МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И. Носова»

УТВЕРЖДАЮ
Директор ИЭиАС
_____ В.Р. Храмшин

13.02.2024 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

АЛГОРИТМЫ И ТЕОРИЯ СЛОЖНОСТИ

Направление подготовки (специальность)
09.03.01 Информатика и вычислительная техника

Направленность (профиль/специализация) программы
Программное обеспечение средств вычислительной техники и автоматизированных систем

Уровень высшего образования - бакалавриат

Форма обучения
очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Вычислительной техники и программирования
Курс	2, 3
Семестр	4, 5

Магнитогорск
2024 год

Рабочая программа составлена на основе ФГОС ВО - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 929)

Рабочая программа рассмотрена и одобрена на заседании кафедры Вычислительной техники и программирования
25.01.2024, протокол № 5

Зав. кафедрой _____ О.С. Логунова

Рабочая программа одобрена методической комиссией ИЭиАС
13.02.2024 г. протокол № 4

Председатель _____ В.Р. Храмшин

Рабочая программа составлена:
доцент кафедры ВТиП, канд. пед. наук _____ М.М. Гладышева

Рецензент:
директор НИИ «Промбезопасность», д-р техн. наук _____ М.Ю. Наркевич

Лист актуализации рабочей программы

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2025 - 2026 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от __ _____ 20__ г. № __
Зав. кафедрой _____ О.С. Логунова

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2026 - 2027 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от __ _____ 20__ г. № __
Зав. кафедрой _____ О.С. Логунова

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2027 - 2028 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от __ _____ 20__ г. № __
Зав. кафедрой _____ О.С. Логунова

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2028 - 2029 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от __ _____ 20__ г. № __
Зав. кафедрой _____ О.С. Логунова

1 Цели освоения дисциплины (модуля)

Целями освоения дисциплины (модуля) «Алгоритмы и теория сложности» является ознакомление студентов с базовыми понятиями теории алгоритмов, формирование представлений о вычислительной сложности алгоритмов и их использовании для решения прикладных задач.

Для достижения поставленной цели в курсе «Алгоритмы и теория сложности» ре-шаются задачи:

– изучение основных положений теории алгоритмов;

– изучение и исследование представлений понятия «алгоритм» с помощью различ-ных математических моделей (детерминированная машина Тьюринга, вычислимая функ-ция);

– подсчёт вычислительной сложности алгоритмов, классификация задач по степени вычислительной сложности;

– освоение точных, приближённых и эвристических методов решения NP-трудных задач.

2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Алгоритмы и теория сложности входит в обязательную часть учебного плана образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Математическая логика и дискретная математика

Структуры и модели данных

Программирование

Информатика

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Теория автоматов

Функциональное программирование

Логическое программирование

Выполнение и защита выпускной квалификационной работы

Подготовка к сдаче и сдача государственного экзамена

Теория языков программирования

3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Алгоритмы и теория сложности» обучающийся должен обладать следующими компетенциями:

Код индикатора	Индикатор достижения компетенции
ОПК-1 Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности;	
ОПК-1.1	Решает стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования
ОПК-1.2	Решает профессиональные задачи с применением методов теоретического и экспериментального исследования
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения;	
ОПК-8.1	Определяет средства разработки программных средств для решения

	практических задач профессиональной деятельности
ОПК-8.2	Разрабатывает алгоритмы и программы для решения прикладных задач различных классов

4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 7 зачетных единиц 252 акад. часов, в том числе:

– контактная работа – 175,95 акад. часов:

– аудиторная – 170 акад. часов;

– внеаудиторная – 5,95 акад. часов;

– самостоятельная работа – 40,35 акад. часов;

– в форме практической подготовки – 0 акад. час;

– подготовка к экзамену – 35,7 акад. час

Форма аттестации - зачет, курсовая работа, экзамен

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в акад. часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Раздел 1. Математические модели представления алгоритма								
1.1 Детерминированная машина Тьюринга (ДМТ): «чёрный ящик» и структурная схема. Универсальная машина Тьюринга (универсальный интерпретатор). Алгоритмически неразрешимые проблемы. Проблема остановки машины Тьюринга. Тезис Тьюринга	4	4	10		1	1. Самостоятельное изучение учебной и научной литературы. 2. Выполнение лабораторных работ, работа с электронным учебником.	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
1.2 Примитивно-рекурсивные функции. Вычислимые функции. Тезис Чёрча. Эквивалентность детерминированной машины Тьюринга и вычислимой функции		4	10		2	1. Самостоятельное изучение учебной и научной литературы. 2. Выполнение лабораторных работ, работа с электронным учебником.	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		8	20		3			
2. Раздел 2. Теория сложности								

2.1 Определение вычислительной сложности алгоритма как число шагов ДМТ. Недетерминированная машина Тьюринга. Определение вычислительной сложности алгоритма как число шагов НДМТ. Классификация алгоритмов по степени вычислительной сложности. Классы P, NP, E	4	3	8		1	1. Самостоятельное изучение учебной и научной литературы. 2. Выполнение лабораторных работ, работа с электронным учебником.	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
2.2 Определение полиномиальной сводимости. Класс NP-полных задач, эквивалентность NP-полных задач. Доказательство NP-полноты методом сужения. Задачи NP-трудные в сильном смысле. NP-полные задачи, решаемые за псевдополиномиальное время		3	10		1	1. Самостоятельное изучение учебной и научной литературы. 2. Выполнение лабораторных работ, работа с электронным учебником.	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Коллоквиум.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		6	18		2			
3. Раздел 3. Точные методы решения NP-полных задач								
3.1 Общая схема алгоритма с возвратом. Генерация решений в лексикографическом порядке. Все решения задачи «Сумма размеров» с точностью до перестановки слагаемых внутри суммы	4	4	10		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
3.2 Модификация общей схемы для решения задач на максимум. Принцип включения-невключения. Задача о рюкзаке.		4	10		3,2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учеб-ником, выполнение лаборатор-ных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1

3.3 Модификация общей схемы для решения задач на минимум. Задача об упаковке в минимальное число контейнеров (точный алгоритм)		2	10		4	1. Самостоятельное изучение учебной и научной литературы. 2. Выполнение лабораторных работ, работа с электронным учебником.	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный вопрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
3.4 Задачи, решаемые за псевдополиномиальное время. Решение задач «Сумма размеров» и «Рюкзак» с помощью динамического программирования		2	9		3	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный вопрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		12	39		12,2			
4. Раздел 4. Приближённые методы решения NP-полных задач оптимизации								
4.1 Понятие задачи оптимизации. Понятие абсолютной и относительной погрешности приближённого решения задачи оптимизации. Верхние и нижние оценки погрешности приближённых алгоритмов	4	4	4		4	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
4.2 Приближённое решение задачи «Упаковка в контейнеры». FF-алгоритм и FFD-алгоритм		4	4		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учеб-ником, выполнение лаборатор-ных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный вопрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		8	8		6			
Итого за семестр		34	85		23,2		зачёт	
5. Раздел 5. Машинное представление графов								

5.1 Базовые понятия из современной теории графов. Классификация различных типов графов	5	1	2		1	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
5.2 Структуры данных для машинного представления ориентированных и неориентированных нагруженных графов		1	4		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		2	6		3			
6. Раздел 6. Алгоритмы на неориентированных графах								
6.1 Методы систематического обхода графов: поиск в глубину, поиск в ширину, стягивающие деревья	5	2	4		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
6.2 Фундаментальные циклы, циклы Эйлера.		3	4		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа - обсуждение 2. Проверка индивидуальных заданий 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		5	8		4			
7. Раздел 7. Алгоритмы на взвешенных ориентированных графах								

7.1 Общая постановка задачи о поиске кратчайшего пути. Алгоритм Форда–Беллмана	5	2	4		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
7.2 Алгоритм Дейкстры		2	2		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
7.3 Алгоритм Флойда. Оптимальное размещение различных типов обслуживающих центров на сильносвязных ориентированных нагруженных графах с целыми положительными весами дуг и вершин		4	8		2	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ. 3. Выполнение курсовой работы	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		8	14		6			
8. Раздел 8. Классические NP-полные задачи на сетях и графах								
8.1 Алгоритм с возвратом. Генерация всех гамильтоновых циклов, задача коммивояжера	5	1	2		1	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Устный опрос.	

