

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Нижегородский государственный технический университет
им. Р.Е. Алексеева» (НГТУ)

Учебно-научный институт радиоэлектроники и информационных технологий
(ИРИТ)

(Полное и сокращенное название института, реализующего данное направление)

УТВЕРЖДАЮ:

Директор института:

_____ А.В. Мякинков
подпись ФИО

02 июня 2025г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.Б.21 Языки и методы программирования

(индекс и наименование дисциплины по учебному плану)

для подготовки бакалавров

Направление подготовки: 01.03.02 Прикладная математика и информатика

Направленность: Математическое моделирование и компьютерные технологии

Форма обучения: очная

Год начала подготовки 2023, 2024, 2025

Выпускающая кафедра ПМ

Кафедра-разработчик ПМ

аббревиатура кафедры

Объем дисциплины 288/8

часов/ з. е

Промежуточная аттестация экзамен

Разработчик: Горенкова А. В., ассистент

Нижний Новгород, 2025

Рабочая программа дисциплины: разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО 3++) по направлению подготовки 01.03.02 Прикладная математика и информатика, утвержденного приказом МИНОБРНАУКИ РОССИИ от 10 января 2018 года № 9, на основании учебного плана принятого УМС НГТУ протокол № 21 от 18.05.2023,
№ 16 от 21.05.2024,
№ 6 от 17.12.2024.

Рабочая программа одобрена на заседании кафедры протокол от 16.05.2025 № 8

Зав. кафедрой д.ф.-м.н, профессор А.А. Куркин

Программа рекомендована к утверждению ученым советом ИРИТ.

Протокол от 20.05.2025 № 1

Рабочая программа зарегистрирована в УМУ регистрационный № 01.03.02-п-21

Начальник МО _____ Е.Г. Севрюкова

Заведующая отделом комплектования НТБ _____ Н.И.Кабанина
(подпись)

СОДЕРЖАНИЕ

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	4
1.1 Цель освоения дисциплины	4
1.2 Задачи освоения дисциплины (модуля)	4
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	4
3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ).....	5
4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	7
4.1 РАСПРЕДЕЛЕНИЕ ТРУДОЁМКОСТИ ДИСЦИПЛИНЫ ПО ВИДАМ РАБОТ ПО СЕМЕСТРАМ.....	7
4.2 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ	8
5. ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ	13
5.1 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности	13
5.2 Описание показателей и критериев контроля успеваемости, описание шкал оценивания.....	13
6. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	16
6.1 Учебная литература, печатные издания библиотечного фонда	16
6.2. Справочно-библиографическая литература.	16
7. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	18
7.1 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)	18
7.2 Перечень программных продуктов, используемых при проведении различных видов занятий по дисциплине.....	18
7.3 Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства необходимого для освоения дисциплины.....	18
8. ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОВЗ.....	19
9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ	20
10. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ	20
11. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	22

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1 Цель освоения дисциплины

Целью освоения дисциплины «Языки и методы программирования» является знакомство с понятием программного обеспечения, изучение базовых и продвинутых техник и технологий разработки программного обеспечения, получение навыков самостоятельной разработки программного обеспечения для решения прикладных задач в соответствии с Федеральным государственным образовательным стандартом.

1.2 Задачи освоения дисциплины (модуля)

Дисциплина «Языки и методы программирования» способствует подготовке студентов к решению следующих профессиональных задач:

1. Изучение и разработка математических моделей, алгоритмов, методов, программного обеспечения и инструментальных средств для решения прикладных задач;
2. Изучить и освоить базовые техники программирования на языках высокого уровня: императивное и процедурное программирование на языке Си;
3. Изучить и освоить технологии объектно-ориентированного программирования и объектно-ориентированного проектирования, реализация на языке C++.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Учебная дисциплина Б1.Б.21 «Языки и методы программирования» включена в перечень дисциплин базовой части, определяющий направленность образовательной программы «Прикладная математика и информатика». Дисциплина реализуется в соответствии с требованиями ФГОС, ОП ВО и УП.

Дисциплина базируется на дисциплинах программы бакалавриата по направлению «Прикладная математика и информатика» профиля «Математическое моделирование и компьютерные технологии».

Дисциплина «Языки и методы программирования» является основополагающей для изучения дисциплин «Структуры данных», «Технология программирования», «Численные методы», «Методы оптимизации», а также для подготовки к сдаче и сдачи государственного экзамена, а также выполнения и защиты ВКР.

Рабочая программа дисциплины «Языки и методы программирования» для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)¹

Таблица 1. Формирование компетенций дисциплинам

Наименование дисциплин, формирующих компетенцию совместно	Семестры, формирования дисциплины Компетенции берутся из Учебного плана по направлению подготовки бакалавра							
	1	2	3	4	5	6	7	8
<i>ОПК-2(Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач), ОПК-5(Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения)</i>								
<i>Языки и методы программирования</i>	*	*						
<i>Структуры данных</i>				*				
<i>Технология программирования</i>			*					
<i>Численные методы</i>					*	*		
<i>Методы оптимизации</i>							*	
<i>Подготовка к сдаче и сдача государственного экзамена</i>								*
<i>Выполнение и защита ВКР</i>								*
<i>ОПК-5. (Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения)</i>								
<i>Языки и методы программирования</i>	*	*						
<i>Технология программирования</i>			*					
<i>Выполнение и защита ВКР</i>								*

ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОПВО

Таблица 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине			Оценочные средства	
					Текущего контроля	Промежуточной аттестации
ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	ИОПК-2.1. Выбирает математические методы решения прикладных задач, строит алгоритмы решения	Знать: основы алгоритмизации; основы императивной парадигмы программирования на языке C++; основы объектно-ориентированной парадигмы программирования на языке C++	Уметь: выполнять задания, связанные с разработкой абстрактного типа данных некоторой предметной области, включая: создание типов данных, методов обработки данных с помощью структур и классов	Владеть: методологией анализа, проектирования и программирования сложных программных систем на основе объектно-ориентированного подхода	Задания для лабораторных работ	Задания для лабораторных работ
ОПК-5. Способен разрабатывать алгоритмы и	ИОПК-5.1. Применяет современные компьютерные	Знать: языки программирования, методы структурного и	Уметь: разрабатывать ясные и надежные программы,	Владеть: навыками работы в современных средах	Задания для лабораторных работ	Задания для лабораторных работ

компьютерные программы, пригодные для практического применения	технологии и существующие системы проектирования программных средств для разработки программного обеспечения.	объектно-ориентированного программирования; систему программирования на алгоритмическом языке высокого уровня; физические основы компьютерной техники и средств передачи информации, принципы работы технических устройств ИКТ.	использовать основные принципы объектно-ориентированного программирования; - определять принципы построения классов, критерии проверки правильности построения классов, основные тенденции в области развития технологий объектно-ориентированного программирования.	программирования, основными методами объектно-ориентированного программирования.		
	ИОПК-5.2. Осуществляет тестирование и доработку компьютерных программ с целью их дальнейшего практического применения.	Знать: основные приемы алгоритмизации и программирования на языке высокого уровня, методы тестирования программ.	Уметь: применять полученные знания к решению вопросов внедрения и эксплуатации сложных программных систем в зависимости от информационных проблем, с которыми сталкиваются конкретные предприятия.	Владеть: методами и средствами тестирования программного обеспечения.	Задания для лабораторных работ	Задания для лабораторных работ

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1 Распределение трудоёмкости дисциплины по видам работ по семестрам

Общая трудоёмкость дисциплины составляет 8 зач.ед. 288 часа, распределение часов по видам работ семестрам представлено в таблице 3.

