

**МИНОБРНАУКИ РОССИИ**

**Федеральное государственное автономное образовательное  
учреждение высшего образования "Пермский  
государственный национальный исследовательский  
университет"**

**Кафедра прикладной математики и информатики**

**Авторы-составители: Перескокова Ольга Ивановна  
Бузмакова Мария Михайловна  
Русакова Ольга Леонидовна  
Шейна Татьяна Юрьевна  
Кнутова Наталия Сергеевна**

Рабочая программа дисциплины  
**АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ II**  
Код УМК 95931

Утверждено  
Протокол №1  
от «31» августа 2021 г.

Пермь, 2021

## **1. Наименование дисциплины**

Алгоритмизация и программирование II

## **2. Место дисциплины в структуре образовательной программы**

Дисциплина входит в обязательную часть Блока « Б.1 » образовательной программы по направлениям подготовки (специальностям):

Направление: **02.03.02** Фундаментальная информатика и информационные технологии  
направленность Открытые информационные системы

### **3. Планируемые результаты обучения по дисциплине**

В результате освоения дисциплины **Алгоритмизация и программирование II** у обучающегося должны быть сформированы следующие компетенции:

**02.03.02** Фундаментальная информатика и информационные технологии (направленность : Открытые информационные системы)

**ОПК.2** Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения

#### **Индикаторы**

**ОПК.2.1** Использует знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности для практического применения

**ОПК.2.2** Анализирует типовые языки программирования, составляет программы

**ОПК.2.3** Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения

#### 4. Объем и содержание дисциплины

<b>Направления подготовки</b>	02.03.02 Фундаментальная информатика и информационные технологии (направленность: Открытые информационные системы)
<b>форма обучения</b>	очная
<b>№№ триместров, выделенных для изучения дисциплины</b>	2
<b>Объем дисциплины (з.е.)</b>	6
<b>Объем дисциплины (ак.час.)</b>	216
<b>Контактная работа с преподавателем (ак.час.), в том числе:</b>	84
<b>Проведение лекционных занятий</b>	28
<b>Проведение практических занятий, семинаров</b>	28
<b>Проведение лабораторных работ, занятий по иностранному языку</b>	28
<b>Самостоятельная работа (ак.час.)</b>	132
<b>Формы текущего контроля</b>	Защищаемое контрольное мероприятие (2) Итоговое контрольное мероприятие (1) Письменное контрольное мероприятие (1)
<b>Формы промежуточной аттестации</b>	Экзамен (2 триместр)

## 5. Аннотированное описание содержания разделов и тем дисциплины

**Структуры и объединения. Перебор с возвратом. Динамическое программирование. Динамические структуры данных.**

Структуры (struct): описание и использование. Перечисления (enum). Объединения (union). Вложенные структуры. Массивы структур. Совместное использование структур и перечислений, структур и объединений.

Рекурсия. Перебор с возвратом. Генерация перестановок.

Динамическое программирование

Динамические структуры данных. Связные списки, их типы. Стек и очередь. Основные операции со связными списками: создание списка, уничтожение списка, поиск.

Вставка и удаление в списке. Перестановка элементов списка.

Бинарные деревья: основные операции. Деревья выражений.

Понятие сбалансированности дерева. AVL-деревья. Вставка и удаление в AVL-дереве. B-деревья.

Красно-черные деревья.

Деревья выражений. Инфиксная, префиксная и постфиксная формы записи выражений. Построение дерева выражения. Тожественные преобразования в деревьях выражений.

Контейнеры и итераторы. Стандартные классы: vector, list, stack, queue.

Стандартные контейнерные классы: map, multimap, set и т.д. Библиотека algorithm, numeric.

Графы. Алгоритмы на графах.

**Деревья. AVL-деревья. Графы. Алгоритмы на графах. Библиотека STL**

Бинарные деревья: основные операции. Деревья выражений.

Понятие сбалансированности дерева. AVL-деревья. Вставка и удаление в AVL-дереве. B-деревья.

Красно-черные деревья.

Деревья выражений. Инфиксная, префиксная и постфиксная формы записи выражений. Построение дерева выражения. Тожественные преобразования в деревьях выражений.

Контейнеры и итераторы. Стандартные классы: vector, list, stack, queue.

