



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение
высшего образования

«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФГБОУ ВО «ИГУ»

Кафедра радиофизики и радиоэлектроники



Декан Буднев Н.М.

«02» апреля 2025 г.

Рабочая программа дисциплины

Наименование дисциплины **Б1.О.14.01 Алгоритмы и основы программирования**

Направление подготовки **09.03.02 Информационные системы и технологии**

Направленность (профиль) подготовки **Электронный и программный инжиниринг**

Квалификация выпускника **бакалавр**

Форма обучения **очная**

Согласовано с УМК физического факультета

Протокол № 49 от «26» марта 2025 г.

Председатель Буднев Н.М.

Рекомендовано кафедрой радиофизики и радиоэлектроники:

Протокол № 8 от «24» февраля 2025 г.

И.О. зав. кафедрой Колесник С.Н.

Иркутск 2025 г.

Содержание

I. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ	3
II. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО	3
III. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ	3
IV. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ	11
4.1. Содержание дисциплины, структурированное по темам, с указанием видов учебных занятий и отведенного на них количества академических часов	11
4.2. План внеаудиторной самостоятельной работы обучающихся по дисциплине	12
4.3. Содержание учебного материала	13
4.3.1. Перечень семинарских, практических занятий и лабораторных работ	13
4.3.2. Перечень тем (вопросов), выносимых на самостоятельное изучение студентами в рамках самостоятельной работы (СРС)	13
4.4. Методические указания по организации самостоятельной работы студентов	14
4.5. Примерная тематика курсовых работ	14
V. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	14
VI. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	14
6.1. Учебно-лабораторное оборудование:	14
6.2. Программное обеспечение:	15
6.3. Технические и электронные средства:	15
VII. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ	15
VIII. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ	15

І. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Цели: научить базовому программированию и реализации алгоритмов на языке C++

Задачи: изучение основ языка C++, изучение базовых программных алгоритмов.

ІІ. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Учебная дисциплина (модуль) Б1.О.14.01 Алгоритмы и основы программирования относится к обязательной части программы.

Для изучения данной учебной дисциплины (модуля) необходимы знания, умения и навыки, формируемые предшествующими дисциплинами: школьные курсы математики, физики и информатики.

Перечень последующих учебных дисциплин, для которых необходимы знания, умения и навыки, формируемые данной учебной дисциплиной: численные методы и программирование, основы робототехники, теория и обработка информации, обработка данных на языке Python, компьютерные вычислительные сети, Web-программирование, технологии искусственного интеллекта.

ІІІ. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс освоения дисциплины направлен на формирование компетенций в соответствии с КРМ ТОП ИТ и ОПОП ВО по направлению подготовки **09.03.02 Информационные системы и технологии**.

Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Компетенция	Индикаторы компетенций	Результаты обучения
<i>ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий</i>	<i>ИДКОПК-6.1. Выполняет формализацию и алгоритмизацию поставленных задач для разработки программного кода</i>	<i>Знать: алгоритмы решения типичных задач, области и способы их применения. Уметь: использовать методы и приемы формализации и алгоритмизации поставленных задач; применять алгоритмы решения типовых задач в соответствующих областях. Владеть: базовыми навыками программирования</i>
	<i>ИДКОПК-6.2. Разрабатывает программный код с использованием языков программирования</i>	<i>Знать: синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования. Уметь: применять выбранные языки программирования для написания программного кода. Владеть: технологиями программирования.</i>

<i>ОПК6.ППК-Р1. Способен разрабатывать и отлаживать программный код</i>	<i>ОПК6.ППК-Р1.1. Выполняет формализацию и алгоритмизацию поставленных задач для разработки программного кода</i>	<i>ОПК6.ППК-Р1.1. 3-1. Знает алгоритмы решения типичных задач, области и способы их применения.</i> <i>ОПК6.ППК-Р1.1. 3-2. Знает нотации и программное обеспечение для графического отображения алгоритмов.</i> <i>ОПК6.ППК-Р1.1. 3-3. Знает методы и приемы алгоритмизации поставленных задач.</i> <i>ОПК6.ППК-Р1.1. У-1. Умеет использовать методы и приемы формализации и алгоритмизации поставленных задач.</i> <i>ОПК6.ППК-Р1.1. У-2. Умеет применять алгоритмы решения типовых задач в соответствующих областях.</i>
	<i>ОПК6.ППК-Р1.2. Формализует задачу ИТ отрасли в язык естественнонаучных дисциплин</i>	<i>ОПК6.ППК-Р1.2. 3-1. Знает основные разделы математики, применяемые для анализа и моделирования непрерывных процессов и дискретных систем в прикладных задачах.</i> <i>ОПК6.ППК-Р1.2. У-1. Умеет выбирать адекватный математический аппарат для формализованного описания сущностей и отношений на основе ТЗ и бизнес-требований.</i> <i>ОПК6.ППК-Р1.2. В-1. Владеет навыками описания задач предметной области в виде формальных математических моделей, пригодных для последующей алгоритмизации и программной реализации.</i>
	<i>ОПК6.ППК-Р1.3. Осуществляет обоснованный выбор методов и алгоритмов для программной реализации формальной математической модели</i>	<i>ОПК6.ППК-Р1.3. 3-1. Знает основные классы методов программной реализации моделей и критерии выбора алгоритмов.</i> <i>ОПК6.ППК-Р1.3. У-1. Умеет проводить сравнительный анализ и обоснование выбора алгоритмов для программной реализации модели.</i> <i>ОПК6.ППК-Р1.3. В-1. Владеет методами адаптации методов и алгоритмов под специфику задачи.</i>