Таблица 3. Распределение трудоёмкости дисциплины по видам работ по семестра для студентов очного обучения

Вид учебной работы	Трудоёмкость в час		
	Всего час.	В т.ч. по семестрам	
		1 сем	2 сем
Формат изучения дисциплины	с использованием элементов электронного обучения		
Общая трудоёмкость дисциплины по учебному плану	288	144	144
1. Контактная работа:	110	55	55
Аудиторная работа, в том числе:	102	51	51
занятия лекционного типа (Л)	34	17	17
занятия семинарского типа (ПЗ-семинары, практ. занятия и др)			
лабораторные работы (ЛР)	68	34	34
Внеаудиторная, в том числе	8	4	4
курсовая работа (проект) (КР/КП) (консультация, защита)			
текущий контроль, консультации по дисциплине			
контактная работа на промежуточном контроле (КРА)	8	4	4
2. Самостоятельная работа (СРС)	106	53	53
реферат/эссе (подготовка)			
расчётно-графическая работа (РГР) (подготовка)			
контрольная работа			
курсовая работа/проект (КР/КП) (подготовка)			
самостоятельное изучение разделов, самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к лабораторным и практическим занятиям, коллоквиум и т.д.)	106	53	53
Подготовка к экзамену (контроль)	72	36	36

4.2Содержание дисциплины, структурированное по темам

Таблица 4. Содержание дисциплины, структурированное по темам для студентов очного обучения

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы (час)				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий	Реализация в рамках Практической подготовки (трудоемкость в часах)	Наименование разработанного Электронного курса (трудоемкость в часах)
		Контактная работа			Самостоятельная работа студентов (час)				
		Лекции (час)	Лабораторные работы (час)	Практические занятия (час)					
1 семестр									
Раздел 1. Парадигмы программирования									
ОПК-2, ОПК-5, ИОПК-2.1, ИОПК-5.1, ИОПК-5.2	Тема 1.1 Парадигмы программирования. Определение и основные модели. Обзор популярных языков программирования их различия по поддержке парадигм	1			2	Подготовка к лекциям и практическим занятиям [6.1.1., 6.1.2., 6.2.1.]	Лекция-визуализация	—	—
	Тема 1.2 Компиляция и интерпретация. Различия, компиляторы и интерпретаторы различных языков программирования. Этапы компиляции. Работа с компилятором gcc.	1			2		Лекция-объяснение с частичным привлечением формы дискуссии, беседы	—	—
	Итого по 1 разделу	2			4		—	—	—
Раздел 2. Процедурное программирование на примере языка С									
ОПК-2, ОПК-5, ИОПК-2.1, ИОПК-5.1, ИОПК-5.2	Тема 2.1 Алфавит, синтаксис и грамматика языка С. Отличия от других языков программирования. Директивы препроцессора.	1	2		2	Подготовка к лекциям и практическим занятиям [6.1.1., 6.1.2., 6.2.1.]	Лекция-объяснение с частичным привлечением формы дискуссии, беседы	—	—
	Тема 2.2 Фундаментальные типы данных. Представление переменных в целочисленном формате, хранение чисел с плавающей точкой в памяти. Объявление и инициализация переменных в языке С. Константы.	1	2		3		—	—	—

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы (час)				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий	Реализация в рамках Практической подготовки (трудоемкость в часах)	Наименование разработанного Электронного курса (трудоемкость в часах)	
		Контактная работа			Самостоятельная работа студентов (час)					
		Лекции (час)	Лабораторные работы (час)	Практические занятия (час)						
	Тема 2.3 Операции в языках программирования: арифметические, логические, присваивание. Последовательность и приоритеты вычислений.	1	2		2		—	—	—	
	Тема 2.4 Функции в языке C: объявление, реализация и вызов функций; аргументы, возвращаемое значение функций. Рекурсивные функции.	1	2		3		—	—	—	
	Тема 2.5 Операторы: условные операторы (if, switch, тернарный оператор), циклические операторы (for, while, do while).	2	3		4			—	—	
	Тема 2.6 Указатели и адресная арифметика. Передача аргументов функции через указатель.	1	1		4		—	—	—	
	Тема 2.7 Управление памятью: стек, куча, data segment, BSS. Массивы на стеке. Массивы на куче (динамические массивы), особенности реализации многомерных динамических массивов.	1	4		7		Контрольная работа	—	—	
	Тема 2.8 Строковые массивы. Стандартная библиотека языка C для управления строковыми массивами.	1	2		4			—	—	—
	Тема 2.9 Работа с текстовыми и бинарными файлами: режимы открытия и перемещение по файлу. Функции для работы с файлами в языке C.	1	4		3			—	—	—
	Тема 2.10 Сложные типы данных: структуры, объединения,	2	6		10		—	—	—	

Планируемые (контролируем ые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы (час)				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий	Реализация в рамках Практической подготовки (трудоемкость в часах)	Наименование разработанного Электронного курса (трудоемкость в часах)
		Контактная работа			Самостоятельная работа студентов (час)				
		Лекции (час)	Лабораторны е работы (час)	Практические занятия (час)					
	перечисления.								
	Тема 2.11 Аргументы функции main. Функции с переменным количеством аргументов.	1	2		2		—	—	—
	Тема 2.12 Односвязные и двусвязные списки. Реализация на языке C.	2	4		5		—	—	—
	Итого по 2 разделу	15	30		49	—	—	—	—
	Итого за семестр 1	17	34		53	—			
Раздел 3. Парадигма объектно-ориентированного программирования									
ОПК-3 ИОПК-3.3.	Тема 3.1 Принципы объектно- ориентированного программирования.	0.5			1	Подготовка к лекциям и практическим занятиям [6.1.1., 6.1.2., 6.2.1.]	Лекция-диалог	—	—
	Тема 3.2 Отличия в построении программы согласно объектно- ориентированной и процедурной парадигмам.	0.5			1		Лекция-объяснение с частичным привлечением формы дискуссии, беседы	—	—
		Итого по 3 разделу	1			2		—	—
Раздел 4. Объектно-ориентированное программирование на примере языка C++									
ОПК-3 ИОПК-3.3.	Тема 4.1 Пространства имён в C++. Пространство имён стандартной библиотеки.	0.5			2	Подготовка к лекциям и практическим занятиям [6.1.1., 6.1.2., 6.2.1.]	Лекция-диалог	—	—
	Тема 4.2 Виды инициализации в C++: инициализация копированием, списковая инициализация. Запрет неявного приведения типов.	0.5			2		Лекция-визуализация	—	—
	Тема 4.3 Ссылки и указатели. Отличие ссылки от указателя, передача аргументов функции по ссылке.	1	2		4		Лекция-объяснение с частичным привлечением формы дискуссии, беседы	—	—
	Тема 4.4 Классы: объявление класса, создание объекта класса. Обращение	1	2		4		Контрольная работа	—	—