8.2 Правильная раскраска неориентированного графа в минимальное число цветов точным и приближённым алгоритмом		1	4		3,15	1. Самостоятельное изучение учебной и научной литературы. 2. Работа с электронным учебником, выполнение лабораторных работ. 3. Подготовка к экзамену	1. Беседа – обсуждение. 2. Проверка индивидуальных заданий. 3. Проверка курсовой работы.	ОПК-1.1, ОПК-1.2, ОПК-8.1
Итого по разделу		2	6		4,15			
Итого за семестр		17	34		17,15		экзамен,кр	
Итого по дисциплине		51	119		40,35		зачет, курсовая работа, экзамен	

5 Образовательные технологии

1. Традиционные образовательные технологии, ориентированные на организацию образовательного процесса и предполагающую прямую трансляцию знаний от преподавателя к аспиранту.

Формы учебных занятий с использованием традиционных технологий:

Информационная лекция – последовательное изложение материала в дисциплинарной логике, осуществляемое преимущественно вербальными средствами (монолог преподавателя).

Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.

2. Технологии проблемного обучения – организация образовательного процесса, которая предполагает постановку проблемных вопросов, создание учебных проблемных ситуаций для стимулирования активной познавательной деятельности аспирантов.

3. Интерактивные технологии – организация образовательного процесса, которая предполагает активное и нелинейное взаимодействие всех участников, достижение на этой основе личностно значимого для них образовательного результата.

Формы учебных занятий с использованием специализированных интерактивных технологий:

Лекция «обратной связи» – лекция–провокация (изложение материала с заранее запланированными ошибками), лекция-беседа, лекция-дискуссия, лекция-конференция.

4. Информационно-коммуникационные образовательные технологии – организация образовательного процесса, основанная на применении программных сред и технических средств работы со знаниями в различных предметных областях.

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

Представлено в приложении 1.

7 Оценочные средства для проведения промежуточной аттестации

Представлены в приложении 2.

8 Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) Основная литература:

1. Миков, А.Ю. Алгоритмы и теория сложности: учебное пособие [Текст]. / А.Ю. Миков, Н.С. Сибилёва – М.: ФГУП НТЦ «Информрегистр», 2017. – № 0321702297.

2. Миков, А.Ю. Алгоритмы на сетях и графах: учебное пособие [Текст]. / А.Ю. Миков, С.И. Файнштейн – М.: ФГУП НТЦ «Информрегистр», 2016. – № 0321601532.

б) Дополнительная литература:

3. Кормен Т.Х. Алгоритмы: построение и анализ [Текст] / Т.Х. Кормен, Ч.И. Лейзерсон, Р. Л. Ривест и др. – ВИЛЬЯМС, 2013. – 1328 с.

4. Гэри, М. Вычислительные машины и труднорешаемые задачи [Текст]/ М. Гэри, Д. Джонсон. – М. : Мир, 1982. – 416 с.

5. Шумова, Л.В.Частично-рекурсивные функции и алгоритмы: учебное пособие [Текст] / Л.В. Шумова, О.Б. Воронина, А.А. Шпонько. – Магнитогорск: МГТУ, 2004. – 57 с.

в) Методические указания:

1. Миков, А.Ю. Алгоритмы и теория сложности: учебное пособие [Текст]. / А.Ю. Миков, Н.С. Сибилёва – М.: ФГУП НТЦ «Информрегистр», 2017. – № 0321702297.

2. Миков, А.Ю. Алгоритмы на сетях и графах: учебное пособие [Текст]. / А.Ю. Миков, С.И. Файнштейн – М.: ФГУП НТЦ «Информрегистр», 2016. – № 0321601532.

г) Программное обеспечение и Интернет-ресурсы:

Программное обеспечение

Наименование ПО	№ договора	Срок действия лицензии
MS Office 2007 Professional	№ 135 от 17.09.2007	бессрочно
Borland Turbo C++	№112301 от 23.11.2005	бессрочно
Borland Turbo Delphi	№112301 от 23.11.2005	бессрочно
MS Visual Studio 2017 Community Edition	свободно распространяемое ПО	бессрочно

Профессиональные базы данных и информационные справочные системы

Название курса	Ссылка

9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

Лекционная аудитория ауд. 282 – Мультимедийные средства хранения, передачи и представления информации;

Компьютерные классы Центра информационных технологий ФГБОУ ВПО «МГТУ им. Г.И. Носова» – Персональные компьютеры, объединенные в локальные сети с выходом в Internet, оснащенные современными программно-методическими комплексами для решения задач в области информатики и вычислительной техники;

Аудитории для самостоятельной работы: компьютерные классы; читальные залы библиотеки – ауд. 282 и классы УИТ и АСУ;

Помещения для самостоятельной работы обучающихся, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и наличием доступа в электронную информационно-образовательную среду организации – классы УИТ и АСУ;

Помещения для хранения и профилактического обслуживания учебного оборудования – Центр информационных технологий – ауд. 379.

Приложение 1

Учебно-методическое обеспечение самостоятельной работы обучающихся

Перечень теоретических вопросов для подготовки к зачету (4 семестр)

1. Понятие алгоритма. Математические модели алгоритма. Классификация языков программирование по математической модели алгоритма.
2. Детерминированная машина Тьюринга (ДМТ): «чёрный ящик» и структурная схема.
3. Универсальная машина Тьюринга (универсальный интерпретатор). Архитектура фон Неймана.
4. Алгоритмически неразрешимые проблемы. Проблема остановки машины Тьюринга.
5. Тезис Тьюринга.
6. Примитивно-рекурсивные функции.
7. Доказательство примитивной рекурсивности арифметических операций.
8. Частично-рекурсивные функции.
9. Тезис Чёрча.
10. Эквивалентность моделей ДМТ и вычислимой функции.
11. Понятие вычислительной сложности алгоритма как числа шагов детерминированной машины Тьюринга.
12. Недетерминированная машина Тьюринга (НДМТ).
13. Понятие вычислительной сложности алгоритма как числа шагов недетерминированной машины Тьюринга.
14. Классификация алгоритмов и задач по вычислительной сложности.
15. Класс P.
16. Класс NP.
17. Класс E.
18. Определение полиномиальной сводимости.
19. Класс NP-полных задач.
20. Эквивалентность NP-полных задач.
21. Доказательство NP-полноты задачи методом сужения.
22. Точные методы решения NP-полных задач.
23. Общая схема алгоритма с возвратом.
24. Общая схема алгоритма с возвратом. Отсечение повторяющихся решений. Генерация решений в лексикографическом порядке.
25. Модификация общей схемы для решения задач на минимум.
26. Модификация общей схемы для решения задач на максимум. Принцип включения-невключения.
27. Понятие задачи оптимизации. Решение NP-полных задач оптимизации приближёнными алгоритмами.
28. Понятие абсолютной погрешности приближённого решения задачи оптимизации.
29. Понятие относительной погрешности приближённого решения задачи оптимизации.
30. Верхние и нижние оценки погрешности приближённых алгоритмов.
31. Приближённые алгоритмы для задачи «Упаковка в контейнеры». FF-алгоритм. FFD-алгоритм.

Перечень индивидуальных заданий по ДМТ

1. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для детерминированной машины Тьюринга, которая заменяет исходный 0 на 1, а исходную 1 на 0.

2. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для детерминированной машины Тьюринга, которая заменяет каждый второй исходный 0 на 1, а каждую исходную 1 на 0.

3. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для детерминированной машины Тьюринга, которая заменяет каждый исходный 0 на 1, а каждую вторую исходную 1 на 0.
4. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для детерминированной машины Тьюринга, которая заменяет каждый второй исходный 0 на 1, а каждую вторую исходную 1 на 0.
5. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет каждый исходный третий 0 на 1.
6. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет каждую исходную третью 1 на 0.
7. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет второй исходный 0 на 1 и каждую исходную 1 на 0.
8. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет третий исходный 0 на 1 и каждую исходную 1 на 0.
9. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет каждый исходный 0 на 1 и вторую исходную 1 на 0.
10. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет каждый исходный 0 на 1 и третью исходную 1 на 0.
11. На ленте записано слово из 0 и 1. Составить систему команд и диаграмму переходов для машины Тьюринга, которая заменяет исходный второй 0 на 1 и исходную вторую 1 на 0.
12. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 3) - 1$.
13. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 3) + 1$.
14. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 4) - 1$.
15. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 4) - 2$.
16. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 4) + 1$.
17. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 4) + 2$.
18. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 5) - 1$.
19. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 5) - 2$.
20. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 5) - 3$.
21. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 5) + 1$.
22. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \mod 5) + 2$.

