

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Лельчицкий Игорь Давыдович

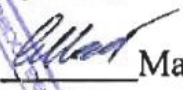
Должность: и.о. проректора по образовательной деятельности

Дата подписания: 30.09.2026 14:44:44

Уникальный программный ключ:

aa5b5ee17d97a2e4d94e98e995320af94f043ce2

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «ТВЕРСКОЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Утверждаю:
Руководитель ООП

Малышкина О.В.
«30»  2025 г.


Рабочая программа дисциплины (с аннотацией)

МЕТОДЫ ПРОГРАММИРОВАНИЯ

Специальность

10.05.01 Компьютерная безопасность

Специализация

Математические методы защиты информации

Для студентов 3, 4 курсов очной формы обучения

Уровень образования

СПЕЦИАЛИТЕТ

Составитель: к.ф.-м.н., доцент В.М Цирулева

Тверь, 2025

I. Аннотация

1. Цель и задачи дисциплины

Целью освоения дисциплины является теоретическая и практическая подготовка специалистов к деятельности, связанной с применением технологий программирования и с анализом вычислительной сложности алгоритмов для обеспечения информационной безопасности.

Задачами освоения дисциплины являются:

- 1) получение базовых знаний и умений, связанных с основными подходами к организации процесса разработки программного обеспечения;
- 2) получение теоретических сведений об основных методах оценки вычислительной сложности алгоритмов;
- 3) изучение базовых структур данных;
- 4) изучение основных алгоритмов сортировки и поиска в массивах;
- 5) изучение основных алгоритмов поиска в тексте;
- 6) изучение алгоритмов генерации перестановок, подмножеств и псевдослучайных последовательностей;
- 7) изучение основных алгоритмов на графах.

2. Место дисциплины в структуре ООП

Данная дисциплина входит в обязательную часть учебного плана, связана с другими дисциплинами образовательной программы. Её освоение базируется на знаниях, умениях и навыках, сформированных в процессе изучения дисциплин:

«Информатика» – работа с программными средствами общего назначения;

«Языки программирования» – знание одного из языков программирования высокого уровня;

«Математический анализ» – знание основных положений теории пределов функций, теории числовых и функциональных рядов;

«Теория вероятностей и математическая статистика» – основные понятия, виды распределений;

«Дискретная математика» – основные понятия и методы дискретной математики, включая дискретные функции, конечные автоматы, комбинаторный анализ и теорию графов.

Дисциплины, для которых освоение данной дисциплины необходимо как предшествующее: «Сети и системы передачи информации», «Криптографические методы защиты информации», «Компьютерные сети», «Системы управления базами данных», «Основы построения защищённых операционных систем», «Основы построения защищённых СУБД», «Основы построения защищённых сетей», «Защита программ и данных». Знания и практические навыки, полученные при изучении дисциплины «Методы программирования», используются студентами при разработке курсовых и дипломных работ.

Дисциплина изучается на 3 и 4 курсах (6 и 7 семестры).

3. Объем дисциплины:

6 семестр. 3 зачетные единицы, 108 академических часов, в том числе:

контактная аудиторная работа: лекции – 34 часов, в том числе практическая подготовка – 8 часов;

лабораторные занятия – 34 часов, в т.ч. практическая подготовка – 0 часов;

самостоятельная работа: 40 часов.

7 семестр. 5 зачетных единиц, 180 академических часов, в том числе:

контактная аудиторная работа: лекции – 34 часов, в том числе практическая подготовка – 0 часов;

лабораторные занятия – 51 часов, в т.ч. практическая подготовка – 0 часов;

самостоятельная работа: 68 часов,

контроль: 27 часов.

4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине (модулю)
ОПК-2. Способен применять программные средства системного и прикладного назначений, в том числе отечественного производства, для решения задач профессиональной деятельности	ОПК-2.1. Способен разрабатывать алгоритмы, реализующие современные математические методы защиты информации
ОПК-7. Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ОПК-7.1. Разрабатывает и применяет на языке высокого уровня алгоритмы решения типовых профессиональных задач
	ОПК-7.2. Применяет известные методы программирования и возможности базового языка программирования для решения типовых профессиональных задач
	ОПК-7.3. Использует основные принципы разработки, документирования, тестирования и отладки программ
ОПК-12. Способен администрировать операционные системы и выполнять работы по восстановлению работоспособности прикладного и системного программного обеспечения	ОПК-12.2. Использует методы системного программирования
ОПК-13. Способен разрабатывать компоненты программных и программно-аппаратных средств защиты	ОПК-13.2. Работает с интегрированными средами разработки программного обеспечения

информации в компьютерных системах и проводить анализ их безопасности	ОПК-13.3. Оценивает качество готового программного обеспечения
---	---

6. Форма промежуточной аттестации

Зачет в 6 семестре.

Экзамен в 7 семестре.

7. Язык преподавания русский.

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

Очная форма обучения

№п/п	Учебная программа – наименование разделов и тем	Всего (час)	Контактная работа (час)			Самостоятельная работа, в том числе Контроль (час)
			Лекции	Лабораторные занятия		
				всего	в т.ч. практическая подготовка	
1.	Тема 1. Жизненный цикл программного обеспечения.	6	2			4
2.	Тема 2. Качество программных систем	6	2			4
3.	Тема 3. Анализ и разработка требований	6	2			4
4.	Тема 4. Проектирование архитектуры программных систем	10	2	4		4
5.	Тема 5. Аттестация и	10	2	4		4

№п/ п	Учебная программа – наименование разделов и тем	Всего (час)	Контактная работа (час)			Самостоятель ная работа, в том числе Контроль (час)
			Лекции	Лабораторные занятия		
				всего	в т.ч. практич еская подгото вка	
	верификация					
6.	Тема 6. Управление проектами	6	2			4
7.	Тема 7. Базовые структуры данных	14	4	6		4
8.	Тема 8. Понятие алгоритма. Оценки сложности работы алгоритмов	14	4	6		4
9.	Тема 9. Деревья	20	8	8		4
10.	Тема 10. Алгоритмы поиска	16	6	6		4
	ИТОГО в 6 семестре	108	34	34		40
11.	Тема 11. Алгоритмы сортировки. Внутренние сортировки	31	6	9		12+4
12	Тема 12. Алгоритмы сортировки. Внешние сортировки	22	4	6		8+4