Планируемые (контролируем ые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы (час)				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий	Реализация в рамках Практической подготовки (трудоемкость в часах)	Наименование разработанного Электронного курса (трудоемкость в часах)
		Контактная работа			Самостоятельная работа студентов (час)				
		Лекции (час)	Лабораторны е работы (час)	Практические занятия (час)					
	к полям и методам класса. Отличие класса от структуры.								
	Тема 4.5 Классы: модификаторы доступа к полям и методам класса.	1	2		2		—	—	—
	Тема 4.6 Инициализация членов класса. Идиома RAII: конструкторы и деструкторы.Виды конструкторов: по умолчанию, с параметры, копирующий. Удаление и автоматическая реализация конструктора.	1	2		4		—	—	—
	Тема 4. Наследование: публичное, защищённое и закрытое. Изменения режимов доступа к полям и методам класса при наследовании. Множественное наследование. Проблема ромба.	1	2		4		—	—	—
	Тема 4.8 Порядок вызова конструкторов и деструкторов при наследовании. Виртуальные деструкторы.	0.5			2		—	—	—
	Тема 4.9 Перегрузка функций. Виртуальные функции и полиморфизм. Чисто виртуальные функции и абстрактные классы. Особенности наследования классов с виртуальными функциями и ключевое слово final.	1	2		3		—	—	—
	Тема 4.10 Приведение типов при наследовании: от базового к наследнику и наоборот. Динамическая идентификация типов и функции приведения типов.	1	2		2		—	—	—

Планируемые (контролируем ые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы (час)				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий	Реализация в рамках Практической подготовки (трудоемкость в часах)	Наименование разработанного Электронного курса (трудоемкость в часах)
		Контактная работа			Самостоятельная работа студентов (час)				
		Лекции (час)	Лабораторны е работы (час)	Практические занятия (час)					
	Тема 4.11 Шаблоны функций и классов. Шаблон строкового массива std::string.	1	4		4		—	—	—
	Тема 4.12 Функции для работы с файлами в C++.	1	2		2		Контрольная работа	—	—
	Тема 4.13 Перегрузка операторов для класса: арифметических, логических, присваивания.	1	4		4		—	—	—
	Тема 4.14 Разделение объявления класса и реализации методов класса на несколько файлов. Особенности реализации перегрузки операторов вне класса.	1	2		2		—	—	—
	Тема 4.15 Обработка исключительных ситуаций	0.5	1		2		—	—	—
	Тема 4.16 Контейнеры стандартной библиотеки: массив, вектор, список.	1	3		4		—	—	—
	Тема 4.17 Современные стандарты C++: умные указатели, лямбда-функции и ключевое слово auto.	2	4		4		—	—	—
	Итого по 4 разделу	16	34		51			—	—
	Итого за 2 семестр	17	34		53		—	—	—
	Итого по дисциплине	34	68		106		—	—	—

5. ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Текущий контроль осуществляется по всем видам учебного процесса: беседы, дискуссии по темам лекционных занятий, выполнение практических заданий, расчетно-графической и контрольных работ. Промежуточный контроль проводится в устно-письменной форме.

5.1 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе текущего контроля успеваемости представлены в ФОС дисциплины.

Перечень вопросов, выносимых на промежуточную аттестацию в форме экзамена, хранятся на кафедре «Прикладная математика» ауд. 1204 по адресу Н. Новгород, ул. Минина, 24 и находятся в свободном доступе.

5.2 Описание показателей и критериев контроля успеваемости, описание шкал оценивания

Для оценки знаний, умений и навыков, и формирования компетенций по дисциплине может применяться балльно-рейтинговая/традиционная система контроля и оценивания успеваемости студентов.

Таблица5.

Балльно-рейтинговая система оценивания при текущем контроле (контрольные недели) и оценка выполнения контрольных работ

Шкала оценивания	Экзамен
$40 < R \leq 50$	Отлично
$30 < R \leq 40$	Хорошо
$20 < R \leq 30$	Удовлетворительно
$0 < R \leq 20$	Неудовлетворительно

При промежуточном контроле успеваемость студентов оценивается по четырехбалльной системе «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Таблица 6. Критерии оценивания результата обучения по дисциплине и шкала оценивания

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Критерии оценивания результатов обучения			
		Оценка «неудовлетворительно» / «не зачтено» 0-59% от max рейтинговой оценки контроля	Оценка «удовлетворительно» / «зачтено» 60-74% от max рейтинговой оценки контроля	Оценка «хорошо» / «зачтено» 75-89% от max рейтинговой оценки контроля	Оценка «отлично» / «зачтено» 90-100% от max рейтинговой оценки контроля
ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	ИОПК-2.1. Выбирает математические методы решения прикладных задач, строит алгоритмы решения	Изложение учебного материала бессистемное, неполное, не освоены правовые нормы принятия управленческого решения, непонимание их использования в рамках поставленных целей и задач; неумение делать обобщения, выводы, что препятствует усвоению последующего материала	Фрагментарные, поверхностные знания лекционного курса; изложение полученных знаний неполное, однако это не препятствует усвоению последующего материала; допускаются отдельные существенные ошибки, исправленные с помощью преподавателя; затруднения при формулировании результатов и их решений	Знает материал на достаточно хорошем уровне; представляет основные задачи в рамках постановки целей и выбора оптимальных способов их достижения при управлении проектом. Умеет использовать правовую документацию для определения круга задач.	Имеет глубокие знания всего материала структуры дисциплины; освоил новации лекционного курса по сравнению с учебной литературой; изложение полученных знаний полное, системное; допускаются единичные ошибки, самостоятельно исправляемые при собеседовании
ОПК-5. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ИОПК-5.1. Применяет современные компьютерные технологии и существующие системы проектирования программных средств для разработки программного обеспечения.	Изложение учебного материала бессистемное, незнание правовых норм, что препятствует усвоению последующей информации; Демонстрирует частичные и слабые умения в определяет имеющихся ресурсов и ограничений	Фрагментарные, поверхностные знания важнейших разделов. Посредственно - осуществляет поиск, критический анализ и синтез информации, ошибки при применении системного подхода для решения поставленных задач	Владеет знаниями и навыками при применении ресурсов и их использованием; формулирует ограничения для решения ПЗ ; допускает незначительные ошибки, которые сам исправляет; комментирует выполняемые действия не всегда точно.	Имеет глубокие знания всего материала; в полной мере владеет классификацией ресурсов; свободно осуществляет поиск правовых и нормативных документов в практических примерах в различных ситуациях.
	ИОПК-5.2. Осуществляет тестирование и	Изложение учебного материала бессистемное, незнание правовых норм,	Фрагментарные, поверхностные знания важнейших разделов.	Владеет знаниями и навыками при применении ресурсов и	Имеет глубокие знания всего материала;

