгоритмов».

Семестр посвящен изучению подхода динамического программирования и применительно к задачам на графах. Изучаются деревья поиска (splay, AVL, декартово, В-дерево) вместе с подробными доказательствами корректности и асимптотики, а также с описанием прикладных преимуществ каждой структуры.

1. Цели и задачи

Цель дисциплины

Целями дисциплины являются первичное ознакомление студентов с основными принципами проектирования и анализа алгоритмов и структур данных, обучение навыкам обоснования корректности алгоритмов, их практической реализации, теоретической и экспериментальной оценки их временной сложности.

Задачи дисциплины

- научить формулировать задачи в терминах изученных теорий, выбирать подходящий алгоритм для поставленной задачи;
- изучить такие разделы, как хеш-таблицы, динамическое программирование и графы;
- изучить такие разделы, как прикладные задачи теории чисел и комбинаторики, поисковые задачи на строках, сжатие текста, вычислительная геометрия и эвристические алгоритмы.
- научить разрабатывать композиции алгоритмов для решения поставленных задач, оценивать сложности алгоритмов, выбирать подходящие структуры данных для поставленных задач, реализовывать алгоритмы на языках программирования C и C++.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Формулирует совокупность взаимосвязанных задач в рамках поставленной цели работы, обеспечивающих ее достижение. Определяет ожидаемые результаты решения поставленных задач
	УК-2.2 Проектирует решение конкретной задачи проекта, выбирая оптимальный способ ее решения, исходя из действующих правовых норм и имеющихся ресурсов и ограничений
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности

ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
---	---

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- определения асимптотик в O-нотации;
- определения простейших линейных структур данных (стек, очередь, дек, вектор) и сложность обработки запросов в них;
- алгоритм быстрой сортировки;
- алгоритм сортировки слиянием
- определение и практическую необходимость деревьев поиска;

- классификацию и сферы применения хеш-функций;
- организацию и использование хеш-таблиц;
- условия возникновения задачи динамического программирования;
- сведение задачи динамического программирования к функции Беллмана;
- виды задач динамического программирования и методы их решения;
- понятие графов и организацию их хранения;
- алгоритмы обхода графов;
- алгоритмы нахождения специальных элементов в графах;
- алгоритмы нахождения минимальных остовных деревьев;
- алгоритмы нахождения кратчайших путей в графах;
- алгоритмы нахождения потоков в графах;
- алгоритмы нахождения минимальных разрезов и максимальных потоков в графах.
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;

- основы теории чисел в применении к информационным технологиям;
- быстрые алгоритмы работы с длинными числами и матрицами;
- применение алгоритмов теории чисел к задачам криптографии;
- методы комбинаторного поиска;
- Z-функцию и префикс-функцию и их применение;
- структуру данных бор и алгоритм Ахо-Корасик;
- суффиксное дерево и суффиксный автомат;
- алгоритмы сжатия информации: Huffman, LZ77, LZ78, BWT, Arithmetic coding;
- алгоритмы вычислительной геометрии: нахождение выпуклой оболочки, триангуляцию, поиск на плоскости.
- методы приближённого решения NP-сложных задач

уметь:

- оценивать сложность алгоритмов;
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;

владеть:

- разнообразными деревьями поиска и методикой выбора наиболее предпочтительного в каждой конкретной ситуации;
 - методами доказательства корректности утверждений об алгоритмах;
 - методами декомпозиции задач на более простые;
 - методами синтеза решения сложных задач из простых.
-
- техникой выбора необходимой хеш-функции для данной задачи;
 - техникой сведения уравнения Беллмана к программному коду;
 - техникой реализаций изученных алгоритмов на графах;
 - методами доказательства корректности утверждений об алгоритмах;
 - методами декомпозиции задач на более простые;
 - методами синтеза решения сложных задач из простых.
-
- техникой выбора необходимых алгоритмом для решения задач теории чисел;
 - техникой быстрого поиска информации в текстовых данных;
 - умением обратимо преобразовывать информацию в более компактное представление;
 - методами аналитической геометрии для решения задач вычислительной геометрии;
 - умением находить приближённые решения сложных задач эвристическими методами.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Хеш-функции и хеш-таблицы.	2		2	6
2	Задача динамического программирования.	2		2	6
3	Выбор декомпозиции.	2		2	6
4	Многомерные варианты.	2		2	6
5	Графы, их организация.	2		2	6
6	Отношение сильной связности.	2		2	6
7	Мосты и точки сочленения.	2		2	6
8	Поиск кратчайшего расстояния в графах.	2		2	6
9	Алгоритм Флойда-Уоршалла.	2		2	6
10	Поиск минимальных основных деревьев.	2		2	6
11	Паросочетания в произвольном графе.	2		2	6
12	Потоки в графах.	2		2	6
13	Слоистая сеть.	2		2	6
14	Деревья.	2		2	6
15	Задача RMQ.	2		2	6
Итого часов		30		30	90
Подготовка к экзамену		30 час.			
Общая трудоёмкость		180 час., 4 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

1. Хеш-функции и хеш-таблицы.