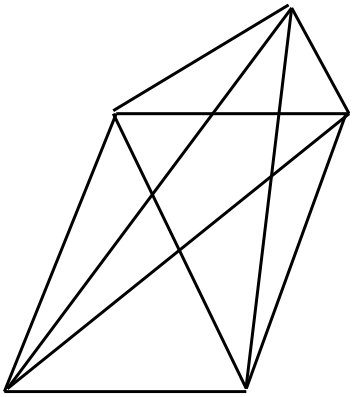
23. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \ \ mod \ \ 5) + 3$.
24. На ленте машины Тьюринга записано число x в унарном коде. Составить систему команд и диаграмму переходов для машины Тьюринга, которая вычисляет функцию $f(x) = (x \ \ div \ \ 2) - 1$.

Перечень вопросов для подготовки к экзамену (5 семестр)

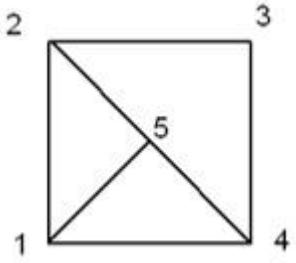
1. Машинное представление графов. Списки инцидентности.
2. Различные стратегии систематического обхода графов.
3. Поиск в глубину на неориентированном графе.
4. Поиск в ширину на неориентированном графе.
5. Построение стягивающих деревьев неориентированных графов.
6. Фундаментальное множество циклов неориентированного графа.
7. Блоки, точки сочленения неориентированного графа.
8. Эйлеров путь, эйлеров цикл в неориентированном графе.
9. Переносимость алгоритмов поиска в глубину и ширину на ориентированные графы.
10. Классификация задач по степени сложности.
11. Алгоритм с возвратом. Генерация всех гамильтоновых циклов полного неориентированного графа.
12. Алгоритм с возвратом. Задача коммивояжера.
13. Математическая постановка задачи о поиске кратчайших путей на взвешенном ориентированном графе.
14. Алгоритм Форда-Беллмана на взвешенном орграфе.
15. Алгоритм Дейкстры на взвешенном орграфе.
16. Восстановление кратчайшего пути от источника до всех остальных вершин по вектору расстояний.
17. Алгоритм Флойда на взвешенном орграфе.
18. Восстановление кратчайшего пути между двумя вершинами по матрице расстояний.
19. Переносимость алгоритмов поиска кратчайших путей на неориентированные графы.
20. Центры и медианы. Оптимальное размещение центра на взвешенном сильносвязном орграфе с неотрицательными весами дуг и вершин.
21. Центры и медианы. Оптимальное размещение медианы на сильносвязном орграфе с неотрицательными весами дуг и вершин.

Перечень заданий для подготовки к экзамену (5 семестр)

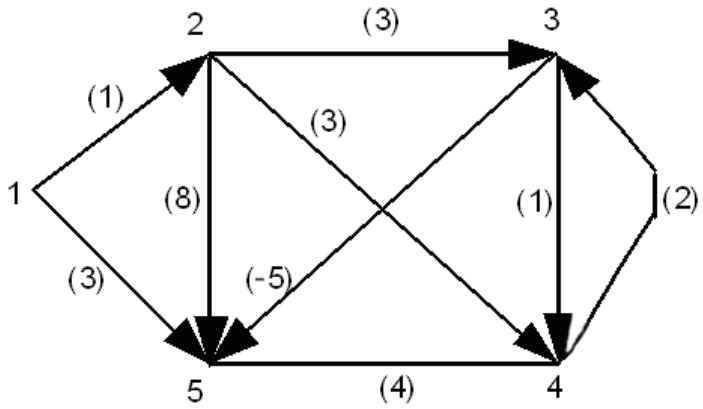
1. Изобразить списки инцидентности графа, заданного списком рёбер.
2. Изобразить списки инцидентности графа, заданного матрицей смежности.
3. Изобразить списки инцидентности графа, заданного матрицей инциденций.
4. Изобразить списки инцидентности модельного графа, заданного рисунком.
5. Протрассировать поиск в глубину на модельном графе.
6. Протрассировать поиск в ширину на модельном графе.
7. Разделить неориентированный граф на компоненты связности с помощью поиска в глубину.
8. Разделить неориентированный граф на компоненты связности с помощью поиска в ширину.
9. На модельном ненагруженном неориентированном графе найти кратчайший путь между парой фиксированных вершин.
10. Найти стягивающее дерево модельного неориентированного графа поиском в глубину.
11. Найти стягивающее дерево модельного неориентированного графа поиском в ширину.
12. Найти множество фундаментальных циклов модельного неориентированного графа.
13. Найти компоненты двусвязности и точки сочленения модельного неориентированного графа.
14. Проверить, существует ли эйлеров цикл в графе:



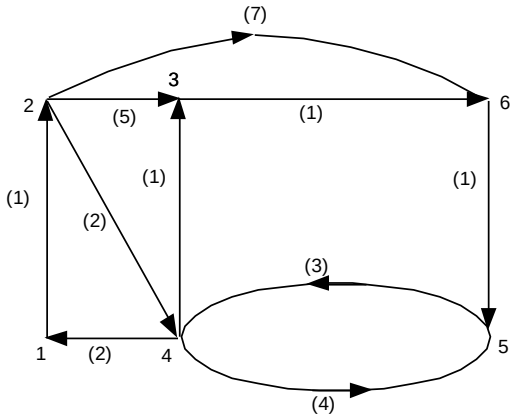
15. Найти все гамильтоновы циклы в графе:



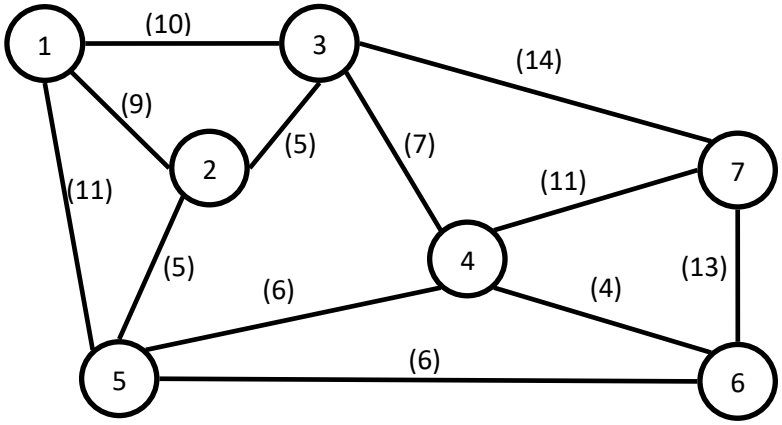
16. Протрассировать алгоритм Форда-Беллмана на модельном графе:



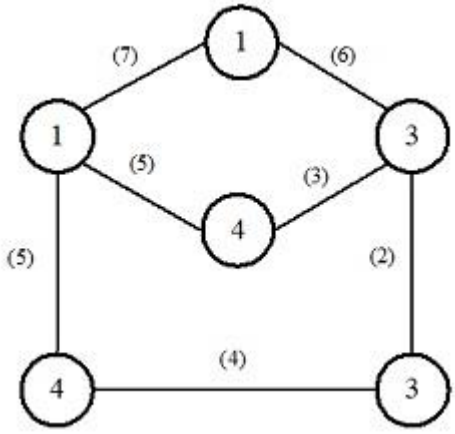
17. Протрассировать алгоритм Дейкстры на модельном графе:



18. Разместить медиану на модельном графе, считая веса вершин равными 1:



19. Разместить центр на модельном графе, считая веса вершин равными 1:



Тесты (5 семестр)

I. Графы

1. Списки инцидентности для машинного представления графа это

а) список рёбер графа в виде пар (начало ребра; конец ребра);

б) двумерный массив $n \times n$, где n – количество вершин графа, $A[i, j] = 1$, если ребро $i–j$ существует; 0, если не существует;

в) двумерный массив $n \times n$, где n – количество вершин графа, $A[i, j]$ = весу ребра или $+\infty$, если ребро не существует;

*г) списки инцидентных вершин.

2. Списки инцидентности в программе

*а) создаются динамически;

б) задаются константами;

в) хранятся в статических переменных;

г) хранятся в стеке.

3. Поиск в глубину в графе

а) просматривают каждое ребро ровно один раз;

*б) просматривают каждую вершину ровно один раз;

в) просматривают только заданное множество вершин;

г) просматривают только заданное множество рёбер.

4. Поиск в ширину в графе

а) просматривают каждое ребро ровно один раз;

*б) просматривают каждую вершину ровно один раз;
в) просматривают только заданное множество вершин;
г) просматривают только заданное множество рёбер.