	доработку компьютерных программ с целью их дальнейшего практического применения.	что препятствует усвоению последующей информации; Демонстрирует частичные и слабые умения в определяет имеющихся ресурсов и ограничений	Посредственно - осуществляет поиск, критический анализ и синтез информации, ошибки при применении системного подхода для решения поставленных задач	их использованием; формулирует ограничения для решения ПЗ; допускает незначительные ошибки, которые сам исправляет; комментирует выполняемые действия не всегда точно.	в полной мере владеет классификацией ресурсов; свободно осуществляет поиск правовых и нормативных документов в практических примерах в различных ситуациях.
--	--	---	---	--	---

Таблица 7. Критерии оценивания

Оценка	Критерии оценивания
Высокий уровень «5» (отлично)	оценку «отлично» заслуживает студент, свободно и уверенно оперирующий предоставленной информацией, отлично владеющий навыками написания и анализа программного кода, знающий все основные методы решения проблем, предусмотренные учебной программой, знающий типичные ошибки и возможные сложности при решении той или иной проблемы и способный выбрать и эффективно применить метод решения конкретной проблемы. Студент способен легко ориентироваться при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приёмами выполнения практических заданий.
Средний уровень «4» (хорошо)	оценку «хорошо» заслуживает студент, способный логично мыслить, излагающий материал, не допуская существенных неточностей. Студент способен эффективно применять теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приёмами их выполнения, допускает единичные ошибки в решении проблем.
Пороговый уровень «3» (удовлетворительно)	оценку «удовлетворительно» заслуживает студент, способный применять знания только основного материала, допускающий неточности, недостаточно правильные формулировки. Студент допускает нарушения логической последовательности в изложении программного материала, имеются затруднения с выводами, способен к решению конкретных практических задач из числа предусмотренных рабочей программой.
Минимальный уровень «2» (неудовлетворительно)	оценку «неудовлетворительно» заслуживает студент, не способный излагать материал последовательно, допускающий существенные ошибки, неуверенно, с большими затруднениями выполняющий практические задания.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Учебная литература, печатные издания библиотечного фонда

Библиотечный фонд укомплектован печатными изданиями из расчета не менее 0,25 экземпляра каждого из изданий, указанных ниже на каждого обучающегося из числа лиц, одновременно осваивающих соответствующую дисциплину (модуль).

6.1.1 Подбельский, Вадим Валериевич. Программирование на языке Си : [Учеб. пособие для вузов по направлениям<Прикладная математика и информатика>,<Информатика и вычисл. техника>, специальностям<Прикладная математика>,<Вычисл. машины, комплексы, системы и сети упр.>] / В. В. Подбельский, С. С. Фомин. - 2-е изд., доп. - М. : Финансы и статистика, 2003. - 600 с. : ил.; 21 см.; ISBN 5279021806

6.1.2 Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма [и др.] ; [пер. с англ. А. Слинкин]. – Москва [и др.] : Питер, 2012. – 366 с. – (Библиотека программиста). – ISBN 978-5-469-01136-1.

6.2. Справочно-библиографическая литература.

6.2.1 Страуструп, Б. Дизайн и эволюция языка C++. Учебное пособие / Б. Страуструп. – Москва : ДМК Пресс, 2008. – 448 с. – ISBN 5-94074-005-7

6.2.2 Свердлов С.З. Языки программирования и методы трансляции: Учеб пособие / С.З.Свердлов. - СПб.: Питер, 2007. - 638 с.: ил. - ISBN 978-5-469-00378-6.

6.2.3 Симонович С.П. Информатика. Базовый курс.: Учеб пособие / С.П. Симонович. -3-е изд. - СПб.: Питер, 2011. - 640 с. - ISBN 978-5-496-00217-2.

7. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебный процесс по дисциплине обеспечен необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства (состав по дисциплине определен в настоящей РПД и подлежит обновлению при необходимости).

7.1 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

Перечень программных продуктов, используемых при проведении различных видов занятий по дисциплине (открытый доступ):

- 1) консультантПлюс [Электронный ресурс]: Справочная правовая система. - Режим доступа: <http://www.consultant.ru>;
- 2) научная электронная библиотека E-LIBRARY.ru. – Режим доступа: <http://elibrary.ru/defaultx.asp>;
- 3) электронная библиотечная система Поволжского государственного университета сервиса [Электронный ресурс]. – Режим доступа: <http://elib.tolgas.ru>;
- 4) электронно-библиотечная система Znanium.com [Электронный ресурс]. - Режим доступа: <http://znanium.com>;
- 5) открытое образование [Электронный ресурс]. - Режим доступа: <https://openedu.ru>;
- 6) polpred.com. Обзор СМИ. Полнотекстовая, многоотраслевая база данных (БД) [Электронный ресурс]. - Режим доступа: <http://polpred.com>;
- 7) базы данных Всероссийского института научной и технической информации (ВИНИТИ РАН) по естественным, точным и техническим наукам [Электронный ресурс]. - Режим доступа: <http://www.viniti.ru>;
- 8) университетская информационная система Россия [Электронный ресурс]. - Режим доступа: <http://uisrussia.msu.ru>.

7.2 Перечень программных продуктов, используемых при проведении различных видов занятий по дисциплине

Таблица 8. Перечень электронных библиотечных систем

№	Наименование ЭБС	Ссылка к ЭБС
1	Консультант студента	http://www.studentlibrary.ru/
2	Лань	https://e.lanbook.com/
3	Юрайт	https://biblio-online.ru/
4	TNT-ebook	https://www.tnt-ebook.ru/

7.3 Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства необходимого для освоения дисциплины

Таблица 9. Перечень программного обеспечения

Программное обеспечение, используемое в университете на договорной основе	Программное обеспечение свободного распространения
Microsoft Windows XP, Prof, S/P3 (подписка DreamSpark Premium, договор №Tr113003 от 25.09.2014)	Calculate Linux (свободное ПО)
Microsoft Windows 7 (подписка MSDN 4689, подписка DreamSparkPremium, договор № Tr113003 от 25.09.2014)	Open Office 4.1.1 (лицензия Apache License 2.0)
Visual Studio 2008 (подписка DreamSpark Premium, договор №Tr113003 от 25.09.2014)	Adobe Reader 11 (проприетарное ПО)
Microsoft Office Professional Plus 2007 (лицензия № 42470655)	Libre office 5.2.4.2 (свободное ПО, лицензия Mozilla Public License)
Microsoft Office (лицензия № 43178972)	Eclipse (открытое ПО, лицензия Eclipse Public

Программное обеспечение, используемое в университете на договорной основе	Программное обеспечение свободного распространения
	License)
Windows XP лиц. № 65609340	
Office 2007 лиц. № 43178971	
Microsoft Windows XP Professional (лицензия № 43178980)	
Microsoft Office 2007 (лицензия № 44804588)	
Adobe Design Premium CS 5.5.5 (лицензия № 65112135)	

Перечень современных профессиональных баз данных и информационных справочных систем

В таблице 10 указан перечень профессиональных баз данных и информационных справочных систем, к которым обеспечен доступ (удаленный доступ). Данный перечень подлежит обновлению в соответствии с требованиями ФГОС ВО.