Стандартные контейнерные классы: map, multimap, set и т.д. Библиотека algorithm, numeric.

Графы. Алгоритмы на графах.

**Индивидуальная работа**

Индивидуальная работа по теме "Динамические структуры данных, деревья, графы", оформление отчета, создание презентации и защита работы

**Экзамен**

Контрольное мероприятие в письменной форме, включающее два теоретических вопроса (билеты) и практические задания (написание кода программ для поставленных задач)

## **6. Методические указания для обучающихся по освоению дисциплины**

Освоение дисциплины требует систематического изучения всех тем в той последовательности, в какой они указаны в рабочей программе.

Основными видами учебной работы являются аудиторские занятия. Их цель - расширить базовые знания обучающихся по осваиваемой дисциплине и систему теоретических ориентиров для последующего более глубокого освоения программного материала в ходе самостоятельной работы. Обучающемуся важно помнить, что контактная работа с преподавателем эффективно помогает ему овладеть программным материалом благодаря расстановке необходимых акцентов и удержанию внимания интонационными модуляциями голоса, а также подключением аудио-визуального механизма восприятия информации.

Самостоятельная работа преследует следующие цели:

- закрепление и совершенствование теоретических знаний, полученных на лекционных занятиях;
- формирование навыков подготовки текстовой составляющей информации учебного и научного назначения для размещения в различных информационных системах;
- совершенствование навыков поиска научных публикаций и образовательных ресурсов, размещенных в сети Интернет;
- самоконтроль освоения программного материала.

Обучающемуся необходимо помнить, что результаты самостоятельной работы контролируются преподавателем во время проведения мероприятий текущего контроля и учитываются при промежуточной аттестации.

Обучающимся с ОВЗ и инвалидов предоставляется возможность выбора форм проведения мероприятий текущего контроля, альтернативных формам, предусмотренным рабочей программой дисциплины. Предусматривается возможность увеличения в пределах 1 академического часа времени, отводимого на выполнение контрольных мероприятий.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации.

При проведении текущего контроля применяются оценочные средства, обеспечивающие передачу информации, от обучающегося к преподавателю, с учетом психофизиологических особенностей здоровья обучающихся.

## **7. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

При самостоятельной работе обучающимся следует использовать:

- конспекты лекций;
- литературу из перечня основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля);
- текст лекций на электронных носителях;
- ресурсы информационно-телекоммуникационной сети "Интернет", необходимые для освоения дисциплины;
- лицензионное и свободно распространяемое программное обеспечение из перечня информационных технологий, используемых при осуществлении образовательного процесса по дисциплине;
- методические указания для обучающихся по освоению дисциплины.

## 8. Перечень основной и дополнительной учебной литературы

### Основная:

1. Топп Уильям, Форд Уильям Структуры данных в С++: Пер. с англ./Под ред. В. Кузьменко. - М.: Бином, 2000, ISBN 5-7989-0017-7.-816.
2. Букунов С.В. Основы программирования на языке С++ [Электронный ресурс] : учебное пособие / С.В. Букунов. — Электрон. текстовые данные. — СПб. : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. — 201 с. — 978-5-9227-0619-3. — Режим доступа: <http://www.iprbookshop.ru/63631.html> <http://www.iprbookshop.ru/63631.html>
3. Практикум по информатике: учебное пособие/Н. М. Андреева [и др.].-Санкт-Петербург: Лань, 2018, ISBN 978-5-8114-2961-5.-248.-Библиогр.: с. 244
4. Павловская Т. А. С/С++. Программирование на языке высокого уровня: Учеб. для студентов вузов, обучающихся по напр. "Информатика и вычисл. техника"/Т. А. Павловская.-СПб.: Питер, 2002, ISBN 5-318-00001-0.-464.
5. Подбельский В. В. Язык Си++: учебное пособие для студентов вузов, обучающихся по направлениям "Прикладная математика" и "Вычислительные машины, комплексы, системы и сети"/В. В. Подбельский.-Москва: Финансы и статистика, 2003, ISBN 5-279-02204-7.-560.-Библиогр.: с. 538-539
6. Страуструп Б. Язык программирования Си++/Б. Страуструп ; пер. с англ.: М. Г. Пиголкин, В. А. Ясницкий.-Москва: Радио и связь, 1991, ISBN 5-256-00454-9.-352.