5. **Поиск в глубину в графе**
а) хранит просмотренные, но ещё не использованные вершины в очереди;
б) хранит просмотренные, но ещё не использованные вершины в деке;
*в) хранит просмотренные, но ещё не использованные вершины в стеке;
г) не запоминает информацию о просмотре вершин.

6. **Поиск в ширину в графе**
*а) хранит просмотренные, но ещё не использованные вершины в очереди;
б) хранит просмотренные, но ещё не использованные вершины в дэке;
в) хранит просмотренные, но ещё не использованные вершины в стеке;
г) не запоминает информацию о просмотре вершин.

7. **Поиск в глубину в графе имеет вычислительную сложность**
а) O(n*m), где n- число вершин, m – количество рёбер;
б) O(2^n), где n- число вершин;
в) растущую как экспонента от количества вершин графа;
*г) O(n+m), где n- число вершин, m – количество рёбер.

8. **Поиск в ширину в графе имеет вычислительную сложность**
а) O(n*m), где n- число вершин, m – количество рёбер;
б) O(2^n), где n- число вершин;
в) растущую как экспонента от количества вершин графа;
*г) O(n+m), где n- число вершин, m – количество рёбер.

9. **Поиском в глубину в графе**
а) нельзя найти путь между двумя заданными вершинами;
*б) можно найти произвольный путь между двумя заданными вершинами;
в) можно найти кратчайший путь между двумя заданными вершинами;
г) можно найти все пути между двумя заданными вершинами.

10. **Поиском в ширину в графе**
а) нельзя найти путь между двумя заданными вершинами;
б) можно найти произвольный путь между двумя заданными вершинами;
*в) можно найти кратчайший путь между двумя заданными вершинами;
г) можно найти все пути между двумя заданными вершинами.

11. **Разделить неор. граф на компоненты связности можно**
а) только поиском в глубину;
б) только поиском в ширину;
*в) и поиском в глубину, и поиском в ширину;
г) неориентированный граф не имеет компонент связности.

12. **Разделить орграф на компоненты связности можно**
а) только поиском в глубину;
б) только поиском в ширину;
в) и поиском в глубину, и поиском в ширину;
*г) ориентированный граф не имеет компонент связности.

13. Поиск в глубину и в ширину можно проводить

а) только для орграфов;
б) только для неориентированных графов;
в) только для взвешенных графов;
*г) для любых графов.

14. **Фундаментальное множество циклов определяется**
*а) только для неориентированных графов;
б) только для ориентированных графов;
в) только для ациклических графов;
г) для любых графов.

15. **Фундаментальное множество циклов**
а) максимальное множество всех пересекающихся циклов графа;
*б) базис линейного пространства всех циклов графа;
в) максимальное множество всех непересекающихся циклов графа;
г) множество всех циклов, проходящих через одну вершину.

16. **Фундаментальное множество циклов можно построить**
*а) добавлением всех хорд к каркасу графа;
б) сгенерировав все циклы графа и выбрав непересекающиеся;
в) сгенерировав все циклы графа и выбрав пересекающиеся;
г) удалив из графа все висячие вершины.

17. **Вычислительная сложность алгоритма построения и печати всех Ф-циклов графа равна**
*а) O(n*m), где n- число вершин, m – количество рёбер;
б) O(2^m), где m- число вершин;
в) растёт как экспонента от количества вершин графа;
г) O(n+m), где n- число вершин, m – количество рёбер.

18. Точка сочленения
а) имеет не менее двух сыновей при поиске в глубину;
б) является корнем при поиске в глубину;
*в) содержится в любом пути между двумя вершинами для некоторой пары вершин;
г) является точкой пересечения всех фундаментальных циклов.

19. **Компоненты двусвязности (блоки) графа**
а) состоят из висячих вершин и инцидентных им рёбер;
*б) не содержат точек сочленения;
в) состоят из вершин степени 2;
г) состоят из точек сочленения.

20. **Вычислительная сложность алгоритма построения и печати всех блоков графа равна**
а) O(n*m), где n- число вершин, m – количество рёбер;
б) O(2^m), где m- число вершин;
в) растёт как экспонента от количества вершин графа;
*г) O(n+m), где n- число вершин, m – количество рёбер.

II. Классификация задач по степени сложности
21. **Вычислительная сложность алгоритма**
а) равна числу машинных команд;
б) равна числу команд транслятора;

в) измеряется встроенным таймером;
*г) показывает рост числа шагов, необходимых для решения задачи, при неограниченном увеличении размерности входных данных.

22. Сложность задачи
*а) равна вычислительной сложности наилучшего алгоритма, известного для её решения;
б) зависит от времени, потраченного на написание программы;
в) зависит от времени, потраченного на разработку алгоритма;
г) измеряется в человеко-часах.

23. Класс P полиномиальных алгоритмов
а) алгоритмы для решения систем линейных уравнений;
б) алгоритмы для решения задач линейного программирования;
*в) задачи, вычислительная сложность которых выражается многочленом фиксированной степени от размерности входных данных;
г) задачи, вычислительная сложность которых выражается экспонентой от размерности входных данных.

24. Класс E экспоненциальных алгоритмов
а) алгоритмы для решения комбинаторных задач;
б) алгоритмы для решения задач статистики и теории вероятности;
*в) задачи, имеющие экспоненциальное число ответов;
г) задачи, вычислительная сложность которых выражается экспонентой от размерности входных данных.

25. NP-полные задачи
а) решаются точными алгоритмами за полиномиальное время;
*б) известными точными алгоритмами решаются за экспоненциальное время;
в) не решаются точными алгоритмами;
г) задачи, имеющие экспоненциальное число ответов.

26. Алгоритм с возвратом
а) применяется для решения задач из класса P;
б) применяется для решения задач из класса E;
в) применяется для решения NP-полных задач;
*г) применяется для решения NP-полных задач и задач из класса E.

27. Задача о пути коммивояжера через сеть городов
а) принадлежит классу E;
б) принадлежит классу P;
*в) принадлежит к классу NP-полных задач;
г) решается за линейное время от суммы вершин и рёбер графа.

III. Кратчайшие пути
30. Алгоритм Форда-Беллмана применим для
а) произвольных неориентированных графов;
б) ациклических графов;
в) ориентированных графов с положительными весами;
*г) ориентированных графов без контуров отрицательного веса.

31. Алгоритм Дейкстры применим для
а) произвольных неориентированных графов;
б) ациклических графов;

*в) ориентированных графов с положительными весами;
г) ориентированных графов без контуров отрицательного веса.

32. Алгоритм Флойда применим для
а) произвольных неориентированных графов;
б) ациклических графов;
в) ориентированных графов с положительными весами;
*г) ориентированных графов без контуров отрицательного веса.

33. Результатом работы алгоритма Форда-Беллмана является
а) кратчайший путь между двумя выделенными вершинами;
б) кратчайшие пути для любой пары вершин;
*в) вектор расстояний от источника до остальных вершин графа;
г) матрица расстояний для всех пар вершин.

34. Результатом работы алгоритма Дейкстры является
а) кратчайший путь между двумя выделенными вершинами;
б) кратчайшие пути для любой пары вершин;
*в) вектор расстояний от источника до остальных вершин графа;
г) матрица расстояний для всех пар вершин.

35. Результатом работы алгоритма Флойда является
а) кратчайший путь между двумя выделенными вершинами;
б) кратчайшие пути для любой пары вершин;
в) вектор расстояний от источника до остальных вершин графа;
*г) матрица расстояний для всех пар вершин.

36. **Вычислительная сложность алгоритма Форда – Беллмана равна**
*а) $O(n*m)$, где n- число вершин, m – количество рёбер;
б) $O(2^n)$, где n- число вершин;
в) $O(n^2)$, где n- число вершин;
г) $O(n+m)$, где n- число вершин, m – количество рёбер.

37. **Вычислительная сложность алгоритма Дейкстры равна**
а) $O(n*m)$, где n- число вершин, m – количество рёбер;
*б) $O(n^2)$, где n- число вершин;
в) $O(n^3)$, где n- число вершин;
г) $O(n+m)$, где n- число вершин, m – количество рёбер.

38. **Вычислительная сложность алгоритма Флойда равна**
а) $O(n*m)$, где n- число вершин, m – количество рёбер;
б) $O(n^2)$, где n- число вершин;
*в) $O(n^3)$, где n- число вершин;
г) $O(n+m)$, где n- число вершин, m – количество рёбер.