Таблица 10

Перечень современных профессиональных баз данных и информационных справочных систем

Наименование профессиональной базы данных, информационно-справочной системы	Доступ к ресурсу (удаленный доступ с указанием ссылки/доступ из локальной сети университета)
База данных стандартов и регламентов РОССТАНДАРТ	https://www.gost.ru/portal/gost//home/standarts
Электронная база избранных статей по философии	http://www.philosophy.ru
Единый архив экономических и социологических данных	http://sophist.hse.ru/data_access.shtml
Базы данных Национального совета по оценочной деятельности	http://www.ncva.ru
Информационно-справочная система «Техксперт»	доступ из локальной сети

8. ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОВЗ

В таблице 11 указан перечень образовательных ресурсов, имеющих формы, адаптированные к ограничениям их здоровья, а также сведения о наличии специальных технических средств обучения коллективного и индивидуального пользования. При заполнении таблицы может быть использована информация, размещенная в подразделе «Доступная среда» специализированного раздела сайта НГТУ «Сведения об образовательной организации» <https://www.nntu.ru/sveden/accenv/>

Таблица 11. Образовательные ресурсы для инвалидов и лиц с ОВЗ

№	Перечень образовательных ресурсов, приспособленных для использования инвалидами и лицами с ОВЗ	Сведения о наличии специальных технических средств обучения коллективного и индивидуального пользования
1	ЭБС «Консультант студента»	озвучение книг и увеличение шрифта
2	ЭБС «Лань»	специальное мобильное приложение – синтезатор речи, который воспроизводит тексты книг и меню навигации
3	Образовательная платформа «Юрайт»	версия для слабовидящих

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Учебные аудитории для проведения занятий по дисциплине, оснащены оборудованием и техническими средствами обучения.

В таблице 12 перечислены: учебные аудитории для проведения учебных занятий, оснащенные оборудованием и техническими средствами обучения; помещения для самостоятельной работы обучающихся, которые должны оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду НГТУ.

Таблица 12. Оснащенность аудиторий для проведения учебных занятий по дисциплине

Номер аудитории	Наименование аудиторий и помещений для самостоятельной работы	Оснащенность аудиторий помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения и реквизиты подтверждающего документа
6421	Мультимедийная аудитория № 6421 учебно-лабораторного корпуса № 6	1. Доска меловая – 1 шт. 3. Экран – 1 шт. 4. Мультимедийный проектор Epson X12 – 1 шт. 5. Компьютер PC MB Asus на чипсете Nvidia/AMDAthlonXII CPU 2.8Ggz/ RAM 4 Ggb/SVGASandartGraphics +Ge-FORCE Nvidia GT210/HDD 250Ggb,SATAinterface, монитор 19", с выходом на проектор. 6. Рабочее место студента - 74 7. Рабочее место для преподавателя – 1 шт.	1. Windows 7 32 bit корпоративная; VL 49477S2 2. Adobe Acrobat Reader DC-Russian (беспл.) 3. Microsoft Office Professional Plus 2007 (лицензия № 42470655); 4. Dr.Web (с/н 758S-TDJP-N7HB-ZH2F от 26.05.2025, до 31.05.26)
6543	Помещение для самостоятельной работы студентов (Компьютерный класс № 1) № 6543 учебно-лабораторного корпуса № 6	1. Рабочие места студента, оснащенные ПК на базе Intel Core i5 с мониторами – 8 шт. 2. Рабочие места студента, оснащенные ПК на базеCore 2 Duo с мониторами –2 шт. 3. Рабочее место преподавателя, оснащенное ПК на базе Intel Core i5 с монитором – 1 шт. 4. Проектор Accer, проекционный экран – 1 шт. ПК подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета 5. Принтер HP LaserJet 1200 – 1 шт.	1. Microsoft Windows 7 MSDN реквизиты договора - подписка DreamSpark Premium, договор № 0509/KMP от 15.10.18 2. Бесплатное ПО: Пакет программ Open Office, True Conf, Браузер Google Chrome, Браузер Mozilla Firefox, Браузер Opera, McAfee Security Scan, Adobe Acrobat Reader DC, AutoCAD2013

10. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

Контактная работа: аудиторная, внеаудиторная, а также проводится в электронной информационно-образовательной среде университета (ЭИОС).

Преподавание дисциплины ведется с применением следующих видов образовательных технологий: балльно-рейтинговая технология оценивания.

При преподавании дисциплины «Языки и методы программирования», используются современные образовательные технологии, позволяющие повысить активность студентов при освоении материала курса и предоставить им возможность эффективно реализовать часы самостоятельной работы.

На лекциях, практических занятиях реализуются интерактивные технологии, приветствуются вопросы и обсуждения, используется личностно-ориентированный подход, технология работы в малых группах, что позволяет студентам проявить себя, получить навыки самостоятельного изучения материала, выровнять уровень знаний в группе.

Все вопросы, возникшие при самостоятельной работе над домашним заданием, подробно разбираются на практических занятиях и лекциях. Проводятся индивидуальные и групповые консультации с использованием, как встреч студентами, так и современных информационных технологий: электронная почта, *ZOOM*.

Иницируется активность студентов, поощряется задание любых вопросов по материалу, практикуется индивидуальный ответ на вопросы студента, рекомендуются методы успешного самостоятельного усвоения материала в зависимости от уровня его базовой подготовки.

Для оценки знаний, умений, навыков и уровня сформированности компетенции применяется балльно-рейтинговая система контроля и оценки успеваемости студентов в процессе текущего контроля.

Промежуточная аттестация проводится в форме экзамена с учётом текущей успеваемости.

Результат обучения считается сформированным на повышенном уровне, если теоретическое содержание курса освоено полностью. При устных собеседованиях студент исчерпывающе, последовательно, четко и логически излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, использует в ответе дополнительный материал. Все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты, проявляет самостоятельность при выполнении заданий.

Результат обучения считается сформированным на пороговом уровне, если теоретическое содержание курса освоено полностью. При устных собеседованиях студент последовательно, четко и логически стройно излагает учебный материал; справляется с задачами, вопросами и другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий

Результат обучения считается несформированным, если студент при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет задания, не демонстрирует необходимых умений, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже трех по оценочной системе, что соответствует допороговому уровню.

10.1. Методические указания для занятий лекционного типа

Лекционный курс предполагает систематизированное изложение основных вопросов тематического плана. В ходе лекционных занятий раскрываются базовые вопросы в рамках каждой темы дисциплины (Таблица 4). Обозначаются ключевые аспекты тем, а также делаются акценты на наиболее сложные и важные положения изучаемого материала. Материалы лекций являются опорной основой для подготовки обучающихся к практическим занятиям и выполнения заданий самостоятельной работы, а также к мероприятиям текущего контроля успеваемости и промежуточной аттестации по дисциплине.

В ходе лекционных занятий рекомендуется вести конспектирование учебного материала.

10.2. Методические указания по освоению дисциплины на лабораторных работах

Лабораторные работы являются практической частью освоения дисциплины, направленной на закрепление теоретического материала, полученного на лекциях. Они охватывают все основные разделы курса и проводятся в целях формирования уверенных навыков работы с инструментами, технологиями и приёмами программно-аппаратного взаимодействия.