### Дополнительная:

1. Страуструп Б. Язык программирования С++ (стандарт С++11): Краткий курс : перевод с английского/Б. Страуструп ; пер., ред. Н. Н. Мартынов.-Москва: Издательский дом Бином, 2017, ISBN 978-5-9518-0699-4.-176.-Библиогр.: с. 173-175
2. Вирт Н. Алгоритмы и структуры данных/Н. Вирт ; пер. Д. Б. Подшивалов.-Москва: Мир, 1989, ISBN 5-03-001045-9.-360.
3. Вирт Н. Алгоритмы + структуры данных - программы/Н. Вирт ; пер. Л. Ю. Иоффе ; ред. Д. Б. Подшивалов.-Москва: Мир, 1985.-406.-Указ.: с. 400-404
4. Ахо Альфред В., Хопкрофт Джон Э., Ульман Джеффри Д. Структуры данных и алгоритмы: Пер. с англ./Альфред В. Ахо, Джон Э. Хопкрофт, Джеффри Д. Ульман.-М.; СПб.; Киев: Вильямс, 2001, ISBN 5-8459-0122-7.-384.-Библиогр.: с. 369-374
5. Королев Л. Н., Миков А. И. Информатика. Введение в компьютерные науки: [учебник для вузов]/Л. Н. Королев, А. И. Миков.-Москва: Высшая школа, 2012, ISBN 978-5-4372-0020-9.-3661.-Библиогр.: с. 346-347
6. Алексеев, В. Е. Графы и алгоритмы : учебное пособие / В. Е. Алексеев, В. А. Таланов. — 3-е изд. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 153 с. — ISBN 978-5-4497-0366-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. <http://www.iprbookshop.ru/89434.html>

## **9. Перечень ресурсов сети Интернет, необходимых для освоения дисциплины**

<https://code-live.ru/tag/cpp-manual/> Портал о программировании

<http://cppstudio.com/cat/274/> Язык программирования C++

<http://www.psu.ru/elektronnye-resursy-dlya-psu> Электронные ресурсы для ПГНИУ

<http://window.edu.ru/> Единое окно доступа к образовательным ресурсам

<http://ptaskbook.com> Электронный задачник Programming TaskBook

## **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

Образовательный процесс по дисциплине **Алгоритмизация и программирование II** предполагает использование следующего программного обеспечения и информационных справочных систем: используемых информационных технологий Образовательный процесс по дисциплине предполагает использование следующего программного обеспечения и информационных справочных систем:

- доступ в режиме on-line в Электронную библиотечную систему (ЭБС);
- доступ в электронную информационно-образовательную среду университета.

Необходимое лицензионное и (или) свободно распространяемое программное обеспечение:

- приложение позволяющее просматривать и воспроизводить медиаконтент PDF-файлов «Adobe Acrobat Reader DC».
- офисный пакет приложений «LibreOffice», Alt Linux.

Специализированное программное обеспечение:

- среда разработки на языке программирования высокого уровня C++;
- Programming Taskbook - свободно распространяемый электронный задачник по программированию (автор - Абрамян М.Э.).

При освоении материала и выполнения заданий по дисциплине рекомендуется использование материалов, размещенных в Личных кабинетах обучающихся ЕТИС ПГНИУ ([student.psu.ru](http://student.psu.ru)).

При организации дистанционной работы и проведении занятий в режиме онлайн могут использоваться:

- система видеоконференцсвязи на основе платформы BigBlueButton (<https://bigbluebutton.org/>).
- система LMS Moodle (<http://e-learn.psu.ru/>), которая поддерживает возможность использования текстовых материалов и презентаций, аудио- и видеоконтент, а так же тесты, проверяемые задания, задания для совместной работы.
- система тестирования Indigo (<https://indigotech.ru/>).

## **11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**

Для лекционных занятий требуется аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Для проведения практических занятий - аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Для проведения лабораторных занятий - компьютерный класс. Состав оборудования определен в Паспорте компьютерного класса.

Для групповых (индивидуальных) консультаций - аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Для проведения текущего контроля - аудитория, оснащенная меловой (и) или маркерной доской.