39. Алгоритмом Форда-Беллмана можно найти расстояние между парой вершин неориентированного графа
а) в произвольном неориентированном графе;
*б) в неориентированном графе с неотрицательными весами;
*в) в ациклическом неориентированном графе с произвольными весами;
г) алгоритм Форда-Беллмана не применим для неориентированных графов.

40. Алгоритмом Дейкстры можно найти расстояние между парой вершин неориентированного графа
а) в произвольном неориентированном графе;
*б) в неориентированном графе с неотрицательными весами;
в) в ациклическом неориентированном графе с произвольными весами;
г) алгоритм Дейкстры не применим для неориентированных графов.

41. Алгоритмом Флойда можно найти расстояние между любой парой вершин неориентированного графа
а) в произвольном неориентированном графе;
*б) в неориентированном графе с неотрицательными весами;
*в) в ациклическом неориентированном графе;
г) алгоритм Флойда не применим для неориентированных графов.

Перечень заданий для курсовых работ по теме «Оптимальное размещение обслуживающих центров» (5 семестр)

Задание к курсовой работе

1. Оптимально разместить заданный тип обслуживающего центра на графе заданного типа.

2. Придумать реальную задачу, соответствующую математической постановке.

1. Размещение внешнего центра на взвешенном ориентированном графе.
2. Размещение внутреннего центра на взвешенном ориентированном графе.
3. Размещение внутреннего центра на взвешенном смешанном графе.
4. Размещение внешнего центра на взвешенном смешанном графе.
5. Размещение внешне-внутреннего центра на взвешенном ориентированном графе.
6. Размещение внешне-внутреннего центра на взвешенном смешанном графе.
7. Размещение внешней медианы на взвешенном ориентированном графе.
8. Размещение внутренней медианы на взвешенном ориентированном графе.
9. Размещение внешней медианы на взвешенном смешанном графе.
10. Размещение внутренней медианы на взвешенном смешанном графе.
11. Размещение внешне-внутренней медианы на взвешенном ориентированном графе.
12. Размещение внешне-внутренней медианы на взвешенном смешанном графе.
13. Размещение главной внешней медианы на взвешенном ориентированном графе.
14. Размещение главной внутренней медианы на взвешенном ориентированном графе.
15. Размещение главной внутренней медианы на взвешенном смешанном графе.
16. Размещение абсолютного внешнего центра на взвешенном ориентированном графе.
17. Размещение абсолютной внешней медианы на взвешенном ориентированном графе.
18. Размещение абсолютной внешне-внутренней медианы на взвешенном ориентированном графе.
19. Размещение абсолютной внутренней медианы на взвешенном ориентированном графе.
20. Размещение абсолютного внутреннего центра на взвешенном ориентированном графе.
21. Размещение абсолютного внешне-внутреннего центра на взвешенном ориентированном графе.

СПИСОК МОДЕЛЬНЫХ NP–ПОЛНЫХ ЗАДАЧ

Многие задачи из этого списка допускает две возможных постановки:
– задача оптимизации;
– задача распознавания.
В качестве примера рассмотрим задачу о коммивояжере.
Оптимизационная постановка выглядит так:

УСЛОВИЕ. Заданы конечное множество C городов, целые положительные расстояния $d(c_1,c_2)$ для каждой пары городов c_1, c_2.
ВОПРОС ОПТИМИЗАЦИИ. Найти маршрут минимальной длины, проходящий через все города и возвращающийся в исходный пункт.
Чтобы получить теперь задачу распознавания, введем дополнительный параметр – числовую границу B.
ВОПРОС РАСПОЗНАВАНИЯ. Существует ли маршрут, проходящий через все города, длина которого не превосходит B, где B – целое положительное число? (Если решается задача на максимум, то условие «не превосходит B» заменяется условием «не меньше B».)
Традиционно к NP-полным задачам относятся задачи распознавания, однако соответствующие им задачи оптимизации эквивалентны им по сложности, поэтому при формулировке задач можно использовать любую постановку.

1. РАСКРАШИВАЕМОСТЬ ГРАФА (ХРОМАТИЧЕСКОЕ ЧИСЛО)
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$, где $|V|$ – число вершин графа.
ВОПРОС. Верно ли, что граф K–раскрашиваем? (Граф называется K–раскрашиваемым, если его вершины можно раскрасить в K цветов так, чтобы любые две соседние вершины были окрашены в различные цвета).

2. КЛИКА
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Верно ли, что граф G содержит клику размера не менее K?
(Существует ли подграф графа G такой, что число вершин в нем не меньше K и любые две вершины соединены ребром?)

3. ВЕРШИННОЕ ПОКРЫТИЕ
УСЛОВИЕ. Заданы граф $G = <V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Имеется ли в графе вершинное покрытие не более чем из K элементов? (Имеется ли такое подмножество вершин $V' \subseteq V$ мощности не более K, что для любого ребра $(u–v) \in V$ хотя бы одна из вершин u, v принадлежит V'?)
4. РАЗБИЕНИЕ
УСЛОВИЕ. Заданы конечное множество A и положительный целый вес $w(a)$ для каждого $a \in A$.
ВОПРОС. Существует ли подмножество $A' \subseteq A$ такое, что суммарный вес элементов, принадлежащих A', равен суммарному весу элементов, не принадлежащих A'? (Иными словами, можно ли A разбить на два подмножества, равные по весу?)
5. РЮКЗАК
УСЛОВИЕ. Задано конечное множество A, положительные целые веса $w(a)$, стоимости $s(a)$ для каждого $a \in A$ и общее ограничение на вес K.
ВОПРОС. Найти из всевозможных выборок $A' \subseteq A$ такую, чтобы суммарный вес входящих в него элементов не превосходил K, а суммарная стоимость была максимальна.

6. РЮКЗАК С КРАТНЫМ ВЫБОРОМ ЭЛЕМЕНТОВ
УСЛОВИЕ. Задано конечное множество A, положительные целые веса $w(a)$, стоимости $s(a)$ для каждого $a \in A$, общее ограничение на вес K и минимальная стоимость B.
ВОПРОС. Можно ли так сопоставить каждому элементу $a \in A$ целое число $c(a)$ (кратность), чтобы суммарный вес всех предметов из A с учетом кратностей не превосходил K, а суммарная стоимость тех же предметов была не меньше B?

7. УПАКОВКА В КОНТЕЙНЕРЫ
УСЛОВИЕ. Заданы конечное множество A и размеры $s(a) \in [0, 1]$ каждого предмета.

ВОПРОС. Найти такое разбиение множества A на непересекающиеся $A_1, A_2, \ldots, A_k$, чтобы сумма размеров предметов в каждом A_i не превосходила 1 и k было минимальным.

8. ДИНАМИЧЕСКОЕ РАСПРЕДЕЛЕНИЕ ПАМЯТИ

УСЛОВИЕ. Заданы множество элементов данных A, целые положительные $s(a)$ – размер каждого элемента, $r(a)$ – время его поступления, $d(a)$ – время его жизни, положительное целое число D – размер области памяти.

ВОПРОС. Существует ли для множества данных A допустимое распределение памяти? Иными словами, существует ли такая функция $\xi: A \to \{1, 2, \ldots, D\}$, что для каждого элемента $a \in A$ интервал $(\xi(a), \xi(a) + s(a) - 1)$ (т.е. место, занимаемое им в динамической памяти) содержался бы в $[1, D]$ и в любой фиксированный момент времени t_0 интервалы для данных с $r(a) \le t_0 \le r(a)+d(a)$ не пересекались?

9. МНОЖЕСТВО ПРЕДСТАВИТЕЛЕЙ

УСЛОВИЕ. Задано семейство C подмножеств множества S, положительное целое число K.

ВОПРОС. Содержит ли S множество представителей для C, т.е. существует ли в S подмножество S' мощности не более K такое, что S' содержит, по крайней мере, один элемент из каждого множества семейства C.

10. УПОРЯДОЧЕНИЕ ВНУТРИ ИНТЕРВАЛОВ

УСЛОВИЕ. Задано конечное множество заданий T и для каждого задания $t \in T$ целое число $r(t) \ge 0$ – время готовности, целые положительные $d(t)$ и $l(t)$ – директивный срок и длительность.

ВОПРОС. Существует ли для T допустимое расписание, т.е. функция $\xi: T \to N$ (N – множество натуральных чисел), сопоставляющая заданию t момент начала его выполнения $\xi(t)$? (Иными словами, задание t выполняется с момента времени $\xi(t)$ до $\xi(t)+l(t)$, оно не может быть начато ранее момента $r(t)$, должно быть закончено не позднее $d(t)$, и временной интервал выполнения одного задания не может перекрываться с интервалом другого).