Основной формой выполнения лабораторных работ является практическое выполнение заданий и написание скриптов и программ с их последующим тестированием и анализом.

Лабораторные занятия обеспечивают:

- закрепление и углубление знаний, полученных на лекциях;
- формирование устойчивых практических навыков работы в среде командной строки, git, bash, системах сборки и операционной системе Linux;
- развитие умений применять знания для выполнения прикладных задач в рамках курса;
- контроль успеваемости и фиксацию уровня освоения материала через рубежную и итоговую систему оценивания.

10.3. Методические указания по освоению дисциплины на практических занятиях

Практические занятия не предусмотрены курсом.

10.4. Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обеспечивает подготовку обучающегося к аудиторным занятиям и мероприятиям текущего контроля и промежуточной аттестации по изучаемой дисциплине. Результаты этой подготовки проявляются в активности обучающегося на занятиях и в качестве выполненных практических заданий и других форм текущего контроля.

При выполнении заданий для самостоятельной работы рекомендуется проработка материалов лекций по каждой пройденной теме, а также изучение рекомендуемой литературы, представленной в разделе 6. В процессе самостоятельной работы при изучении дисциплины студенты могут работать на компьютере в специализированных аудиториях для самостоятельной работы (указано в таблице 12). В аудиториях имеется доступ через информационно-телекоммуникационную сеть «Интернет» к электронной информационно-образовательной среде университета (ЭИОС) и электронной библиотечной системе (ЭБС), где в электронном виде располагаются учебные и учебно-методические материалы, которые могут быть использованы для самостоятельной работы при изучении дисциплины.

10.5. Методические указания для выполнения РГР

РГР не предусмотрены курсом.

10.6. Методические указания для выполнения курсового проекта/работы

Выполнение курсового проекта/работы не предусмотрено учебным планом.

11. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

11.1. Оценочные средства для проведения текущего контроля успеваемости

Текущий контроль осуществляется по всем видам учебного процесса:

- выполнение лабораторных работ,
- обсуждение тем лекционных занятий в форме дискуссии, беседы,
- контрольные работы.

1. Задания для лабораторных работ

1. Вывести на экран таблицу значений функции при всех значениях переменных с

указанным шагом. Для вычисления \sin , \cos , корня и т.д. использовать библиотеку `math.h`.

2. Написать программу, которая а) запросит у пользователя размеры и все элементы матрицы; б) выполнит действия с матрицей согласно варианту и выведет на экран результат данных действий.

Варианты заданий:

1	<ol style="list-style-type: none"> 1. Найти S - сумму элементов среди элементов расположенных на главной диагонали матрицы;; 2. Найти максимальный элемент в 3-ей строке матрицы; 3. Поменять местами найденный максимальный элемент и значение S.
2	<ol style="list-style-type: none"> 1. Создать новый одномерный массив B из минимальных элементов каждой строки матрицы A. Вывести массив на экран; 2. Найти сумму элементов матрицы A; 3. Заменить последний элемент матрицы A на сумму S.
3	<ol style="list-style-type: none"> 1. Найти произведение положительных элементов среди элементов, расположенных выше побочной диагонали матрицы; 2. Найти минимальный элемент в 2-ом столбце матрицы; 3. Поменять местами найденный минимальный элемент и значение 3-его элемента в 4-ой строке
4	<ol style="list-style-type: none"> 1. Создать новый одномерный массив из сумм элементов каждого столбца матрицы A. Вывести массив на экран; 2. Найти максимальный элемент, среди элементов главной диагонали матрицы A; 3. Заменить третий элемент в 1 строке матрицы на максимальный элемент.
5	<ol style="list-style-type: none"> 1. Найти произведение элементов среди элементов расположенных на главной диагонали матрицы; 2. Найти минимальный элемент в 3-ей строке матрицы; 3. Поменять местами найденный минимальный элемент и значение S.
6	<ol style="list-style-type: none"> 1. Создать новый одномерный массив из сумм элементов каждой строки матрицы. Вывести массив на экран; 2. Найти максимальный элемент матрицы в 5-ой строке; 3. Заменить последний элемент в 3-ей строке матрицы на найденный максимальный элемент.
7	<ol style="list-style-type: none"> 1. Найти сумму отрицательных элементов, среди элементов расположенных на побочной диагонали матрицы; 2. Найти минимальный элемент в 3-ем столбце матрицы; 3. Поменять местами найденный минимальный элемент и значение 3-его элемента в 4-ой строки.
8	<ol style="list-style-type: none"> 1. Создать новый одномерный массив C из произведений элементов каждой строки матрицы A. Вывести массив на экран; 2. Найти максимальный элемент, среди положительных элементов матрицы A; 3. Заменить третий элемент в 1 строке матрицы A на найденный максимальный элемент.
9	<ol style="list-style-type: none"> 1. Найти строку, содержащую минимальный элемент матрицы; 2. Найденную строку поменять местами с 4-ой строкой; 3. В полученной матрице подсчитать сумму всех элементов 2 и 4 строки.
10	<ol style="list-style-type: none"> 1. Найти сумму отрицательных элементов на побочной диагонали; 2. Найти наибольший элемент 2-го столбца и его местоположение; 3. Заменить найденный максимальный элемент во 2-ом столбце на значение суммы.
11	<ol style="list-style-type: none"> 1. Найти сумму положительных элементов над главной диагональю матрицы; 2. Найти максимальный элемент среди отрицательных элементов 4-ой строки;

	3. Заменить найденный максимальный элемент в задании 2) на значение суммы.
12	1. Поменять местами первую и последнюю строки матрицы; 2. Подсчитать S для элементов каждой строки матрицы по формуле: $S_i = 5 + \sum_{j=0}^M \frac{1-a_{ij} + a_{ij}^2}{6}$, где i – номер строки.
13	1. Найти количество отрицательных элементов под главной диагональю; 2. Найти S для элементов каждой строки матрицы по формуле: $S_i = 5 + \sum_{j=0}^M \sin a_{ij} + \frac{a_{ij}}{7}$, где i – номер строки.
14	1. Найти максимальный элемент среди отрицательных элементов матрицы, расположенных над побочной диагональю; 2. Подсчитать S для элементов каждого столбца по формуле: $S_j = \sin(\sum_{i=0}^N a_{ij} + 3 * a_{23})$, где j – номер столбца.
15	1. Создать новый одномерный массив, каждый элемент которого равен среднему арифметическому элементов соответствующей строки данной матрицы; 2. Найти наименьший элемент 3 столбца; 3. Поменять местами наименьший элемент 3 столбца и элемент a_{23} .
16	1. Найти максимальный элемент 4 столбца; 2. Подсчитать сумму для элементов 2-ой строки по формуле: $S = 10 + \sum_{j=0}^M e^{(1-a_{2j})}$ 3. Заменить найденный максимальный элемент на значение S.
17	1. Заменить на главной диагонали все отрицательные элементы их абсолютными значениями; 2. Найти максимальный элемент 2-ой строки среди элементов, удовлетворяющих условию: $-1.2 \leq a_{2j} \leq 4.7$; 3. Поменять местами элемент из задача 2) с элементом a_{33}
18	1. Найти наибольший элемент матрицы; 2. В строке, содержащей максимальный элемент матрицы, подсчитать сумму для элементов по формуле: $S = \sum_{j=0}^M (\sin a_{kj} + \frac{1}{a_{kj}})$, где k – номер строки, содержащей максимальный элемент матрицы.
19	1. Найти наибольший элемент среди элементов четных столбцов матрицы; 2. Подсчитать произведение для элементов 4 столбца по формуле: $P = \prod_{i=1}^N (1 + \cos(1 - \frac{1}{a_{i4}})) + 8.3$
20	1. Найти минимум среди положительных элементов матрицы и его местоположение; 2. Подсчитать произведение для элементов 2-ой строки по формуле: $P = \prod_{j=0}^M (\frac{1}{a_{2j}} + 4.2)$ 3. Поменять местами найденный минимальный элементы и P.
21	1. Найти максимальный элемент матрицы среди элементов, расположенных левее 4 столбца; 2. Подсчитать сумму для элементов 2 столбцов по формуле: $S = \sum_{i=0}^N \frac{a_{i2} + a_{i2}^2}{a_{43}}$ 3. Поменять местами найденные максимальный элемент и значение S
22	1. Создать новый одномерный массив A, элементами которого будут