Самостоятельная работа студентов: аудитория, оснащенная компьютерной техникой с возможностью подключения к сети «Интернет», с обеспеченным доступом в электронную информационно-образовательную среду университета, помещения Научной библиотеки ПГНИУ

Помещения научной библиотеки ПГНИУ для обеспечения самостоятельной работы обучающихся:

1. Научно-библиографический отдел, корп.1, ауд. 142. Оборудован 3 персональными компьютера с доступом к локальной и глобальной компьютерным сетям.

2. Читальный зал гуманитарной литературы, корп. 2, ауд. 418. Оборудован 7 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

3. Читальный зал естественной литературы, корп.6, ауд. 107а. Оборудован 5 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

4. Отдел иностранной литературы, корп.2 ауд. 207. Оборудован 1 персональным компьютером с доступом к локальной и глобальной компьютерным сетям.

5. Библиотека юридического факультета, корп.9, ауд. 4. Оборудована 11 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

6. Читальный зал географического факультета, корп.8, ауд. 419. Оборудован 6 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

Все компьютеры, установленные в помещениях научной библиотеки, оснащены следующим программным обеспечением:

Операционная система ALT Linux;

Офисный пакет Libreoffice.

Справочно-правовая система «КонсультантПлюс»

**Фонды оценочных средств для аттестации по дисциплине  
Алгоритмизация и программирование II**

**Планируемые результаты обучения по дисциплине для формирования компетенции.  
Индикаторы и критерии их оценивания**

**ОПК.2**

**Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения**

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
<p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p>	<p>Умеет использовать метод перебора с возвратом и метод динамического программирования решения конкретной задачи. Умеет выбирать эффективный метод решения задачи по ее постановке и обосновывать эффективность выбранного подхода.</p>	<p align="center"><b>Неудовлетворител</b></p> <p>Не знать синтаксис описания структур, перечислений, массивов структур и объединений. Не уметь использовать структуры при реализации алгоритмов. Не уметь выполнять сортировку массива структур с помощью вспомогательного массива указателей. Не знать принципы работы алгоритма перебора с возвратом и не умение реализовывать этот алгоритм для решения конкретных задач. Не знание метода динамического программирования и не умение реализовывать этот метод для решения конкретной задачи. Не умение восстановить подмножество данных, соответствующее оптимальному решению задачи по ответу в методе динамического программирования.</p> <p align="center"><b>Удовлетворительн</b></p> <p>Знать синтаксис описания структур, перечислений, массивов структур и объединений, уметь использовать структуры при реализации простейших алгоритмов. Знать принципы реализации сортировки массива структур с помощью вспомогательного массива указателей. Знать принципы работы алгоритма перебора с возвратом и умение реализовывать этот алгоритм для решения типовых конкретных задач по образцу. Знание метода динамического программирования и умение реализовывать этот метод для решения типовых конкретных задач по образцу. Иметь представление о способах восстановить подмножество данных, соответствующее оптимальному решению</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Удовлетворительн</b></p> <p>задачи по ответу в методе динамического программирования.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>Знать синтаксис описания структур, перечислений, массивов структур и объединений. Уметь использовать структуры при реализации алгоритмов. Уметь выбирать эффективную структуру данных в зависимости от поставленной задачи. Уметь выполнять сортировку массива структур с помощью вспомогательного массива указателей и уметь доказывать эффективность такой сортировки в сравнении с традиционными методами. Знать принципы работы алгоритма перебора с возвратом и не умение реализовывать этот алгоритм для решения конкретных задач. Знание метода динамического программирования и не умение реализовывать этот метод для решения конкретной задачи. Уметь восстановить подмножество данных, соответствующее оптимальному решению задачи по ответу в методе динамического программирования. Уметь выбирать эффективным метод решения задачи по ее постановки и обосновывать эффективность выбранного подхода. При реализации алгоритмов на языке программирования допускаются алгоритмические ошибки.