11. МИНИМИЗАЦИЯ ШТРАФА ЗА НЕВЫПОЛНЕНИЕ ЗАДАНИЙ

УСЛОВИЕ. Задано конечное множество заданий T и для каждого задания $t \in T$ целые положительные $d(t)$ – директивный срок, $l(t)$ – длительность и $w(t)$ – вес, а также целое положительное число K.

ВОПРОС. Существует ли для T однопроцессорное расписание такое, что сумма $\sum_t (\sigma(t) + l(t) - d(t))w(t)$, взятая по тем $t \in T$, для которых $\sigma(t) + l(t) > d(t)$, не превосходит K? (Однопроцессорное расписание – это функция $\xi: T \to N$ (N – множество натуральных чисел), сопоставляющая заданию t момент начала его выполнения $\xi(t)$ такое, что задание t выполняется с момента времени $\xi(t)$ до $\xi(t)+l(t)$, оно должно быть закончено не позднее $d(t)$, и временной интервал выполнения одного задания не может перекрываться с интервалом другого).

12. МИНИМАКСНОЕ МНОЖЕСТВО ЦЕНТРОВ

УСЛОВИЕ. Заданы граф $G=<V, E>$, положительный целый вес $w(v)$ каждой вершины $v \in V$, положительная целая длина $l(e)$ каждого ребра $e \in E$; положительное целое число $K \le |V|$ и положительное рациональное число B.

ВОПРОС. Существует ли такое множество P мощности K «точек на G» (точка на G либо вершина графа, либо точка на ребре, где ребро e рассматривается как прямолинейный отрезок длины $l(e)$), что если $d(v)$ – длина кратчайшего пути от вершины v до ближайшей к ней точке из P, то $\max\{d(v) \times w(v) : v \in V\} \le B$?

Комментарий. Вариант этой задачи, в котором P – подмножество вершин, также NP-полон, но если K фиксировано или G – дерево, то разрешим за полиномиальное время.

13. МНОЖЕСТВО ЦЕНТРОВ С МИНИМАКСНОЙ СУММОЙ

УСЛОВИЕ. Заданы граф $G=<V, E>$, положительный целый вес $w(v)$ каждой вершины $v \in V$, положительная целая длина $l(e)$ каждого ребра $e \in E$; положительное целое число $K \leq |V|$ и положительное рациональное число B.

ВОПРОС. Существует ли такое множество P мощности K «точек на G» (точка на G либо вершина графа, либо точка на ребре, где ребро e рассматривается как прямолинейный отрезок длины $l(e)$), что если $d(v)$ – длина кратчайшего пути от вершины v до ближайшей к ней точке из P, то $\sum_{v \in V} d(v) \times w(v) \leq B$?

Комментарий. Вариант этой задачи, в котором P – подмножество вершин, также NP-полон, но если K фиксировано или G – дерево, то разрешим за полиномиальное время.

14. СКОПЛЕНИЯ

УСЛОВИЕ. Заданы конечное множество X, целые положительныерасстояния $d(x, y)$ для каждой пары x, y элементов из X, целые положительные K и B.

ВОПРОС. Существует ли такое разбиение множества X на непересекающиеся $X_1, X_2, \ldots, X_k$, что для всех $x, y \in X_i$ $d(x, y) \leq B$ при $1 \leq i \leq k$?.

15. СОСТАВЛЕНИЕ УЧЕБНОГО РАСПИСАНИЯ

УСЛОВИЕ. Заданы множество H рабочих часов, множество C преподавателей, множество T учебных дисциплин, для каждого преподавателя c дано подмножество $A(c)$ из H, называемое «допустимыми часами преподавателя c», для каждой дисциплины t подмножество $A(t)$ множества H, называемое «допустимыми часами дисциплины t», и для каждой пары $(c, t) \in C \times T$ целое положительное число $R(c, t)$, называемое «требуемой нагрузкой».

ВОПРОС. Существует ли учебное расписание, обслуживающее все дисциплины? Иными словами, существует ли функция $f: C \times T \times H \to \{0,1\}$ (где $f(c, t, h) = 1$ означает, что преподаватель c занимается дисциплиной t в момент h), удовлетворяющая следующим условиям:

$f(c, t, h) = 1$ только тогда, когда $h \in$ пересечению $A(c)$ и $A(t)$;

для каждого $h \in H$ и $c \in C$ существует не более одного $t \in T$ такого, что $f(c, t, h) = 1$;

для каждого $h \in H$ и $t \in T$ существует не более одного $c \in C$ такого, что $f(c, t, h) = 1$;

для каждой пары $(c, t) \in C \times T$ существует ровно $R(c, t)$ значений h, для которых $f(c, t, h) = 1$.

16. МИНИМАЛЬНОЕ ПОКРЫТИЕ

УСЛОВИЕ. Задано семейство C подмножеств конечного множества S и положительное целое число $K \leq |C|$.

ВОПРОС. Существует ли покрытие C' из C мощности не более K? Иными словами, существует ли в C такое подмножество C', что любой элемент из S принадлежит, по крайней мере, одному подмножеству из C'?

17. МИНИМАЛЬНЫЙ НАБОР ТЕСТОВ

УСЛОВИЕ. Задано конечное множество A возможных диагнозов, набор C подмножеств множества A, представляющий тесты, и положительное целое число $J \leq |C|$.

ВОПРОС. Существует ли в C поднабор C' мощности не более J, такой, что для любой пары A_i, A_j возможных диагнозов имеется некоторый тест $c \in C'$, содержащий ровно один из них?

18. САМЫЙ ДЛИННЫЙ ПУТЬ

УСЛОВИЕ. Заданы граф $G=<V, E>$, целые положительные длины $l(e)$ для всех ребер, выделенные вершины s и t и положительное целое число B.

ВОПРОС. Существует ли в G элементарный путь из s в t, имеющий длину не менее B?

19. НЕЗАВИСИМОЕ МНОЖЕСТВО
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \leq |V|$.
ВОПРОС. Существует ли в V независимое множество мощности не менее K? Иными словами, верно ли, что существует подмножество $V' \in V$ такое, что $|V'| \geq K$ и никакие две вершины из V' не соединены ребром из E?

20. КУБИЧЕСКИЙ ПОДГРАФ
УСЛОВИЕ. Задан граф $G=<V, E>$.
ВОПРОС. Существует ли в E непустое подмножество E' такое, что в графе $G'=<V, E'>$ любая вершина имеет степень, равную 3 или 0?

21. МИНИМАЛЬНЫЙ РАЗРЕЗ
УСЛОВИЕ. Заданы граф $G=<V, E>$, вес $w(e)$ каждого ребра $e \in E$ и положительное целое число K.
ВОПРОС. Существует ли разбиение множества V на два таких непересекающихся множества V_1 и V_2, что сумма весов ребер из E, соединяющих вершины из множеств V_1 и V_2, не превосходит K?

22. МИНИМАЛЬНЫЙ РАЗРЕЗ С ОГРАНИЧЕНИЯМИ
УСЛОВИЕ. Заданы граф $G=<V, E>$ с двумя выделенными вершинами s и t, вес $w(e)$ каждого ребра $e \in E$ и положительные целые числа B и K ($B \leq |V|$).
ВОПРОС. Существует ли разбиение множества V на два таких непересекающихся множества V_1 и V_2, что $s \in V_1$, $t \in V_2$, $|V_1| \leq B$, $|V_2| \leq B$ и сумма весов ребер из E, соединяющих вершины из множеств V_1 и V_2, не превосходит K?

23. НАДЕЖНОСТЬ СЕТИ
УСЛОВИЕ. Заданы граф $G=<V, E>$, подмножество $V' \in V$, для каждого $e \in E$ рациональное число $p(e)$, $0 \leq p(e) \leq 1$ – вероятность неисправности, положительное рациональное число $q \leq 1$.
ВОПРОС. Верно ли, что с вероятностью, не меньшей q, любые две вершины из V соединены хотя бы одним путем, не содержащим неисправных ребер?

24. КРАТЧАЙШИЙ ПУТЬ С ОГРАНИЧЕНИЯМИ ПО ВЕСУ
УСЛОВИЕ. Заданы граф $G=<V, E>$ с двумя выделенными вершинами s и t, целые положительные веса $w(e)$ и длина $l(e)$ каждого ребра $e \in E$ и положительные целые числа B и K.
ВОПРОС. Существует ли в G элементарный путь из s в t, веса не более B и длины не более K?