Хеширование во внешней памяти. Применение хеширования для задач поиска дублирующейся информации. Соокло хеш-таблицы.

2. Задача динамического программирования.

Уравнение Беллмана. Решение задачи динамического программирования в прямом и обратном порядке. Восстановление ответа.

3. Выбор декомпозиции.

Задача Левенштейна. Задача о рюкзаке. Битовые множества и побитовые операции.

4. Многомерные варианты.

Использование дерева отрезков и дерева Фенвика. Динамическое программирование по контуру. Быстрое возведение матрицы в степень.

5. Графы, их организация.

Обход графов. Алгоритмы BFS и DFS. Лемма о белых путях. Топологическая сортировка.

6. Отношение сильной связности.

Компоненты связности. Алгоритмы Косарайю и Тарджана. Конденсация графа.

7. Мосты и точки сочленения.

Классификация рёбер в дереве обхода DFS. Отношение эквивалентности R. Рёберная двусвязность. Мосты и точки сочленения.

8. Поиск кратчайшего расстояния в графах.

Алгоритм Дейкстры. Реализации алгоритма Дейкстры. Алгоритм A*. Допустимые и монотонные эвристики. Корректность и сложность.

9. Алгоритм Флойда-Уоршалла.

Реализация, асимптотика. Нахождение отрицательных циклов. Алгоритм Джонсона. Алгоритм Форда-Беллмана. Матрица транзитивного замыкания.

10. Поиск минимальных основных деревьев.

Алгоритм Прима. Лемма о безопасном ребре. Система непересекающихся множеств. Алгоритм Краскала. Алгоритм Боруки.

11. Паросочетания в произвольном графе.

Двудольные графы. Понятие увеличивающего пути. Теорема Бержа. Алгоритм поиска максимального паросочетания в двудольном графе. 2SAT.

12. Потоки в графах.

Определение сети, потока, остаточной сети. Обратные рёбра. Лемма о связи величины произвольного потока и величины произвольного разреза. Теорема Форда-Фалкерсона. Алгоритм Форда-Фалкерсона. Алгоритм Эдмондса-Карпа. Масштабирование в алгоритме Эдмондса-Карпа.

13. Слоистая сеть.

Блокирующий поток. Алгоритм Диница. Единичные сети. Эффективность алгоритма Диница в единичных сетях.

14. Деревья.

Диаметр дерева. Определение центроида в дереве. Лемма о количестве центроидов. Изоморфизм графов. Задача LCA. Решение с помощью Эйлера обхода. Решение LCA с помощью алгоритма Фарах-Колтона и Бендера.

15. Задача RMQ.

Решение за $O(N \log N)$ предподсчёта. Решение за $O(N)$ предподсчёта. Heavy-light декомпозиция. Тяжёлые и лёгкие рёбра. Центроидная декомпозиция.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория с доской.

6. Перечень рекомендуемой литературы

Основная литература

1. Алгоритмы : построение и анализ [Текст] / Т. Кормен, Ч. Лейзерсон, Р. Ривест .— М. : МЦНМО, 2001 .— 960 с.
2. Программирование: теоремы и задачи [Текст], [учеб. пособие] /А. Шень. -М., МЦНМО, 2017
3. Алгоритмы [Текст] : [учеб. пособие для вузов] / С. Дасгупта, Х. Пападимитриу, У. Вазирани ; пер. с англ. А. А. Куликова ; под ред. А. Шеня .— М. : МЦНМО, 2014 .— 320 с.