	<p>отрицательные элементы 3 столбца матрицы;</p> <p>2. Подсчитать их сумму по формуле: $S = \sum_{i=0}^K (i * a_i^3 + \sin a_i)$, где k – размер массива.</p> <p>3. Заменить элемент в 3 строке и 2 столбце на полученное значение S.</p>
23	<p>1. Найти сумму отрицательных элементов, лежащих под побочной диагональю;</p> <p>2. Найти минимальный элемент среди положительных элементов 2 столбца;</p> <p>3. Поменять местами минимальный элемент 2 столбца с элементом a_{24}.</p>
24	<p>1. Создать одномерный массив, элементы которого равны сумме элементов каждого столбца заданной матрицы;</p> <p>2. Найти сумму элементов массива по формуле: $S = \sum_{i=0}^M (5 + \frac{2 * i * a_i^2}{10})^3$</p>
25	<p>1. Найти минимальный элемент A среди элементов, лежащих на главной диагонали;</p> <p>2. Найти произведение A и минимальных элементов каждого столбца матрицы;</p> <p>3. Заменить a_{24} на полученное произведение.</p>
26	<p>1. Найти сумму отрицательных элементов матрицы, лежащих над главной диагональю;</p> <p>2. Найти максимальный элемент 3 строки;</p> <p>3. Заменить максимальный элемент 3 строки на полученное значение суммы.</p>
27	<p>1. Найти минимальный элемент среди отрицательных элементов матрицы и его местоположение;</p> <p>2. В строке, содержащей минимальный элемент, найти сумму элементов по формуле: $S = \sum_{j=0}^M (\sin a_{kj} + \cos(3 * a_{kj}))$, где k – номер строки, содержащей минимальный элемент;</p> <p>3. Поменять местами минимальный элемент и элемент a_{34}.</p>
28	<p>1. Создать новый одномерный массив, состоящий из произведений элементов матрицы по строкам;</p> <p>2. Подсчитать сумму элементов этого массива по формуле: $S = \sum_{i=0}^N 4.2 * (1 - a_i) * \cos a_i$</p>
29	<p>1. Найти максимум среди элементов матрицы, расположенных над побочной диагональю и его местоположение;</p> <p>2. Подсчитать количество положительных элементов во 2 строке и произведение этих элементов по формуле: $P = \prod_{j=0}^k \sin a_{2j}$, где k – количество положительных элементов.</p>
30	<p>1. Найти минимальный элемент на побочной диагонали.</p> <p>2. Найти сумму элементов 3 столбца по формуле: $S = \operatorname{ctga}_{32} + \sum_{i=0}^N e^{a_{i3}}$</p>
31	<p>1. Найти сумму элементов 3 столбца по формуле: $S = \sum_{i=0}^N e^{a_{i3}} * \frac{a_{i3}}{a_{43}}$</p> <p>2. Поменять максимальный элемент матрицы на значение S.</p>
32	<p>1. Создать новый одномерный массив из минимальных элементов каждой строки данной матрицы.</p>

	2. Вычислить произведение P всех элементов массива из задачи 1);
	3. Заменить минимальный элемент каждой строки матрицы на произведение P .

3. Написать программу, работающую с базой данных объектов.

Программа должна выполнять следующие действия:

1. Считывать базу данных (массив объектов) в оперативную память;
2. Сохранять базу данных (массив объектов) в файл в бинарном или текстовом виде;
3. Выводить на экран все объекты базы данных;
4. Редактировать свойства отдельного объекта;
5. Находить объекты по одному свойству;
6. Удалять или добавлять объекты;
7. Управление осуществлять с помощью текстового меню в консоли.

Для сборки проекта реализуется Makefile. Все структуры и объявления должны находиться в заголовочном файле, а реализация функций и реализация main в отдельных файлах с исходным кодом.

Объект базы данных реализуется как структура, поля структуры соответствуют варианту:

- a. Сад
- b. Зоопарк
- c. Бар
- d. Больница
- e. Почта
- f. Оптика
- g. Кафе
- h. Военная техника
- i. Стрелковое оружие
- j. Холодильник
- k. Магазин игровых консолей
- l. Питомник собак
- m. Youtube
- n. Магазин стройматериалов
- o. Кинопоиск
- p. Магазин косметики
- q. Спортивные матчи
- r. Ипподром
- s. Канцтовары
- t. Продуктовый магазин
- u. Склад мебели
- v. Магазин одежды
- w. Редакция журналов
- x. Книжный магазин
- y. Бд компании по найму сотрудников
- z. Магазин парфюмерии
- aa. Аптека
- bb. Авиабазы
- cc. Netflix
- dd. Страны
- ee. Spotify
- ff. Steam
- gg. Музей
- hh. Театр
- ii. Пекарня
- jj. Туроператор

кк. Кинотеатр

II. Магазин ноутбуков

2. Типовые темы групповых и/или индивидуальных творческих заданий/проектов

Групповые творческие задания (проекты):

1. Реализовать программу, управляющую одной или несколькими базами данных согласно групповой теме. Данные в разных БД связаны между собой.

Требования к программе:

1. Управление программой осуществляется с помощью консольного меню.
2. Пользователь может просматривать, добавлять, удалять и редактировать элементы в каждой базе данных.
3. Программа имеет тестовую функцию «прокрутки» времени. После вызова функции базы данных должны обновляться, как если бы прошло N часов реального времени. Например, для темы «больница» пациенты могут «выписаться» из больницы.

Требования к коду:

1. Для хранения набора объектов использовать `std::list` или `std::vector`;
2. Программа должна быть реализована в нескольких файлах с предоставленным для сборки `MakeFile`;
3. Все сложные типы данных реализуются через классы с перегрузкой операторов ввода и вывода при необходимости. Отдельные функции должны находиться или в классе, или в `namespace`;
4. Все функции и классы должны иметь описание в формате `Doxygen`.