№п/п	Учебная программа – наименование разделов и тем	Всего (час)	Контактная работа (час)			Самостоятельная работа, в том числе Контроль (час)
			Лекции	Лабораторные занятия		
				всего	в т.ч. практическая подготовка	
13	Тема 13. Алгоритмы поиска. Хеширование	31	6	9		12+4
14	Тема 14. Алгоритмы поиска подстрок	22	4	6		8+4
15	Тема 15. Алгоритмы на графах	49	10	15		20+4
16	Тема 16. Алгоритмы генерации перестановок подмножеств сочетаний и разбиений.	13	2	3		4+4
17	Тема 17. Генерация псевдослучайных чисел	12	2	3		4+3
	ИТОГО в 7 семестре	180	34	51		68+27

III. Образовательные технологии

Учебная программа – наименование разделов и тем	Вид занятия	Образовательные технологии
---	-------------	----------------------------

Тема 1. Жизненный цикл программного обеспечения.	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, технология развития креативного мышления
Тема 2. Качество программных систем	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, технология развития креативного мышления
Тема 3. Анализ и разработка требований	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, технология развития креативного мышления.
Тема 4. Проектирование архитектуры программных систем	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, технология развития креативного мышления
Тема 5. Аттестация и верификация	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, технология развития креативного мышления

Тема 6. Управление проектами	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, технология развития креативного мышления
Тема 7. Базовые структуры данных	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 8. Понятие алгоритма. Оценки сложности работы алгоритмов	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 9. Деревья	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 10. Алгоритмы поиска	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.

Тема 11. Алгоритмы сортировки. Внутренние сортировки	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 12. Алгоритмы сортировки. Внешние сортировки	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 13. Алгоритмы поиска. Хеширование	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 14. Алгоритмы поиска подстрок	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 15. Алгоритмы на графах	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.

Тема 16. Алгоритмы генерации перестановок подмножеств сочетаний и разбиений.	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.
Тема 17. Генерация псевдослучайных чисел	лекция лабораторные	Дискуссионные технологии, дистанционные образовательные технологии, проблемная лекция, кейс-технология, методы группового решения творческих задач.

IV. Оценочные материалы для проведения текущей и промежуточной аттестации

Оценочные материалы для проведения текущей аттестации

Задания для лабораторных (семинарских) занятий

Тема 1.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Жизненный цикл ПО. Стандартизация жизненного цикла в системе государственных стандартов ЕСПД и ГОСТ Р ИСО/МЭК 12207.

Задание 2 (ОПК-7.3, ОПК-13.3): Реферат. Модели жизненного цикла (каскадная, спиральная, формальные и др.). Гибкие методологии разработки (XP, Scrum и др.).

Тема 2.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Критерии оценки качества программных систем, характеристики качества и метрики качества. Методы контроля качества.

Задание 2 (ОПК-7.3, ОПК-13.3): Реферат. Модель процесса оценивания качества. ГОСТ Р ИСО 9000 и ГОСТ Р ИСО 9126. Инструментальные системы оценки качества программных систем.

Тема 3.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Анализ и разработка требований. Функциональные и нефункциональные требования. Разработка требований. Спецификация требований к программному обеспечению, техническое задание (ГОСТ 19.201-78, ГОСТ 34.602-89). Методы первичного сбора требований. Модели системы.

Задание 2 (ОПК-7.3, ОПК-13.3): Реферат. Языки спецификации требований (сценарии, PDL, потоковые диаграммы и др.). Оценка требований (ГОСТ Р ИСО/МЭК 12207). Методы аттестации требований (обзор, протипирование, тестовые сценарии). Управление требованиями. Инструментальные средства отслеживания и контроля требований.

Тема 4.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Проектирование архитектуры программных систем. Архитектурное и детальное проектирование. Основные принципы проектирования. Сквозная функциональность. Сцепление и связность. Архитектурные стили (парадигмы: объектно-ориентированная, компонентная, многослойная, DDD, клиент-сервер, многоуровневая, SOA и др.).

Задание 2 (ОПК-7.3, ОПК-13.3): Реферат. Принципы объектно-ориентированного проектирования (SOLID). Паттерны проектирования. Унифицированный язык моделирования (UML). Кодогенерация. Документирование архитектуры.

Тема 5.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Аттестация и верификация. Понятие аттестации и верификации. Методы инспектирования. Инструментальные средства инспектирования и оценки качества. Тестирование ПО. Неразрешимость проблемы тестирования. Виды и уровни тестирования.

Задание 2 (ОПК-7.3, ОПК-13.3): Реферат. Стратегии восходящего и нисходящего тестирования. Методы "белого" и "черного" ящика. Автоматизированное и ручное тестирование. Разработка через тестирование

(TDD). Непрерывная интеграция. Покрытие кода тестами. Методы отладки программ. Инструментальные средства поддержки тестирования и отладки.

Тема 6.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Управление проектами. Группы процессов и процедуры управления проектом. План управления проектом. Инструментальные средства управления проектами.

Задание 1 (ОПК-7.3, ОПК-13.3): Реферат. Управление персоналом. Права и обязанности членов коллектива. Мотивация. Организация совместной работы коллектива. Управление рисками. Модель зрелости возможностей. Инструментальные средства планирования и управления процессом разработки.

Тема 7.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): реализовать класс стёк на базе связной динамической структуры. Структура записи: целочисленное поле данных (или шаблон типа), один указатель на следующий элемент. Хранится указатель на вершину стека. Основные методы, реализуемые в классе: добавление элемента в начало; удаление элемента из начала; получение значения из вершины стека; проверка стека на пустоту; уничтожение (очистка) стека. Вспомогательные методы, реализуемые в классе: получение указателя на первый элемент, получение указателя на следующий элемент, получение значения поля данных по значению указателя на элемент стека. Дополнительные операции, которые реализуются в главном модуле: поиск элемента по значению поля данных (возвращает порядковый номер элемента в стеке, если элемент найден, и -1 в противном случае); просмотр стека (вывод на консоль); случайное заполнение стека (параметры: число элементов, начало диапазона, конец диапазона); запись стека в файл (параметр: физическое имя файла), чтение стека из файла (параметр: физическое имя файла). Просмотр стека в главном модуле осуществляется с помощью методов класса (итераторов): получение указателя на первый элемент, получение указателя на следующий элемент, получение значения поля данных по значению указателя на элемент стека. В главном модуле или отдельном файле предусмотреть выполнение нескольких тестов, проверяющих корректность выполнения основных операций. Тестирование работы алгоритма.