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>Знать синтаксис описания структур, перечислений, массивов структур и объединений. Уметь использовать структуры при реализации алгоритмов. Уметь выбирать эффективную структуру данных в зависимости от поставленной задачи. Уметь выполнять сортировку массива структур с помощью вспомогательного массива указателей и уметь доказывать эффективность такой сортировки в сравнении с традиционными методами. Знать принципы работы алгоритма перебора с возвратом и не умение реализовывать этот</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Отлично</b></p> <p>алгоритм для решения конкретных задач. Знание метода динамического программирования и не умение реализовывать этот метод для решения конкретной задачи. Уметь восстановить подмножество данных, соответствующее оптимальному решению задачи по ответу в методе динамического программирования. Уметь выбирать эффективным метод решения задачи по ее постановки и обосновывать эффективность выбранного подхода.</p>
<p><b>ОПК.2.1</b> Использует знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности для практического применения</p>	<p>Знать основные средства для реализации сложных структур данных, уметь использовать библиотеку STL при разработке программ, владеть навыками использования стандартных алгоритмов для работы с динамическими структурами данных.</p>	<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>Не знает синтаксис описания структур, перечислений, массивов структур и объединений. Не умеет использовать структуры при реализации алгоритмов. Не умеет выполнять сортировку массива структур с помощью вспомогательного массива указателей. Не умеет работать с бинарными файлами. Не умеет работать с динамическими структурами данных типа стек, очередь, кольцо, дек, список. Не имеет представления о бинарных деревьях, деревьях минимальной высоты, деревьях общего вида. Не знает алгоритмы их обхода, не умеет выполнять процедуры добавления и удаления вершин дерева. Не умеет выполнять балансировку дерева. Не знает основные алгоритмы работы с графами и не умеет использовать графы при решении широкого круга задач (поиск кратчайшего пути, топологическая сортировка, построение каркасного дерева, раскраска графа и т.д.).</p> <p>Не знает синтаксис описания бинарных деревьев. Не умеет выполнять основные операции с бинарными деревьями. Не умеет выполнять балансировку AVL-деревьев. Не знает принципы работы с различными контейнерами и алгоритмами из библиотеки STL. Не умеет применять инструментарий STL при решении несложных задач. Не знает</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>алгоритмы на графах: поиск в ширину, поиск в глубину, поиск кратчайших путей, подсчет количества компонент связности, топологическая сортировка, построение минимального остовного дерева и т.д. Не умеет реализовывать эти алгоритмы на языке программирования.</p> <p style="text-align: center;"><b>Удовлетворительн</b></p> <p>Уметь использовать структуры и объединения. Уметь работать с бинарными файлами. Уметь работать с динамическими структурами данных типа стек, очередь, кольцо, дек, список. Иметь представление о бинарных деревьях, деревьях минимальной высоты, деревьях общего вида. Знать алгоритмы их обхода, уметь выполнять процедуры добавления и удаления вершин дерева. Уметь выполнять балансировку дерева. Знать основные алгоритмы работы с графами и уметь использовать графы при решении широкого круга задач (поиск кратчайшего пути, топологическая сортировка, построение каркасного дерева, раскраска графа и т.д.).</p> <p>Знать синтаксис описания бинарных деревьев. Уметь выполнять основные операции с бинарными деревьями, создавая код программы по образцу. Уметь выполнять балансировку AVL-деревьев без реализации на ЯП. Знать принципы работы с различными контейнерами и алгоритмами из библиотеки STL. Уметь применять инструментарий STL при решении несложных задач по образцу. Знать алгоритмы на графах: поиск в ширину, поиск в глубину, поиск кратчайших путей, подсчет количества компонент связности, топологическая сортировка, построение минимального остовного дерева и т.д. Уметь реализовывать эти алгоритмы на языке программирования по образцу.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>Уметь использовать структуры и объединения. Уметь работать с бинарными</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Хорошо</b></p> <p>файлами. Уметь работать с динамическими структурами данных типа стек, очередь, кольцо, дек, список. Иметь представление о бинарных деревьях, деревьях минимальной высоты, деревьях общего вида. Знать алгоритмы их обхода, уметь выполнять процедуры добавления и удаления вершин дерева. Уметь выполнять балансировку дерева. Знать основные алгоритмы работы с графами и уметь использовать графы при решении широкого круга задач (поиск кратчайшего пути, топологическая сортировка, построение каркасного дерева, раскраска графа и т.