Групповые темы:

1. Автоматизация учёта рабочего времени удалённых сотрудников.

Каждый сотрудник имеет как минимум следующие параметры: ФИО, количество рабочих часов в неделю, текущая задача (если назначена). Каждая задача имеет трудоёмкость и срок выполнения в сутках.

Программа должна позволять выполнять следующие действия:

- осуществлять управление списком задач и списком сотрудников;
- назначать сотруднику задачу;
- просматривать степень готовности задач;
- просматривать сотрудников и степень их занятости;
- изменять готовность задач в режиме «перемотки времени».

2. Система учёта оценок студентов на дистанционном обучении.

Необходимо создать программу, которая могла бы собирать информацию об оценках и посещаемости студента во время дистанционного обучения.

Каждый преподаватель имеет поля: ФИО, предмет, группа и данные для входа в систему.

3 режима работы программы:

1. Студент. Можно просмотреть свои оценки по предметам.
2. Преподаватель. Можно выбрать студента из группы и задать ему оценку по предмету.
3. Администратор. Может добавлять, изменять, удалять преподавателей и студентов.

3. Система прогнозирования эпидемии.
У вируса есть следующие поля: скорость заражения чел/сутки, продолжительность болезни в сутках, летальность чел/сутки. Пользователь может а) посмотреть информацию по всем текущим эпидемиям: количество заболевших, умерших, выздоровевших; б) добавить попытку противодействия вирусу – меру защиты, снижающую летальность или

распространения вируса; в) промотать время для отслеживания активности вируса.

Вирус считается побеждённым, если количество заболевших в день < 5 .

3. Типовые задания для контрольной работы

Тема: работа с массивами и функциями

1. Сделать функцию с 2 параметрами: указатель на 1 элемент массива и длина массива, которая выводит массив на экран. Ввести с клавиатуры массив из N элементов и вывести его на экран с помощью данной функции
2. Ввести с клавиатуры массив из N чисел и вывести на экран произведение элементов массива
3. Реализовать функцию, вычисляющую $n!$. Использовать функцию в `main`, n ввести с клавиатуры.
4. Написать функцию, вычисляющую площадь круга. В функцию передать радиус. В `main` радиус ввести с клавиатуры. Вывести на экран площадь круга с данным радиусом
5. Ввести с клавиатуры размеры 2-х катетов. Найти гипотенузу и вывести её на экран. Вычисление гипотенузы сделать отдельной функцией. Для вычисления корня (функция `sqrt`) можно использовать библиотеку `<cmath>`.
6. Реализовать функцию $f(x)=x^2$. Инициализировать в `main` массив на 5 чисел (можно захардкодить), вывести на экран в цикле результат функции f для каждого элемента массива.

Тема: работа с матрицами

1. Ввести с клавиатуры массив из N чисел и вывести на экран сумму элементов массива
2. Ввести с клавиатуры матрицу N на M и вывести её на экран
3. Ввести с клавиатуры массив из N чисел и вывести его на экран в обратном порядке
4. Ввести с клавиатуры матрицу 3 на 3. Найти её определитель, вывести его на экран
5. Реализовать функцию, выводящую матрицу на экран. В функцию передать матрицу и её размеры. В `main` использовать функцию для матрицы 3 на 3 (матрицу можно захардкодить)

Тема: работа с файлами

1. Читать из файла 1 строку с пробелами и вывести её на экран.
2. Дан файл с каким-то текстом. Определить размер файла в байтах и вывести его на экран
3. Читать из файла массив из 5 чисел (записаны через пробел), и вывести на экран сумму этих чисел
4. Ввести с клавиатуры строку и сохранить её в конец файла (не стирая предыдущие записи в файле)
5. Ввести с клавиатуры число и сохранить его в файл в бинарном виде
6. Читать из файла массив из 10 чисел (записаны через пробел), и вывести все чётные числа
7. В файле неизвестное количество строк. Читать все и вывести их на экран
8. Сохранить в файл массив из 3х чисел в бинарном виде. Массив можно захардкодить
9. Дан текстовый файл с каким-то текстом. Написать программу, записывающую в файл в любую строку с отступом в 5 символов (т.е. не в начало или конец файла, а переместиться на N символов)
10. Создать матрицу 5 на 5 (значения можно захардкодить или ввести с клавиатуры). Посчитать произведение всех чисел матрицы, записать результат в файл.
11. Читать из файла 10 чисел. Вывести на экран все положительные чётные числа (в файле могут быть любые числа, чётные, нечётные, положительные и отрицательные).
12. Создать программу, рисующую в файле квадратик. Размеры квадрата ввести с клавиатуры
13. В файле задано время в формате час, минута, секунда (через пробел). Реализовать: читать время из файла, ввести с клавиатуры кол-во секунд. Вычесть из считанного из файла времени секунды, вывести результат на экран в формате: `hh:mm:ss`
14. Реализовать подсчёт числа секунд между двумя моментами времени, лежащими в пределах одних суток. Время ввести с клавиатуры (в формате час, минута, секунда). Результат вывести

в файл

Тема: работа с классами

1. Создать класс, хранящий массив из N элементов. Массив инициализировать в конструкторе (использовать динамическую память). Создать в main объект данного класса
 2. Создать любой класс. Создать в main массив из N объектов данного класса и ввести его с клавиатуры
 3. Ввести с клавиатуры массив из N чисел и вывести на экран произведение элементов массива
 4. Создать класс "Студент" с конструктором с параметрами; в main создать объект данного класса и вывести его поля на экран
 5. Создать базовый класс "Фигура" с чисто виртуальной функцией "периметр". Создать класс-наследник "Прямоугольник". Создать в main объект класса "Прямоугольник" и вывести на экран его периметр
 6. Создать класс с 1 приватным полем типа int. Перегрузить для данного класса оператор + как член класса. Сложить в main 2 объекта данного класса и вывести на экран результат
 7. Создать класс с 1 приватным полем типа int. Перегрузить для данного класса оператор + вне класса. Сложить в main 2 объекта данного класса и вывести на экран результат
 8. Создать класс с 2 приватными полями. Перегрузить для данного класса оператор вывода <<. Создать в main объект данного класса и вывести его на экран
 9. Создать класс с 2 приватными полями. Перегрузить для данного класса оператор ввода >>. Создать в main объект данного класса, ввести его поля с клавиатуры
 10. Создать базовый класс "Человек" с полями "возраст" и "имя". Создать класс-наследник "Студент" с новым полем "группа". Все поля сделать приватными. Создать в main объект класса "Студент" и вывести все его поля на экран
 11. Создать класс из 3-х полей: строка, целое число и вещественное число. Ввести с клавиатуры массив из 3-х таких классов. Вывести его на экран.
 12. Создать класс с 1 приватным полем типа int. Перегрузить для данного класса оператор ++ как член класса. Создать в main объект данного класса, использовать ++ и вывести на экран результат
 13. Создать класс с 1 приватным полем типа int. Перегрузить для данного класса оператор == как член класса. Сравнить в main 2 объекта данного класса и вывести на экран результат
-