Задание 2 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Реализовать класс Очередь на базе связной динамической структуры. Структура записи: целочисленное поле данных (или шаблон типа), один указатель на следующий элемент. Хранятся указатели на первый элемент очереди и на последний элемент очереди. Основные методы, реализуемые в классе:

добавление элемента в конец; удаление элемента из начала; проверка очереди на пустоту; уничтожение очереди. Вспомогательные методы, реализуемые в классе: получение указателя на первый элемент, получение указателя на следующий элемент, получение значения поля данных по значению указателя на запись. Дополнительные операции, которые реализуются в главном модуле: поиск элемента по значению поля данных (возвращает порядковый номер элемента в очереди, если элемент найден, и -1 в противном случае); просмотр очереди; случайное заполнение очереди (параметры: число элементов, начало диапазона, конец диапазона); сохранение очереди в файле (параметр: физическое имя файла); чтение очереди из файла (параметр: физическое имя файла). Просмотр очереди осуществляется в главном модуле с помощью методов класса (итераторов): получение указателя на первый элемент, получение указателя на следующий элемент, получение значения поля данных по значению указателя на запись. Тестирование работы алгоритма.

Тема 8.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): В рекурсивном алгоритме, в котором при каждом рекурсивном вызове производится $2n$ операций и параметр n (размер задачи) уменьшается на единицу, возникает следующее рекуррентное соотношение для числа выполняемых операций: $C(n) = C(n-1) + 2n$, где $n > 1$ и $C(1) = 1$. Постепенно раскрывая рекурсию, определите временную сложность данного алгоритма.

Задание 2 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): В рекурсивном алгоритме, в котором при каждом рекурсивном вызове производится 1 операция и параметр n (размер задачи) уменьшается на единицу, возникает следующее рекуррентное соотношение для числа выполняемых операций: $C(n) = 2C(n-1) + 1$, где $n > 1$ и $C(1) = 1$. Постепенно раскрывая рекурсию, определите временную сложность данного алгоритма.

Тема № 9.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Реализовать класс Бинарное дерево поиска (упорядоченное дерево) на базе связной динамической структуры. Структура записи: целочисленное поле данных (ключ), указатели на левый и правый дочерние элементы. Хранится указатель на корень дерева. Основные операции: вставка узла (два способа: рекурсивный и не рекурсивный); удаление узла; поиск элемента по значению ключа (два способа: рекурсивный и не рекурсивный); вычисление высоты дерева; прямой обход дерева (два способа: рекурсивный и не рекурсивный – с помощью стека); симметричный обход дерева; обратный обход дерева; обход дерева по ширине (с помощью очереди); уничтожение дерева; проверка дерева на пустоту. Вывод линейных списков, соответствующих всем обходам дерева, осуществлять в одном пункте меню в главном модуле.

Дополнительные операции, реализуемые в главном модуле: формирование дерева из линейного списка (из файла), вывод всех обходов дерева в файл. Тестирование работы алгоритма.

Задание 2 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Реализовать класс AVL-дерево на базе связной динамической структуры. Структура записи: целочисленное поле данных (ключ), указатели на левый и правый дочерние элементы, высота узла. Хранится указатель на корень дерева. Основные операции: вставка узла; удаление узла; поиск элемента по значению ключа; вычисление высоты дерева; прямой обход дерева; симметричный обход дерева; обратный обход дерева; обход дерева по ширине; уничтожение дерева; проверка дерева на пустоту; вычисление баланса. Вспомогательные методы класса, описываемые в секции private: левое вращение; правое вращение; балансировка; больше левое вращение (можно реализовать через два вращения); большое правое вращение (можно реализовать через два вращения). Вывод линейных списков, соответствующих всем обходам дерева (осуществлять в одном пункте меню). Дополнительные операции: формирование дерева из линейного списка (из файла), вывод всех обходов в файл. Тестирование работы алгоритма.

Тема 10.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Реализовать класс Список с потоками на базе связной динамической структуры. Список содержит четыре потока: поток, упорядоченный по возрастанию фамилий; поток, упорядоченный по убыванию фамилий; поток, упорядоченный по возрастанию номеров телефонов; поток, упорядоченный по убыванию номеров телефонов. Хранятся: единая метка начала – общая для всех потоков (указатель), содержащая данные, меньшие всех возможных значений для потоков, упорядоченных по возрастанию и данные, большие всех возможных значений для потоков, упорядоченных по убыванию; единая метка конца – общая для всех потоков (указатель), содержащая данные, большие всех возможных значений для потоков, упорядоченных по возрастанию и данные, меньшие всех возможных значений для потоков, упорядоченных по убыванию. Наличие одной общей метки начала и одной общей метки конца позволяет вставлять записи единообразно в позиции без нарушения упорядочения, не рассматривая отдельно случаи вставки в пустой список, вставки в начало, вставки в конец и вставки в промежуток. Аналогично – при удалении записей. Структура записи: поля данных (фамилия и номер телефона), указатели на следующий элемент по фамилии и предыдущий элемент по фамилии, указатели на следующий элемент по телефону и предыдущий элемент по телефону. Основные операции: добавление элемента без нарушения упорядочения потоков; удаление элемента по номеру телефона без нарушения упорядочения потоков; удаление элемента по фамилии без нарушения упорядочения потоков; поиск телефона по фамилии; поиск фамилии по телефону; уничтожение списка с потоками; проверка списка с потоками на пустоту. Просмотр списка осуществляется в главном

модуле с помощью методов класса (итераторов) по всем потокам; данные выводятся во всех потоках попарно (фамилия, номер телефона). Вывод списков, соответствующих всем потокам, осуществлять в одном пункте меню. Тестирование работы алгоритма.

Задание 2 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Реализовать класс красно-черное дерево на базе связанной динамической структуры. Структура записи: целочисленное поле данных (ключ), указатели на левый и правый дочерние элементы, цвет узла, указатель на родительский узел. Хранится указатель на корень дерева. Основные операции: вставка узла; удаление узла; поиск элемента по значению ключа; вычисление высоты дерева; вычисление черной высоты дерева; прямой обход дерева; симметричный обход дерева; обратный обход дерева; обход дерева по ширине; уничтожение дерева; проверка дерева на пустоту. Вспомогательные методы класса, описываемые в секции private: все случаи балансировок для вставки и удаления. Вывод линейных списков, соответствующих всем обходам дерева (осуществлять в одном пункте меню). Дополнительные операции: формирование дерева из линейного списка (из файла), вывод всех обходов дерева в файл. Тестирование работы алгоритма.