д.). Знать синтаксис описания бинарных деревьев. Уметь выполнять основные операции с бинарными деревьями. Уметь выполнять балансировку AVL-деревьев без реализации на ЯП. Знать принципы работы с различными контейнерами и алгоритмами из библиотеки STL. Уметь применять инструментарий STL при решении несложных задач. Знать алгоритмы на графах: поиск в ширину, поиск в глубину, поиск кратчайших путей, подсчет количества компонент связности, топологическая сортировка, построение минимального остовного дерева и т.д. Уметь реализовывать эти алгоритмы на языке программирования.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>Уметь использовать структуры и объединения. Уметь работать с бинарными файлами. Уметь работать с динамическими структурами данных типа стек, очередь, кольцо, дек, список. Иметь представление о бинарных деревьях, деревьях минимальной высоты, деревьях общего вида. Знать алгоритмы их обхода, уметь выполнять процедуры добавления и удаления вершин дерева. Уметь выполнять балансировку дерева. Знать основные алгоритмы работы с графами и уметь использовать графы при решении широкого круга задач (поиск кратчайшего пути, топологическая</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Отлично</b></p> <p>сортировка, построение каркасного дерева, раскраска графа и т.д.).  Знать синтаксис описания бинарных деревьев. Уметь выполнять основные операции с бинарными деревьями. Уметь выполнять балансировку AVL-деревьев.  Иметь представление о различных видах деревьев поиска: B-деревья, красно-черные деревья, идеально-сбалансированные деревья. Знать принципы работы с различными контейнерами и алгоритмами из библиотеки STL. Уметь применять инструментарий STL при решении задач.  Уметь расширять стандартные возможности библиотеки STL с учетом специфики задачи при помощи перегрузки методов. Знать алгоритмы на графах: поиск в ширину, поиск в глубину, поиск кратчайших путей, подсчет количества компонент связности, топологическая сортировка, построение минимального остовного дерева и т.д. Уметь реализовывать эти алгоритмы на языке программирования и оценивать эффективность реализации.</p>
<p><b>ОПК.2.3</b>  Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p>	<p>Уметь эффективно применять полученные знания в области программирования при разработке сложной программы, владеть навыками разработки "дружественного" интерфейса для пользователя с защитой от неверного ввода данных, уметь последовательно изложить применяемый в задаче алгоритм решения, уметь доказать правильность его работы.</p>	<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>Не уметь провести все этапы жизненного цикла разработки приложений от проектирования до внедрения. Не уметь выбирать структуры данных и алгоритмы для решения поставленной задачи и обосновывать свой выбор. Не уметь проводить тестирование своего кода, разрабатывать набор тестов. Не способностью оформлять документацию проекта. Не уметь реализовать поставленную задачу в виде проекта, способного преодолеть барьер внешнего тестирования.</p> <p style="text-align: center;"><b>Удовлетворительн</b></p> <p>Уметь провести все этапы жизненного цикла разработки приложений от проектирования до внедрения. Уметь выбирать структуры данных и алгоритмы для решения поставленной задачи на основе типовых решений и шаблонов. Уметь проводить</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Удовлетворительн</b></p> <p>тестирование своего кода, разрабатывать набор тестов по образцу. Умение оформлять документацию проекта аналогично данному шаблону. Уметь реализовать поставленную задачу в виде проекта, способного преодолеть барьер внешнего тестирования с помощью преподавателя.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>Уметь провести все этапы жизненного цикла разработки приложений от проектирования до внедрения. Уметь выбирать структуры данных и алгоритмы для решения поставленной задачи и обосновывать свой выбор. Уметь проводить тестирование своего кода, разрабатывать набор тестов. Умение оформлять документацию проекта. Уметь реализовать поставленную задачу в виде проекта, способного преодолеть барьер внешнего тестирования, возможно не с первого раза</p> <p style="text-align: center;"><b>Отлично</b></p> <p>Уметь провести все этапы жизненного цикла разработки приложений от проектирования до внедрения. Уметь эффективно выбирать структуры данных и алгоритмы для решения поставленной задачи. Уметь качественно проводить тестирование своего кода, разрабатывать максимально полный набор тестов. Умение оформлять документацию проекта. Уметь реализовать поставленную задачу в виде проекта, способного преодолеть барьер внешнего тестирования.</p>