25. СЕЛЬСКИЙ ПОЧТАЛЬОН
УСЛОВИЕ. Заданы граф $G=<V, E>$, целая положительная длина $l(e)$ каждого ребра $e \in E$, E' – подмножество E и положительное целое число K.
ВОПРОС. Существует ли в G цикл, включающий каждое ребро из E' и имеющий длину не более K?

26. КИТАЙСКИЙ ПОЧТАЛЬОН
УСЛОВИЕ. Заданы смешанный граф граф $G=<V, A, E>$, где A – множество ориентированных ребер и E – множество неориентированных ребер; целая положительная длина $l(e)$ каждого ребра $e \in A \bigcup E$, и положительное целое число B.
ВОПРОС. Существует ли в G цикл, включающий по крайней мере один раз каждое ребро и имеющий длину не более B (ориентированные ребра должны входить в цикл только с правильной ориентацией)?

27. МИНИМАЛЬНОЕ ПО МОЩНОСТИ МАКСИМАЛЬНОЕ ПАРОСОЧЕТАНИЕ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |E|$.
ВОПРОС. Существует ли в E непустое подмножество E' такое, что $|E'| \le K$ и E'-максимальное паросочетание, т.е. никакие два ребра из E' не имеют общего конца, а любое ребро из множества $E-E'$ имеет общий конец с одним из ребер множества E'?

28. СУММА РАЗМЕРОВ
УСЛОВИЕ. Заданы конечное множество A, положительные целые размеры $w(a)$ для каждого $a \in A$ и положительное целое число K.
ВОПРОС. Существует ли такое подмножество A' множества A, что сумма размеров его элементов равна K?

29. ДОМИНИРУЮЩЕЕ МНОЖЕСТВО
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Существует ли в V доминирующее множество мощности не более K? Иными словами, верно ли, что существует подмножество $V' \in V$ такое, что $|V'| \le K$ и для любой вершины $u \in V-V'$ сущеествует такая вершина $v \in V'$, что u и v соединены ребром из E?

30. РАЗБИЕНИЕ НА КЛИКИ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Можно ли разбить вершины графа G на $k \le K$ непересекающихся множеств $V_1, V_2, \ldots, V_k$ таких, что для всех i ($1 \le i \le k$) подграф, индуцированный множеством V_i, был бы полным?

31. РАЗБИЕНИЕ НА ТРЕУГОЛЬНИКИ
УСЛОВИЕ. Заданы граф $G=<V, E>$ такой, что для некоторого положительного целого число q: $|V|=3q$.
ВОПРОС. Можно ли разбить вершины графа G на q непересекающихся множеств $V_1, V_2, \ldots, V_q$ таких, что каждое содержит ровно 3 вершины и для всех i ($1 \le i \le q$) подграф, индуцированный множеством V_i, был бы треугольником?

32. РАЗБИЕНИЕ НА ЛЕСА
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Можно ли разбить вершины графа G на $k \le K$ непересекающихся множеств $V_1, V_2, \ldots, V_k$ таких, что для всех i ($1 \le i \le k$) подграф, индуцированный множеством V_i, не содержит циклов?

33. МОНОХРОМАТИЧЕСКИЙ ТРЕУГОЛЬНИК
УСЛОВИЕ. Заданы граф $G=<V,E>$.
ВОПРОС. Можно ли разбить множество ребер E на два непересекающихся подмножества E_1 и E_2, таких, что ни один из графов $G_1=<V, E_1>$ или $G_2=<V, E_2>$ не содержит треугольника?

34. ИНДУЦИРОВАННЫЙ ПУТЬ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Существует ли в V подмножество V' такое, что $|V'| \ge K$ и подграф, индуцированный множеством V', является элементарным путем на $|V'|$ вершинах?

35. ДВУДОЛЬНЫЙ ПОДГРАФ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |E|$.
ВОПРОС. Существует ли в E подмножество E' мощности не менее K такое, что $G'=<V, E'>$ – двудольный подграф?

36. СВЯЗНЫЙ ПОДГРАФ ОГРАНИЧЕННОЙ СТЕПЕНИ
УСЛОВИЕ. Заданы граф $G=<V, E>$, неотрицательное целое число $d \le |V|$ и положительное целое число $K \le |E|$.
ВОПРОС. Существует ли в E подмножество E' мощности не менее K такое, что подграф $G'=<V, E'>$ связен и не имеет вершин степени более d?

37. ГАМИЛЬТОНОВО ПОПОЛНЕНИЕ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \le |V|$.
ВОПРОС. Существует ли множество E', содержащее множество E такое, что $|E'–E| \le K$, а граф $G'=<V, E'>$ имеет гамильтонов цикл?

38. КАРКАС ОГРАНИЧЕННОГО ДИАМЕТРА
УСЛОВИЕ. Заданы граф $G=<V, E>$, вес положительный целый вес $w(e)$ каждого ребра $e \in E$ и положительные целые числа B и $D \le |V|$.
ВОПРОС. Существует ли в графе G каркас T такой, что сумма весов ребер из T не превосходит B и в T нет элементарного пути, число ребер которого не превосходит D?

39. РАСЩЕПЛЕНИЕ МНОЖЕСТВА
УСЛОВИЕ. Задан набор C подмножеств множества S.
ВОПРОС. Существует ли такое разбиение множества S на два подмножества, что ни одно подмножество из C не содержится целиком ни в S_1, ни в S_2?

40. ЗАДАЧА О *P*-ЦЕНТРЕ
УСЛОВИЕ. Заданы граф $G=<V, E>$, целый неотрицательный вес $w(v)$ каждой вершины $v \in V$, целая неотрицательная длина $A(e)$ каждого ребра $e \in E$, положительное целое число $K \le |V|$ и положительное рациональное число B.
ВОПРОС. Существует ли такое множество P мощности K, состоящее из точек на G (точкой на G называется либо вершина графа G, либо точка на ребре $e \in E$, где e рассматривается как прямолинейный отрезок длины $A(e)$), что если $D(v)$ – длина кратчайшего пути от вершины v до ближайшей к ней точке из P, то $Max\{D(v) \times w(v)\} \le B$?
Комментарий. При $P \subseteq V$ (выборе центра среди вершин) задача остается NP-полной. Если граф G – дерево, или K – фиксировано, то задача разрешима за полиномиальное время.

41. ЗАДАЧА О *P*-МЕДИАНЕ
УСЛОВИЕ. Заданы граф $G=<V, E>$, целый неотрицательный вес $w(v)$ каждой вершины $v \in V$, целая неотрицательная длина $A(e)$ каждого ребра $e \in E$, положительное целое число $K \le |V|$ и положительное рациональное число B.
ВОПРОС. Существует ли такое множество P мощности K, состоящее из точек на G (точкой на G называется либо вершина графа G, либо точка на ребре $e \in E$, где e рассматривается как прямолинейный отрезок длины $A(e)$), что если $D(v)$ – длина кратчайшего пути от вершины v до ближайшей к ней точке из P, то $Max\{D(v) \times w(v)\} \le B$?
Комментарий. При $P \subseteq V$ (выборе медианы среди вершин) задача остается NP-полной. Если граф G – дерево, или K – фиксировано, то задача разрешима за полиномиальное время.

42. МАКСИМАЛЬНОЕ ЧИСЛО ОГРАНИЧЕННЫХ НЕПЕРЕСЕКАЮЩИХСЯ ПУТЕЙ
УСЛОВИЕ. Задан граф $G=<V, E>$ с выделенными вершинами s и t и заданы положительные целые числа B и K (B и $K \le |V|$).
ВОПРОС. Верно ли что в G содержится не менее B путей из s в t, попарно не имеющих общих вершин и включающих не более K ребер?

43. КАРКАС ОГРАНИЧЕННОЙ СТЕПЕНИ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \leq |V|$.
ВОПРОС. Существует ли в графе G каркас, у которого степени всех вершин не превосходят K?

44. КАРКАС С МАКСИМАЛЬНЫМ ЧИСЛОМ ЛИСТОВ
УСЛОВИЕ. Заданы граф $G=<V, E>$ и положительное целое число $K \leq |V|$.
ВОПРОС. Существует ли в графе G каркас, у которого не менее K вершин степени 1?

45. ДЕРЕВО ШТЕЙНЕРА
УСЛОВИЕ. Заданы граф $G=<V, E>$, положительный целый вес $w(e)$ каждого ребра $e \in E$, подмножество R вершин графа и положительное целое число B.
ВОПРОС. Существует ли в графе G дерево, содержащее все вершины из R, что сумма весов ребер этого поддерева не превосходит B?