Тема 11.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Сортировка массивов.

- Базовые сортировки:
 - Сортировка выбором
 - Сортировка вставкой
 - Сортировка пузырьком
- Сортировки, основанные на базовых
 - Сортировка перемешиванием (Шейкерная сортировка)
 - Сортировка методом Шелла
 - Сортировка методом прочесывания

Для реализации алгоритмов тем Сортировка массивов и Поиск в массивах создать класс, в котором описывается структура динамический массив (тип элемента массива целочисленный или задается с помощью шаблона) и используются методы:

1. конструктор,
2. деструктор,
3. установка размерности массива и выделение памяти,
4. получение размерности массива,
5. задание элемента массива,
6. получение элемента массива,
7. задание элементов массива случайным образом,
8. методы сортировок,
9. методы поиска.

В главном модуле требуется получать время выполнения алгоритма сортировки или поиска, осуществлять вывод на консоль и в файл.

Создать отчет (WORD) со сравнительными результатами вычислений, оформленными в виде таблицы со столбцами:

- тип сортировки,
- название сортировки,
- число элементов массива,
- характеристика массива (имеет одинаковые элементы, почти упорядочен и т.д.).

Задание 2 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Сортировка массивов.

- Быстрые сортировки
 - Быстрая сортировка
 - Сортировка слиянием
 - Пирамидальная сортировка
- Целочисленные сортировки
 - Сортировка подсчетом
 - Блочная сортировка (можно и с действительными числами)
 - Распределяющая сортировка
 - Битовая сортировка
 - Бинарная сортировка

Для реализации алгоритмов тем Сортировка массивов и Поиск в массивах создать класс, в котором описывается структура динамический массив (тип элемента массива целочисленный или задается с помощью шаблона) и используются методы:

- 10.конструктор,
- 11.деструктор,
- 12.установка размерности массива и выделение памяти,
- 13.получение размерности массива,
- 14.задание элемента массива,
- 15.получение элемента массива,
- 16.задание элементов массива случайным образом,
- 17.методы сортировок,
- 18.методы поиска.

В главном модуле требуется получать время выполнения алгоритма сортировки или поиска, осуществлять вывод на консоль и в файл.

Создать отчет (WORD) со сравнительными результатами вычислений, оформленными в виде таблицы со столбцами:

- тип сортировки,
- название сортировки,
- число элементов массива,
- характеристика массива (имеет одинаковые элементы, почти упорядочен и т.д.).

Тема 12.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Методы для программной реализации

- Естественная сортировка (метод естественного слияния)

- Сортировка методом двухпутевого сбалансированного слияния

Алгоритмы по теме Внешние сортировки можно реализовать различными способами: написать функции, создать класс, создать библиотеку функций. В главном модуле требуется получать время выполнения алгоритма сортировки, осуществлять ввод с консоли и из файла, осуществлять вывод на консоль и в файл.

Задание 2 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Методы для программной реализации Сортировка методом n-путевого слияния

- Многофазная сортировка (Фибоначчиевая)
- Каскадное слияние

Алгоритмы по теме Внешние сортировки можно реализовать различными способами: написать функции, создать класс, создать библиотеку функций. В главном модуле требуется получать время выполнения алгоритма сортировки, осуществлять ввод с консоли и из файла, осуществлять вывод на консоль и в файл.

Тема 13.

Задание 1 (ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Поиск в массивах

- Программная реализация
 - Линейный, последовательный поиск, исчерпывающий
 - Поиск с барьером
 - Бинарный (двоичный, дихотомический) поиск
 - Интерполяционный поиск
 - Следящий поиск
 - Двоичный следящий поиск
 - Интерполяционный следящий поиск.

Задание 2 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Методы для программной реализации

- Метод цепочек (внешнее или открытое хеширование)
- Метод открытой адресации (закрытое хеширование)

Алгоритмы по теме Хеширование можно реализовать различными способами:

- написать функции,
- создать библиотеку функций.
- Вывод: значение хеш функции, значение элемента массива.

Тема 14.

Задание 1 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Метод для программной реализации

- Алгоритм Кнута, Мориса и Пратта

Задание 2 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2): Метод для программной реализации

- Алгоритм Боуера и Мура

Алгоритмы по теме Поиск в тексте можно реализовать различными способами: написать функции, создать библиотеку функций.

В главном модуле требуется получать время выполнения алгоритма, осуществлять ввод с консоли и из файла, осуществлять вывод на консоль и в файл.

Тема 15

Задание 1 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2):

Обходы графа

- теория (определения)
- поиск в глубину
- поиск в ширину

Нахождение кратчайшего пути

- теория (определения)
- алгоритм Дейкстры
- алгоритм Флойда

Задание 2 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2):

- Остовные деревья минимального веса
 - теория (определения)
 - алгоритм Прима
 - алгоритм Крускала
- Проблема связности
 - в неориентированных графах
 - в ориентированных графах

Для реализации алгоритмов по теме Алгоритмы на графах создать класс, в котором описывается структура двумерный динамический массив (тип элемента массива целочисленный или задается с помощью шаблона) и используются методы:

1. конструктор,
2. деструктор,
3. установка размерности массива и выделение памяти,
4. получение размерности массива,
5. задание элемента массива,
6. получение элемента массива,
7. задание элементов массива случайным образом,
8. методы работы с графами,
9. методы преобразования одного представления графа в другое.

В главном модуле требуется получать время выполнения алгоритма, осуществлять вывод на консоль и в файл.

Ввод и вывод графа реализовывать построчно в виде матрицы смежности с элементами вида $(i, j, a(i,j))$.

Вывод обходов и путей в графе осуществлять в виде: $..., (i \rightarrow j, a(i,j)), ...$. И отдельно длина кратчайшего пути или минимальный вес.

Тема 16.

Задание 1 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2):

- Алгоритм порождения перестановок

- теория (определения)
- в лексикографическом порядке
- в порядке минимального изменения

Алгоритмы по темам Алгоритмы порождения перестановок можно реализовать различными способами:

написать функции,

создать библиотеку функций.

Требуется реализовать хотя бы один метод построения перестановок.

Вывод перестановки требуется осуществлять в виде: номер перестановки, перестановка (в столбик).

Задание 2 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2):

- Коды Грея

- теория (определения)
- алгоритм рекуррентного построения кода Грея
- не рекуррентная версия алгоритма генерирования кода Грея

Алгоритмы по темам Коды Грея можно реализовать различными способами:

написать функции,

создать библиотеку функций.