## Оценочные средства текущего контроля и промежуточной аттестации

Схема доставки : Базовая

**Вид мероприятия промежуточной аттестации :** Экзамен

**Способ проведения мероприятия промежуточной аттестации :** Оценка по дисциплине в рамках промежуточной аттестации определяется на основе баллов, набранных обучающимся на контрольных мероприятиях, проводимых в течение учебного периода.

**Максимальное количество баллов :** 100

### Конвертация баллов в отметки

«отлично» - от 81 до 100

«хорошо» - от 61 до 80

«удовлетворительно» - от 41 до 60

«неудовлетворительно» / «незачтено» менее 41 балла

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы	Структуры и объединения. Перебор с возвратом. Динамическое программирование. Динамические структуры данных. <b>Защищаемое контрольное мероприятие</b>	Умеет использовать метод перебора с возвратом и метод динамического программирования решения конкретной задачи. Умеет выбирать эффективный метод решения задачи по её постановке и обосновывать эффективность выбранного подхода. Умение работать с динамическими структурами данных
<b>ОПК.2.1</b> Использует знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности для практического применения	Деревья. AVL-деревья. Графы. Алгоритмы на графах. Библиотека STL <b>Письменное контрольное мероприятие</b>	Знание основных средств для реализации сложных структур данных. Умение использовать библиотеку STL при разработке программ. Владение навыками использования стандартных алгоритмов для работы с динамическими структурами данных (графы, бинарные деревья).

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p>	<p>Индивидуальная работа <b>Защищаемое контрольное мероприятие</b></p>	<p>Умение эффективно применять полученные знания в области программирования при разработке сложной программы. Владение навыками разработки "дружественного" интерфейса для пользователя с защитой от неверного ввода данных. Умение последовательно изложить применяемый в задаче алгоритм решения и доказать правильность его работы.</p>
<p><b>ОПК.2.1</b> Использует знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности для практического применения</p> <p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p> <p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p>	<p>Экзамен <b>Итоговое контрольное мероприятие</b></p>	<p>Использование алгоритмов перебора с возвратом. Знание принципов динамического программирования. Умение использовать основные динамические структуры данных (включая деревья). Умение выполнять балансировку деревьев. Знание основных алгоритмы работы с графами. Умение использовать библиотеку STL.</p>

### Спецификация мероприятий текущего контроля

**Структуры и объединения. Перебор с возвратом. Динамическое программирование. Динамические структуры данных.**

Продолжительность проведения мероприятия промежуточной аттестации: **4 часа**

Условия проведения мероприятия: **в часы самостоятельной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **30**

Проходной балл: **12.3**

<b>Показатели оценивания</b>	<b>Баллы</b>
------------------------------	--------------

Решение задач на структуры и объединения. За каждый не пройденный тест снимается от 0.2 до 0.5 балла	10
Решение задач на перебор с возвратом и динамическое программирование. За каждый не пройденный тест снимается от 0.2 до 0.5 балла	10
Решение задач на использование динамических структур данных. За каждый не пройденный тест снимается от 0.2 до 0.5 балла	10

### **Деревья. AVL-деревья. Графы. Алгоритмы на графах. Библиотека STL**

Продолжительность проведения мероприятия промежуточной аттестации: **4 часа**

Условия проведения мероприятия: **в часы самостоятельной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **30**

Проходной балл: **12.3**

<b>Показатели оценивания</b>	<b>Баллы</b>
Решение задач на графы. За каждый не пройденный тест снимается от 0.2 до 0.5 балла	10
Решение задач по использованию библиотеки STL. За каждый не пройденный тест снимается от 0.2 до 0.5 балла	10
Решение задач по бинарным деревьями их балансировке. За каждый не пройденный тест снимается от 0.2 до 0.5 балла	10

### **Индивидуальная работа**

Продолжительность проведения мероприятия промежуточной аттестации: **3 часа**

Условия проведения мероприятия: **в часы самостоятельной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **10**

Проходной балл: **4.1**

<b>Показатели оценивания</b>	<b>Баллы</b>
Выступление	2.5
Тестирование. За каждый не пройденный тест снимается от 0.1 до 0.5 балла	2.5
Отчет	2.5
Код программы	2.5

### **Экзамен**

Продолжительность проведения мероприятия промежуточной аттестации: **3 часа**

Условия проведения мероприятия: **в часы самостоятельной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **30**

Проходной балл: **12.3**

<b>Показатели оценивания</b>	<b>Баллы</b>
Теоретическая часть	15
Практическая часть	15