Дополнительная литература

Не предусмотрено

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

1. http://neerc.ifmo.ru/wiki/index.php?title=Дискретная_математика,_алгоритмы_и_структуры_данных. «Викиконспекты», сайт Санкт-Петербургского Университета ИМТО.
2. <http://e-maxx.ru/>. Maximal algo: личный сайт Максима Иванова, посвящённый алгоритмам и структурам данных.
3. <https://codeforces.com/>. Международная платформа онлайн-конTESTов по спортивному программированию на базе СГУ и ИМТО.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

Для успешного освоения дисциплины, студент использует следующие программные средства: компилятор языка C.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

В преподавании дисциплины сочетаются практические занятия в компьютерном классе и домашние задания, состоящие из теоретических задач и задач на программирование с автоматической проверкой корректности.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Информатика и вычислительная техника
профиль подготовки:	Системное программирование и прикладная математика Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
курс:	2
квалификация:	бакалавр
Семестр, формы промежуточной аттестации: 3 (осенний) - Экзамен	
Разработчик:	С.Л. Бабичев, канд. физ.-мат. наук, доцент

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Формулирует совокупность взаимосвязанных задач в рамках поставленной цели работы, обеспечивающих ее достижение. Определяет ожидаемые результаты решения поставленных задач
	УК-2.2 Проектирует решение конкретной задачи проекта, выбирая оптимальный способ ее решения, исходя из действующих правовых норм и имеющихся ресурсов и ограничений
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации

2. Показатели оценивания компетенций

В результате изучения дисциплины «Алгоритмы на графах и динамическое программирование» обучающийся должен:

знать:

- определения асимптотик в O-нотации;
- определения простейших линейных структур данных (стек, очередь, дек, вектор) и сложность обработки запросов в них;
- алгоритм быстрой сортировки;
- алгоритм сортировки слиянием
- определение и практическую необходимость деревьев поиска;
- классификацию и сферы применения хеш-функций;
- организацию и использование хеш-таблиц;
- условия возникновения задачи динамического программирования;
- сведение задачи динамического программирования к функции Беллмана;
- виды задач динамического программирования и методы их решения;
- понятие графов и организацию их хранения;
- алгоритмы обхода графов;
- алгоритмы нахождения специальных элементов в графах;
- алгоритмы нахождения минимальных остовных деревьев;
- алгоритмы нахождения кратчайших путей в графах;
- алгоритмы нахождения потоков в графах;
- алгоритмы нахождения минимальных разрезов и максимальных потоков в графах.
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;
- основы теории чисел в применении к информационным технологиям;
- быстрые алгоритмы работы с длинными числами и матрицами;
- применение алгоритмов теории чисел к задачам криптографии;
- методы комбинаторного поиска;
- Z-функцию и префикс-функцию и их применение;
- структуру данных бор и алгоритм Ахо-Корасик;
- суффиксное дерево и суффиксный автомат;
- алгоритмы сжатия информации: Huffman, LZ77, LZ78, BWT, Arithmetic coding;
- алгоритмы вычислительной геометрии: нахождение выпуклой оболочки, триангуляцию, поиск на плоскости.
- методы приближённого решения NP-сложных задач

уметь:

- оценивать сложность алгоритмов;
- строго доказывать утверждения о корректности алгоритмов;
- применять необходимую технику для решения алгоритмических задач;

владеть:

- разнообразными деревьями поиска и методикой выбора наиболее предпочтительного в каждой конкретной ситуации;
- методами доказательства корректности утверждений об алгоритмах;
- методами декомпозиции задач на более простые;
- методами синтеза решения сложных задач из простых.
- техникой выбора необходимой хеш-функции для данной задачи;
- техникой сведения уравнения Беллмана к программному коду;
- техникой реализаций изученных алгоритмов на графах;
- методами доказательства корректности утверждений об алгоритмах;
- методами декомпозиции задач на более простые;
- методами синтеза решения сложных задач из простых.
- техникой выбора необходимых алгоритмов для решения задач теории чисел;
- техникой быстрого поиска информации в текстовых данных;
- умением обратимо преобразовывать информацию в более компактное представление;
- методами аналитической геометрии для решения задач вычислительной геометрии;
- умением находить приближённые решения сложных задач эвристическими методами.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

для 3-го семестра

1. Определите, есть ли в неориентированном графе гамильтонов цикл за время $O(2^n \cdot n)$.
2. Найдите минимальное вершинное покрытие графа.
3. Приведите пример графа, в котором значения, находимые алгоритмом Флойда, экспоненциально растут.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

перечень вопросов для 3-го семестра

- техникой выбора необходимой хеш-функции для данной задачи;
- техникой сведения уравнения Беллмана к программному коду;
- техникой реализаций изученных алгоритмов на графах;
- методами доказательства корректности утверждений об алгоритмах;
- методами декомпозиции задач на более простые;
- методами синтеза решения сложных задач из простых.

1. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

1. Определите, есть ли в неориентированном графе гамильтонов цикл за время $O(2^n \cdot n)$.
2. Найдите минимальное вершинное покрытие графа.
3. Приведите пример графа, в котором значения, находимые алгоритмом Флойда, экспоненциально растут.

2. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Два игрока играют в следующую игру: на доске написан ряд чисел. За один ход игрок стирает любое из чисел, находящихся с краю ряда и записывает его себе. Кто набрал большую сумму - выигрывает. Определите результат игры при наилучшей игре обеих сторон.
2. На заданной клетчатой доске часть клеток недоступна. Определите наибольшее количество непересекающихся домино, которые можно разместить на данной доске.
3. Во взвешенном неориентированном графе найдите такое минимальное число k такое, что в любом минимальном остовном дереве содержится хотя бы одно из этих рёбер.

Пример экзаменационных вопросов.

1. Классификация хеш-функций. Cookoo-hash таблицы.
2. Персистентные хеш-таблицы. Хеш-таблицы во внешней памяти.
3. Задача динамического программирования. Уравнение Беллмана. Решение задачи динамического программирования в прямом и обратном порядке. Восстановление ответа.
4. Декомпозиция задачи в динамическом программировании. Выбор декомпозиции. Критерии выбора. Примеры удачного и неудачного выбора.
5. Задача о расстоянии редактирования. Декомпозиция задачи. Решение. Восстановление ответа.
6. Задача о наполнении целочисленного рюкзака. Декомпозиция задачи. Восстановление ответа. Оценка эвристик.
7. Задача о счастливых билетах. Декомпозиция задачи. Решение.
8. Задача о замощении прямоугольника домино. Декомпозиция задачи. Решение.
9. Задача о красивых ковриках (раскраска в два цвета без квадратов 2×2 одного цвета) за время $O(4^n \cdot (n+m))$ или $O(8^n \cdot \log m)$.
10. Задача о наибольшей возрастающей последовательности. Решение за $O(N \log N)$ с помощью дерева отрезков.

11. Задача о наибольшей возрастающей последовательности. Решение за $O(N \log N)$ с помощью бинарного поиска.
12. Вычисление N -го числа Фибоначчи по заданному модулю за $O(\log N)$
13. Нахождение $A + A^2 + \dots + A^K$ за $O(N^3 \log K)$ для матриц $A(N \times N)$
14. Использование битового представления для кодирования множеств. Реализация операций над множествами.
15. Определение ориентированного и неориентированного графа, пути, вершинно простого пути, рёберно простого пути. Понятие цикла. Определение достижимости между вершинами. Определение связности.
16. Способы организации хранения графа, преимущества и недостатки.
17. Обход графов. Алгоритмы BFS и DFS.
18. Топологическая сортировка. Применение и алгоритм.
19. Обход DFS. Нерекursивный DFS. Лемма о белых путях.
20. Отношение сильной связности. Компоненты связности. Алгоритм Косарайю. Конденсация графа.
21. Отношение сильной связности. Компоненты связности. Алгоритм Тарджана. Конденсация графа.
22. Классификация рёбер в обходе DFS. Отношение эквивалентности R . Нахождение мостов в графе.
23. Классификация рёбер в обходе DFS. Отношение эквивалентности R . Нахождение точек сочленения в графе.

пример экзаменационного билета

1. Два игрока играют в следующую игру: на доске написан ряд чисел. За один ход игрок стирает любое из чисел, находящихся с краю ряда и записывает его себе. Кто набрал большую сумму - выигрывает. Определите результат игры при наилучшей игре обеих сторон.
2. На заданной клетчатой доске часть клеток недоступна. Определите наибольшее количество непересекающихся домино, которые можно разместить на данной доске.
3. Во взвешенном неориентированном графе найдите такое минимальное число k такое, что в любом минимальном остовном дереве содержится хотя бы одно из этих рёбер.

Критерии оценивания

отлично

- 10 Полностью и вовремя решены все задачи без ошибок. Продemonстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы, код оформлен в едином удобочитаемом стиле
- 9 Полностью и вовремя решены все задачи без ошибок. Продemonстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы
- 8 Полностью и вовремя решены все задачи без ошибок. Продemonстрирован грамотный подход к решению задач

хорошо

- 7 Полностью решены все задачи. Допущены несущественные ошибки.
- 6 Полностью решено большинство задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.
- 5 Полностью решено две трети задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

удовлетворительно

- 4 Полностью решено более половины задач. В остальных задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.
- 3 Полностью решено более половины задач.

неудовлетворительно

2 Решено менее половины задач.

1 Не решено ни одной задачи.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

При проведении устного экзамена обучающемуся предоставляется 30 минут на подготовку. Опрос обучающегося по билету на устном экзамене не должен превышать двух астрономических часов.

Во время проведения экзамена обучающиеся могут пользоваться программой дисциплины.