Требуется реализовать хотя бы один метод построения кода Грея.

Выводить код Грея требуется в виде: номер вектора, вектор (в столбик).

Тема 17.

Задание 1 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2):

Генераторы случайных чисел

- метод середины квадрата
- линейный конгруэнтный метод.

Алгоритмы по теме Генерация псевдослучайных последовательностей можно реализовать различными способами:

написать функции,

создать класс,

создать библиотеку функций.

Требуется реализовать как минимум два различных метода генерации

последовательности ПСЧ. Параметры методов должны выбираться

оптимальным способом. В главном модуле требуется получать время

выполнения алгоритма. При выводе последовательности ПСЧ выводить

также параметры метода и период последовательности. Приветствуется

программное определение периода последовательности.

Задание 2 (ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-12.2, ОПК-13.2):

Генераторы случайных чисел

- квадратичный конгруэнтный генератор
- метод Фибоначчи с запаздываниями

Алгоритмы по теме Генерация псевдослучайных последовательностей можно реализовать различными способами:

написать функции,

создать класс,

создать библиотеку функций.

Требуется реализовать как минимум два различных метода генерации последовательности ПСЧ. Параметры методов должны выбираться оптимальным способом. В главном модуле требуется получать время выполнения алгоритма. При выводе последовательности ПСЧ выводить также параметры метода и период последовательности. Приветствуется программное определение периода последовательности.

Оценочные материалы для проведения промежуточной аттестации

Проверяемые индикаторы достижения компетенций: ОПК-2.1, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-12.2, ОПК-13.2, ОПК-13.3.

Каждый студент решает индивидуальные задания и отвечает на теоретический вопрос.

Требования к зачету

I. Реферат по одной из тем раздела Технологии программирования

Тема 1. Жизненный цикл программного обеспечения.

1. Программные системы. Основные проблемы разработки сложных программных систем. Жизненный цикл ПО. Стандартизация жизненного цикла в системе государственных стандартов ЕСПД и ГОСТ Р ИСО/МЭК 12207.
2. Модели жизненного цикла (каскадная, спиральная, формальные и др.). Гибкие методологии разработки (XP, Scrum и др.).

Тема 2. Качество программных систем

3. Критерии оценки качества программных систем, характеристики качества и метрики качества. Методы контроля качества.
4. Модель процесса оценивания качества. ГОСТ Р ИСО 9000 и ГОСТ Р ИСО 9126. Инструментальные системы оценки качества программных систем.

Тема 3. Анализ и разработка требований

5. Функциональные и нефункциональные требования. Разработка требований. Спецификация требований к программному обеспечению, техническое задание (ГОСТ 19.201-78, ГОСТ 34.602-89).
6. Методы первичного сбора требований. Модели системы. Языки спецификации требований (сценарии, PDL, потоковые диаграммы и др.). Оценка требований (ГОСТ Р ИСО/МЭК 12207).

7. Методы аттестации требований (обзор, протипирование, тестовые сценарии). Управление требованиями. Инструментальные средства отслеживания и контроля требований.

Тема 4. Проектирование архитектуры программных систем

8. Архитектурное и детальное проектирование. Основные принципы проектирования. Сквозная функциональность. Сцепление и связность.
9. Архитектурные стили (парадигмы: объектно-ориентированная, компонентная, многослойная,DDD, клиент-сервер, многоуровневая, SOA и др.).
10. Принципы объектно-ориентированного проектирования (SOLID).
11. Паттерны проектирования.
12. Унифицированный язык моделирования (UML).
13. Унифицированный язык моделирования (UML).
14. Кодогенерация. Документирование архитектуры.

Тема 5. Аттестация и верификация

15. Понятие аттестации и верификации. Методы инспектирования. Инструментальные средства инспектирования и оценки качества.
16. Тестирование ПО. Неразрешимость проблемы тестирования. Виды и уровни тестирования. Стратегии восходящего и нисходящего тестирования. Методы "белого" и "черного" ящика. Автоматизированное и ручное тестирование. Разработка через тестирование (TDD). Непрерывная интеграция. Покрытие кода тестами.
17. Методы отладки программ. Инструментальные средства поддержки тестирования и отладки.

Тема 6. Управление проектами

18. Группы процессов и процедуры управления проектом. План управления проектом. Инструментальные средства управления проектами.

19. Управление персоналом. Права и обязанности членов коллектива. Мотивация. Организация совместной работы коллектива.
20. Управление рисками. Модель зрелости возможностей. Инструментальные средства планирования и управления процессом разработки

II. Семестровые задания на лабораторных занятиях

Перечень тем лабораторных занятий

1. Стеки, деки, очереди, линейные списки.
2. Двухнаправленные списки.
3. Потоки.
4. Разреженные матрицы.
5. Деревья. Двоичные деревья поиска.
6. Алгоритмы поиска в деревьях.
7. AVL-деревья.
8. Красно-черные деревья.
9. Алгоритмы сортировки сравнениями. Вывод нижней оценки для трудоемкости работы алгоритмов данного типа. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.
10. Алгоритм сортировки вставками и вывод оценки его трудоемкости. Сортировка Шелла. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.
11. Обменная сортировка. Сортировка расческой. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.
12. Алгоритм быстрой сортировки. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.
13. Сортировка выбором. Пирамидальная сортировка. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.
14. Сортировки слиянием, блочная и распределяющие сортировки. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.

15. Целочисленные сортировки. Сортировка подсчетом и поразрядная сортировки. Сравнение эмпирических оценок вычислительной сложности работы алгоритма с теоретическими.
16. Внешние сортировки слиянием.
17. Поиск в массивах. Алгоритмы исчерпывающего поиска. Бинарный, интерполяционный и следящий поиск.
18. Поиск в тексте. Алгоритмы Кнута - Морриса - Пратта и Боуэра - Мура.
19. Хеширование и способы разрешения коллизий.
20. Способы задания графа. Остовное дерево. Алгоритмы поиска в глубину и в ширину.
21. Поиск кратчайшего пути в графе. Алгоритмы Дейкстры и Флойда.
22. Алгоритм построения минимального остовного дерева. Алгоритмы Прима и Крускала.
23. Нахождение связных и двусвязных компонент графа.
24. Алгоритм поиска сильно связных компонент графа.
25. Генерация псевдо - случайных чисел. Метод середины квадрата, линейный и квадратичный конгруэнтные методы, метод Фибоначчи
26. Алгоритмы порождения перестановок в лексикографическом порядке и порядке минимального изменения.
27. Построение кода Грея. Рекуррентный и не рекуррентный алгоритмы.