46. МАКСИМАЛЬНОЕ ЧИСЛО ОГРАНИЧЕННЫХ НЕПЕРЕСЕКАЮЩИХСЯ ПУТЕЙ
УСЛОВИЕ. Заданы граф $G=<V, E>$ с выделенными вершинами s и t, а также положительные целые числа J и K.
ВОПРОС. Верно ли, что в G содержится не менее J различных путей из s в t, попарно не имеющих общих вершин и включающих не более K ребер?

Задания для оценки сформированности компетенций

Направление подготовки 09.03.01 - Информатика и вычислительная техника

Направленность (профиль) Программное обеспечение средств вычислительной техники и автоматизированных систем

Проверяемая компетенция ОПК-1

Наименование дисциплины Алгоритмы и теория сложности

Задания:

1. Что такое детерминированная машина Тьюринга?

а)Бесконечный автомат.

б)Универсальный отладчик.

в)Универсальный интерпретатор.

г)Модель универсального вычислителя.

2.Для решения NP-полных задач…

а) Нужны очень сложные алгоритмы.

б) Нужны программы с большим количеством строчек программного кода.

в) Нужно напечатать экспоненциальное число ответов.

г)Не придуманы полиномиальные алгоритмы.

3. Экспоненциальная вычислительная сложность – это …

а)линейная сложность от размерности входных данных.

б)квадратичная сложность от размерности входных данных.

в)$O(n^2)$, где n – размерность входных данных.

г)$O(2^n)$и выше, n – размерность входных данных.

Проверяемая компетенция ОПК-8
Наименование дисциплины Алгоритмы и теория сложности
Задания:

4. Алгоритмом с возвратом решаем задачу о минимальном маршруте коммивояжера. Вычислительная сложность этого алгоритма …

а)Линейная от числа городов.

б)Квадратичная от числа городов.

в)$O(n^2)$, где n – число городов.

г)Экспоненциальная от числа городов.

5.Задача об упаковке в минимальное число контейнеров …

а) Принадлежит к классу полиномиальных алгоритмов.

б) Принадлежит к классу экспоненциальных алгоритмов.

в) Принадлежит к классу *NP*-полных задач.

г)Решается за время $O(n^2)$, где n – количество предметов.

6. Задачу о наборе заданной целой положительной суммы с помощью заданных целых положительных слагаемых …

а) Можно решить за полиномиальное время.

б)Нельзя решить за полиномиальное время, это доказано.

в)Это алгоритмически неразрешимая задача.

г)Нет правильного ответа.

Ключ к заданию для оценки сформированности компетенций

Шифр компетенции	**ОПК-1**			**ОПК-8**		
№ вопроса	**1**	**2**	**3**	**4**	**5**	**6**
Правильный вариант ответа	**г**	**г**	**г**	**г**	**в**	**г**

Критерии оценивания:

ОПК-1:
0 правильных ответа – «неудовлетворительно»,
1 правильный ответ – «удовлетворительно»,

2 правильных ответа – «хорошо»,
3 правильных ответов – «отлично»

ОПК-8:
0 правильных ответа – «неудовлетворительно»,
1 правильный ответ – «удовлетворительно»,
2 правильных ответа – «хорошо»,
3 правильных ответов – «отлично»

Приложение 2

Код индикатора	Индикатор достижения компетенции	Оценочные средства
ОПК-1: Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности		
ОПК-1.1:	Решает стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования	*Теоретические вопросы* 1. Понятие алгоритма. Математические модели алгоритма. Классификация языков программирование по математической модели алгоритма. 2. Детерминированная машина Тьюринга (ДМТ): «чёрный ящик» и структурная схема. 3. Универсальная машина Тьюринга (универсальный интерпретатор). Архитектура фон Неймана. 4. Алгоритмически неразрешимые проблемы. Проблема остановки машины Тьюринга. 5. Тезис Тьюринга. 6. Примитивно-рекурсивные функции. 7. Доказательство примитивной рекурсивности арифметических операций. 8. Частично-рекурсивные функции. 9. Тезис Чёрча. 10. Эквивалентность моделей ДМТ и вычислимой функции. 11. Понятие вычислительной сложности алгоритма как числа шагов детерминированной машины Тьюринга. 12. Недетерминированная машина Тьюринга (НДМТ). 13. Понятие вычислительной сложности алгоритма как числа шагов недетерминированной машины Тьюринга. 14. Классификация алгоритмов и задач по вычислительной сложности. 15. Определение полиномиальной сводимости. Класс NP-полных задач. 16. Эквивалентность NP-полных задач. 17. Доказательство NP-полноты задачи методом сужения. 18. Точные методы решения NP-полных задач. 19. Общая схема алгоритма с возвратом. 20. Отсечение повторяющихся решений. Генерация решений в лексикографическом порядке. 21. Модификация общей схемы для решения задач на минимум. 22. Модификация общей схемы для решения задач на максимум. Принцип включения-невключения. 23. Понятие задачи оптимизации. Решение NP-полных задач оптимизации приближёнными алгоритмами. 24. Понятие абсолютной погрешности приближённого решения задачи оптимизации. 25. Понятие относительной погрешности приближённого решения задачи оптимизации. 26. Верхние и нижние оценки погрешности приближённых алгоритмов. 27. Приближённые алгоритмы для задачи «Упаковка в контейнеры». FF-алгоритм. FFD-алгоритм.

ОПК-1.2	Решает профессиональные задачи с применением методов теоретического и экспериментального исследования	*Практические задания* Задание 1 Решить задачу «Упаковка в контейнеры» точным и FFD-алгоритмом. Для FFD-алгоритма найти «плохой» пример, построить бесконечную серию «плохих» примеров, найти относительную погрешность. Контрольные вопросы: 1. Назовите целевую функцию и ограничение данной задачи. 2. Какое условие необходимо проверять при упаковке каждого предмета? 3. В каком случае необходимо увеличить число использованных контейнеров? 4. Сколько контейнеров будет заполнено в худшем случае? 5. Будет ли однажды упакованный предмет переложен в другой контейнер при упаковке точным алгоритмом? FFD-алгоритмом? 6. Какова сложность точного алгоритма? FFD-алгоритма? Задание 2 Запишите рекурсивный вариант алгоритма с возвратом. Контрольные вопросы: 1. При выполнении какого условия рекурсия останавливается? 2. Перечислите последовательность шагов при реализации «прямого хода». 3. Перечислите последовательность шагов возврата. 4. Реализуйте алгоритм с возвратом для решения любой модельной NP-полной задачи
ОПК-8: Способен разрабатывать алгоритмы и программы, пригодные для практического применения		
ОПК-8.1	Определяет средства разработки программных средств для решения практических задач профессиональной деятельности	*Теоретические вопросы:* 1. Машинное представление графов. Списки инцидентности. 2. Различные стратегии систематического обхода графов. Поиск в глубину на неориентированном графе. Поиск в ширину на неориентированном графе. 3. Способы построения стягивающего дерева неориентированного графа. 4. Фундаментальное множество циклов неориентированного графа. 5. Блоки, точки сочленения неориентированного графа. 6. Эйлеров путь, эйлеров цикл в неориентированном графе. 7. Переносимость алгоритмов поиска в глубину и ширину на ориентированные графы. 8. Классификация задач по степени сложности. 9. Сравнить алгоритмы Форда-Беллмана, Дейкстры, Флойда по следующим критериям: - тип графа; - результат работы; - вычислительная сложность.
ОПК-8.2	Разрабатывает алгоритмы и программы для решения прикладных задач различных классов	*Практические заданья*: 1. Разделить неориентированный граф на компоненты связности с помощью поиска в глубину и поиска в ширину. 2. На модельном ненагруженном неориентированном графе найти кратчайший путь между парой фиксированных вершин. 3. Задан неориентированный связный граф и вершина v. За полиномиальное время определить, проходит ли через v хотя бы один цикл.

Код индикатора	Индикатор достижения компетенции	Оценочные средства
		4. Найти компоненты двусвязности и точки сочленения модельного неориентированного графа. 6. На модельном нагруженном орграфе найти кратчайший путь между парой фиксированных вершин. 7. Найти в московском метро кратчайший путь между двумя станциями. Сколькими известными вами алгоритмами это можно сделать? 8. Раскрасить географическую карту в минимальное число цветов так, чтобы соседние страны не были закрашены одним цветом. *Задание к курсовой работе «Оптимальное размещение обслуживающих центров»* 1. Оптимально разместить заданный тип обслуживающего центра на графе заданного типа. 2. Придумать реальную задачу, соответствующую математической постановке.