III. Перечень тем индивидуальных заданий

1. Сложение деревьев поиска.
2. Обменная сортировка со слиянием, алгоритм Бэтчера.
3. Лексикографическая сортировка.
4. Турнирная сортировка.
5. Алгоритмы поиска с возвращениями.
6. Индексно-последовательный поиск.
7. Алгоритм поиска подстрок с помощью построения конечного автомата.
8. Алгоритм Рабина-Карпа поиска подстрок.
9. Построение оптимальных бинарных деревьев поиска.

- 10.Балансировка деревьев по весу.
- 11.Цифровой поиск.
- 12.Алгоритм транзитивного замыкания.
- 13.Композиции и разбиения целых чисел и алгоритмы их порождения.
- 14.Алгоритм Беллмана-Форда нахождения кратчайших расстояний между вершинами.
- 15.Потоки в сетях.
- 16.Алгоритм Эдмондса-Карпа поиска максимального потока в сети.

Вид и способ проведения промежуточной аттестации: индивидуальный устный опрос сочетается с самостоятельной практической работой студента.

Критерии оценивания и шкала оценивания:

Максимально возможное количество баллов – 3 балла. Для получения зачета необходимо выполнить задачу и ответить на теоретический вопрос с суммарной оценкой не менее 2-х баллов.

3 балла:

Ответ на вопрос демонстрирует знание и корректное использование терминологии. Факты и примеры в полном объеме обосновывают выводы. Имеется полное верное решение задачи, включающее правильный ответ.

2 балла:

Ответ на вопрос демонстрирует знание и корректное использование терминологии. Ответ не содержит фактических ошибок. Дано верное решение задачи, но в решении имеются неверные записи И/ИЛИ арифметические ошибки.

1 балл:

Ответ демонстрирует знание и корректное использование терминологии. Решение содержит фактические ошибки, не искажающие общего смысла.

0 баллов:

В ответе преобладают рассуждения общего характера И/ИЛИ содержит существенные фактические ошибки, искажающие смысл. Решение не дано ИЛИ дано неверное решение.

3. Вопросы к зачету

1. Стеки, деки, очереди, линейные списки.

2. Двухнаправленные списки.
3. Потоки.
4. Разреженные матрицы.
5. Деревья. Двоичные деревья поиска.
6. Алгоритмы поиска в деревьях.
7. AVL-деревья.
8. Красно-черные деревья.
9. Программное обеспечение, стадии жизненного цикла: понятие жизненного цикла.
10. Программное обеспечение, стадии жизненного цикла: водопадная (каскадная) модель.
11. Программное обеспечение, стадии жизненного цикла: итерационная модель.
12. Программное обеспечение, стадии жизненного цикла: итерационный подход, прототипирование.
13. Анализ требований: техническое задание и спецификация требований, три подхода к разработке ТЗ.
14. Анализ требований: свойства требований. Анализ требований: способы выражения (записи) требований в ТЗ.
15. Анализ требований: требования заказчика и детальные требования.
16. Анализ требований: способы организации детальных требований.
17. Характеристики качества ПО: удобство применения, универсальность.
18. Характеристики качества ПО: надежность, эффективность.
19. Характеристики качества ПО: сопровождаемость, корректность.
20. Обеспечение качества ПО: обеспечение завершенности и надежности.
21. Обеспечение качества ПО: обеспечение эффективности и легкости применения.
22. Обеспечение качества ПО: обеспечение универсальности.
23. Проектирование: декомпозиция, классификация архитектур.

24.Проектирование: архитектура потоков данных и независимые компоненты.

25.Проектирование: декомпозиция, классификация архитектур.

4.В экзаменационный билет входит два теоретических вопроса и две задачи.

Вопросы к экзамену

1. Определение сложности алгоритма. Общие функции оценки сложности. Правила вычисления сложности алгоритма. Анализ скорости выполнения алгоритмов.
2. Списки, их представление, операции со списками. Списки с потоками.
3. Стеки и очереди. Основные операции. Очереди с приоритетами. Основные операции.
4. Разреженные матрицы, способы представления. Основные операции.
5. Деревья. Полные деревья. Обход бинарного дерева: прямой, симметричный, обратный, в ширину.
6. Бинарные деревья поиска. Основные операции. Алгоритм сортировки с помощью дерева.
7. AVL-деревья. Балансировка деревьев по высоте.
8. Красно-черные деревья. Балансировка деревьев по высоте.
9. Сортировки вставками. Сортировка Шелла. Их вычислительная сложность.
- 10.Обменные сортировки. Шейкерная сортировка. Сортировка прочесыванием. Их вычислительная сложность.
- 11.Сортировка выбором. Пирамидальная сортировка. Их вычислительная сложность
- 12.Сортировки, не основанные на сравнениях. Блочная сортировка. Поразрядная сортировка. Сортировка подсчетом. Их вычислительная сложность.
- 13.Быстрая сортировка. Сортировки слиянием. Их вычислительная сложность.

14. Основные понятия внешней сортировки. Алгоритмы естественного и сбалансированного слияния.
15. Внешние сортировки. Алгоритмы многофазного и каскадного слияния.
16. Поиск в массивах: полный перебор, двоичный и интерполяционный поиск. Следящий поиск.
17. Поиск в тексте. Алгоритмы Кнута – Морриса – Пратта и Боуэра – Мура.
18. Хеш-функция. Виды хеш-функций. Хеш-таблица. Внешнее хеширование (метод цепочек). Блочное хеширование.
19. Закрытое хеширование. Способы разрешения коллизий.
20. Способы представления графов в ЭВМ. Алгоритм поиска в глубину.
Алгоритм поиска в ширину. Реализации алгоритмов.
21. Нахождения транзитивного замыкания в графе.
22. Алгоритм Дейкстры нахождения кратчайших расстояний от источника до всех остальных вершин графа.
23. Алгоритм Флойда нахождения кратчайших расстояний между всеми парами вершин графа.
24. Остовные деревья. Алгоритмы нахождения остовного дерева минимального веса Прима и Крускала.
25. Связные компоненты в неориентированных графах. Алгоритм, основанный на разделенных множествах. Двусвязные компоненты в неориентированных графах. Алгоритм, основанный на поиске точек сочленения.
26. Сильно связные компоненты в ориентированных графах. Алгоритм Косарайю.
27. Перестановки. Алгоритмы порождения перестановок в лексикографическом порядке и в порядке минимального изменения.
28. Коды Грея. Алгоритмы порождения кодов Грея: рекурсивный и не рекурсивный.

29. Источники и генераторы случайных и псевдослучайных чисел.

Алгоритмы генерации псевдослучайных последовательностей. Проверка качества работы генератора.

30. Методы моделирования равномерно распределенных случайных величин.

Моделирование дискретных и непрерывных случайных величин.

5. Пример одной из модульных контрольных работ

1. Опишите необходимые типы, переменные и функцию вставки в связанный линейный однонаправленный список (с полем данных типа `integer`) нового элемента с адресом `P1` после элемента с адресом `P` (предполагается, что элемент с адресом `P` в списке существует). Не используйте никаких других дополнительных переменных-указателей.

2. Опишите необходимые типы, переменные и последовательность операций присваивания, формирующих связанный список с двумя потоками, упорядочивающими элементы списка по возрастанию значений поля данных `поле1` и по возрастанию значений поля данных `поле2` (`поле1` – типа `char`, `поле2` – типа `integer`). Входные данные содержат три набора (значение поля1, значение поля2, указатель на выделенный под элемент участок памяти): ('f', 5, p1), ('w', 1, p2), ('m', 10, p3). Изобразите список графически.

3. В рекурсивном алгоритме, в котором при каждом рекурсивном вызове производится $2n$ операций и параметр n (размер задачи) уменьшается на единицу, возникает следующее рекуррентное соотношение для числа выполняемых операций:

$C(n) = C(n-1) + 2n$, где $n > 1$ и $C(1) = 1$.

Постепенно раскрывая рекурсию, определите временную сложность данного алгоритма.

4. Дайте определение глубины узла и глубины дерева. Приведите пример.

5. Двоичное дерево задано как статическая структура данных в виде одномерного массива:

| 1 | 2 | 3 | 4 | 5 | 6 | X | 7 | X | X | X | 8 | 9 | X | X |, где символ X означает отсутствие значения.

Изобразите его графически: с помощью списочного представления (списков дочерних узлов) и с помощью метода полных узлов (в виде графа с выделенной вершиной – корнем).

6. Выпишите линейные последовательности, получающиеся в результате прямого, симметричного и обратного обхода дерева в задании 6, а также его обхода по ширине.

7. Даны два списка s_1 и s_2 , организованных в виде стека и упорядоченных по возрастанию значений информационных полей. Создать упорядоченный по возрастанию список в виде стека s_3 , являющийся объединением списков s_1 и s_2 , т.е. $s_3 = s_1 \cup s_2$. Напишите алгоритм или программный код.

V. Учебно-методическое и информационное обеспечение дисциплины

1) Рекомендуемая литература

а) обязательная литература:

1. Методы программирования : учебное пособие / Ю. Ю. Громов, О. Г. Иванова, Ю. В. Кулаков [и др.]. — Тамбов : Тамбовский государственный технический университет, ЭБС АСВ, 2012. — 144 с. — ISBN 978-5-8265-1076-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/63867.html>

2. Ковалевская, Е. В. Методы программирования : учебное пособие / Е. В. Ковалевская, Н. В. Комлева. — Москва : Евразийский открытый институт, 2011. — 320 с. — ISBN 978-5-374-00356-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/10784.html>

а) дополнительная литература:

1. Макарова, Н. П. Методы программирования и информатика : лабораторный практикум : в 2 частях : [16+] / Н. П. Макарова. — Гродно : Гродненский Государственный университет имени Янки Купалы, 2012. — Часть 1. — 54 с. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=134239>

2) Программное обеспечение

Adobe Acrobat Reader DC - Russian	бесплатно Государственный контракт на поставку лицензионных программных продуктов
Cadence SPB/OrCAD 16.6	103 - ГК/09 от 15.06.2009
Google Chrome	бесплатно
Java SE Development Kit 8 Update 45 (64-bit)	бесплатно
Kaspersky Endpoint Security 10 для Windows	Акт на передачу прав ПК545 от 16.12.2022
Lazarus 1.4.0	бесплатно Акт предоставления прав ИС00000027 от 16.09.2011;
Mathcad 15 M010	Акт предоставления прав № Us000311 от 25.09.2012;
MATLAB R2012b	бесплатно
Mercurial 3.7.3	бесплатно
Microsoft SQL Server 2012 Express LocalDB	бесплатно
Microsoft Web Deploy 3.5	бесплатно
MiKTeX 2.9	бесплатно
MSXML 4.0 SP2 Parser and SDK	бесплатно

MySQL Workbench 6.3 CE	бесплатно
NetBeans IDE 8.0.2	бесплатно
Notepad++	бесплатно
Origin 8.1 Sr2	договор №13918/M41 от 24.09.2009 с ЗАО «СофтЛайн Трейд»;
Python 3.4.3	бесплатно
WinDjView 2.1	бесплатно
WCF RIA Services V1.0 SP2	бесплатно
Многофункциональный редактор ONLYOFFICE	бесплатное
ПО	бесплатно
ОС Linux Ubuntu	бесплатное ПО

3) Современные профессиональные базы данных и информационные справочные системы

1. ЭБС Лань <https://e.lanbook.com/> Договор № 4-е/23 от 02.08.2023 г.
2. ЭБС Znanium.com <https://znanium.com/> Договор № 1106 эбс от 02.08.2023 г.
3. ЭБС Университетская библиотека online <https://biblioclub.ru> Договор № 02-06/2023 от 02.08.2023 г.
4. ЭБС ЮРАЙТ <https://urait.ru/> Договор № 5-е/23 от 02.08.2023 г.
5. ЭБС IPR SMART <https://www.iprbookshop.ru/> Договор № 3-е/23К от 02.08.2023 г.
6. Репозиторий ТвГУ <http://eprints.tversu.ru>

4) Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины:

<https://cyberleninka.ru/> научная электронная библиотека «Киберленинка».

Научная электронная библиотека eLIBRARY.RU (подписка на журналы)
https://elibrary.ru/projects/subscription/rus_titles_open.asp;

VI. Методические указания для обучающихся по освоению дисциплины

Методические рекомендации по организации самостоятельной работы студентов

На лекциях дается необходимый теоретический материал по темам и представлены семестровые задания для решения на лабораторных занятиях в аудитории под руководством преподавателя и самостоятельно. Многие задачи являются стандартными и имеют уже готовые шаблоны (алгоритмы) решения, тем не менее, для получения большего познавательного и учебного эффекта, рекомендуется написание собственного оригинального кода.

Самостоятельная работа студентов в рамках данной дисциплины в основном состоит в подготовке к лабораторным занятиям, написании реферата,

создании презентации по теме, и работе с разными источниками. Освоению учебного материала большую помощь окажет личный творческий подход, связанный с дополнительным просмотром материала по отдельным темам.

Самостоятельная работа является необходимой на всей стадиях и при всех формах изучения предмета. Часы для самостоятельной работы, из всего объема времени затраченного на дисциплину, будут превосходить иные виды работ.

Рекомендуется немедленно обсуждать любые возникшие в процессе обучения вопросы, проблемы и неясности с преподавателем, не откладывая это обсуждение до контрольной точки. Проконсультироваться с преподавателем можно во время и после лабораторных занятий, во время консультаций, а также по электронной почте и в личном кабинете электронной образовательной среды (LMS) или в чате teams.

Требования к рейтинг-контролю для студентов очной формы обучения.

Текущая работа студентов очной формы обучения в 6 семестре оценивается в 100 баллов, которые распределяются между двумя модулями (периодами обучения) следующим образом:

Модуль (период обучения)	Максимальная сумма баллов в модуле	Максимальная сумма баллов за выполнение семестровых заданий на лабораторных занятиях	Реферирование, создание презентации, выступление с докладом	Максимальный балл за рейтинговую контрольную работу
1	35	20		15
2	65	35	15	15

Текущая работа студентов очной формы обучения в 7 семестре оценивается в 60 баллов, которые распределяются между двумя модулями (периодами обучения) следующим образом:

Модуль (период обучения)	Максимальная сумма баллов в модуле	Максимальная сумма баллов за выполнение семестровых заданий на лабораторных занятиях	Реферирование, создание презентации, выступление с докладом	Максимальный балл за рейтинговую контрольную работу
1	20	10		10
2	40	20	10	10

Правила формирования рейтинговой оценки и шкалу пересчета рейтинговых баллов в оценку на экзамене см. в «Положении о рейтинговой системе обучения в ТвГУ»:

<https://tversu.ru/sveden/files/204->

[R_Pologhenie_o_reytingovoy_sisteme_obucheniya_v_TvGU.pdf](#)

VII. Материально-техническое обеспечение

Учебный процесс по данной дисциплине проводится в аудиториях, оснащенных мультимедийными средствами обучения. Для организации самостоятельной работы студентов необходимо наличие персональных компьютеров с доступом в Интернет.

Наименование специальных* помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения. Реквизиты подтверждающего документа
Помещение для самостоятельной работы, учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, практики Компьютерный класс 16 170002, г.Тверь, Садовый пер-к, д. 35. Учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа,	Столы, стулья, переносной ноутбук, компьютеры Столы, стулья, переносной ноутбук,	Adobe Acrobat Reader DC - Russian-бесплатно; Cadence SPB/OrCAD 16.6-Государственный контракт на поставку лицензионных программных продуктов 103 - ГК/09 от 15.06.2009; Git version 2.5.2.2-бесплатно; Google Chrome-бесплатно; Kaspersky Endpoint Security 10 для Windows-Акт на передачу прав ПК545 от 16.12.2022; Lazarus 1.4.0-бесплатно; Mathcad 15 M010-Акт предоставления прав ИС000000027 от 16.09.2011; MATLAB R2012b-Акт предоставления прав № Us000311 от 25.09.2012; Многофункциональный редактор ONLYOFFICE - бесплатно; ОС Linux Ubuntu бесплатное ПО-бесплатно; Microsoft Web Deploy 3.5-бесплатно; MiKTeX 2.9-бесплатно; MSXML 4.0 SP2 Parser and SDK-бесплатно; MySQL Workbench 6.3 CE-бесплатно; NetBeans IDE 8.0.2-бесплатно; Notepad++-бесплатно; Origin 8.1 Sr2-договор №13918/M41 от 24.09.2009 с ЗАО «СофтЛайн Трейд» ; PostgreSQL 9.6 -бесплатно; Python 3.4.3-бесплатно; Visual Studio 2010 Prerequisites - English-Акт на передачу прав №785 от 06.08.2021 г. ; WCF RIA Services V1.0 SP2-бесплатно; WinDjView 2.1-бесплатно; WinPcap 4.1.3-бесплатно; Wireshark 2.0.0 (64-bit)-бесплатно; R studio-бесплатно. Google Chrome-бесплатно;

курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, учебная аудитория 224, 170002, г.Тверь, Садовый пер-к, д. 35	проектор	Kaspersky Endpoint Security 10 для Windows-Акт на передачу прав ПК545 от 16.12.2022; Lazarus – бесплатно; OpenOffice –бесплатно; Многофункциональный редактор ONLYOFFICE бесплатное ПО- бесплатно; ОС Linux Ubuntu бесплатное ПО- бесплатно
---	----------	---

Наличие учебно-наглядных пособий, презентаций для проведения занятий лекционного и семинарского типа, обеспечивающих тематические иллюстрации.

VIII. Сведения об обновлении рабочей программы дисциплины

№п.п.	Обновленный раздел рабочей программы дисциплины	Описание внесенных изменений	Реквизиты документа, утвердившего изменения
	V. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	Обновление списка литературы. Обновление ссылок из ЭБС.	Протокол № 1 от 27.09.2015
	VII. Методические указания для обучающихся по освоению дисциплины.	Корректировка планов практических (семинарских) занятий и методических рекомендаций к ним.	Протокол № 1 от 01.09.2016
	I - X	Корректировка всех разделов в соответствии с новым стандартом	Протокол № 6 от 28.02.2017
	V. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	Дополнение списков. Обновление ссылок из ЭБС.	Протокол № 1 от 01.09.2018
5.	I - VIII	Корректировка всех разделов в соответствии с новым стандартом	Протокол № 10 от 29.06.2021
6	V. Учебно-методическое и информационное обеспечение дисциплины	Обновление списков ПО. Обновление ссылок из ЭБС.	Протокол № 1 от 1.09.2023
7	II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических	Корректировка наименований разделов и тем. Корректировка	Протокол № 7 от 7.03.2024

	часов и видов учебных занятий, IV. Оценочные материалы для проведения текущей и промежуточной аттестации	оценочных материалов	
--	--	----------------------	--