		ОПК6.ППК-Р1.3. В-2. Владеет навыками оценки эффективности выбранных алгоритмов.
	ОПК6.ППК-Р1.4. Разрабатывает программный код с использованием языков программирования	ОПК6.ППК-Р1.4. 3-1. Знает синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования. ОПК6.ППК-Р1.4. 3-2. Знает методологии разработки компьютерного программного обеспечения. ОПК6.ППК-Р1.4. 3-3. Знает технологии программирования. ОПК6.ППК-Р1.4. У-1. Умеет применять выбранные языки программирования для написания программного кода. ОПК6.ППК-Р1.4. У-2. Умеет использовать выбранную среду программирования. ОПК6.ППК-Р1.4. У-3. Умеет использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода.
	ОПК6.ППК-Р1.5. Оформляет программный код в соответствии с установленными требованиями	ОПК6.ППК-Р1.5. 3-1. Знает нормативно-технические документы (стандарты и регламенты), определяющие требования к оформлению программного кода. ОПК6.ППК-Р1.5. 3-2. Знает основные стандарты оформления технической документации на компьютерное программное обеспечение. ОПК6.ППК-Р1.5. У-1. Умеет применять заданные стандарты и шаблоны для составления и оформления технической документации ОПК6.ППК-Р1.5. У-2. Умеет применять нормативно-технические документы (стандарты и регламенты), определяющие требования к оформлению программного кода. ОПК6.ППК-Р1.5. У-3. Умеет применять инструментарий для

		создания и актуализации исходных текстов программ.
	ОПК6.ППК-Р1.7. Проверяет и отлаживает программный код	ОПК6.ППК-Р1.7. 3-1. Знает методы и приемы отладки программного кода. ОПК6.ППК-Р1.7. 3-2. Знает типы и форматы сообщений об ошибках, предупреждений ОПК6.ППК-Р1.7. 3-3. Знает способы использования технологических журналов, форматы и типы записей журналов. ОПК6.ППК-Р1.7. У-1. Умеет выявлять ошибки в программном коде. ОПК6.ППК-Р1.7. У-2. Умеет отлаживать программный код на уровне программных модулей. ОПК6.ППК-Р1.7. У-3. Умеет отлаживать программный код на уровне межмодульных взаимодействий и взаимодействий с окружением.
ОПК6.ППК-Р2. Способен проверять работоспособность и проводить рефакторинг кода программного обеспечения	ОПК6.ППК-Р2.1. Разрабатывает тестовые наборы данных для проверки работоспособности компьютерного программного обеспечения	ОПК6.ППК-Р2.1. 3-1. Знает методы создания и документирования контрольных примеров и тестовых наборов данных. ОПК6.ППК-Р2.1. 3-2. Знает требования к структуре и форматам хранения тестовых наборов данных. ОПК6.ППК-Р2.1. 3-3. Знает правила, алгоритмы и технологии создания тестовых наборов данных. ОПК6.ППК-Р2.1. У-1. Умеет разрабатывать и оформлять контрольные примеры для проверки работоспособности компьютерного программного обеспечения. ОПК6.ППК-Р2.1. У-2. Умеет готовить тестовые наборы данных в соответствии с выбранной методикой тестирования компьютерного программного обеспечения.

	ОПК6.ППК-Р2.2. Проверяет работоспособность компьютерного программного обеспечения	ОПК6.ППК-Р2.2. 3-1. Знает методы и средства проверки работоспособности компьютерного программного обеспечения. ОПК6.ППК-Р2.2. 3-2. Знает государственные стандарты испытания автоматизированных систем. ОПК6.ППК-Р2.2. 3-3. Знает руководящие документы по стандартизации требований к документам автоматизированных систем. ОПК6.ППК-Р2.2. У-1. Умеет применять методы и средства проверки работоспособности компьютерного программного обеспечения. ОПК6.ППК-Р2.2. У-2. Умеет интерпретировать диагностические данные проверки работоспособности компьютерного программного обеспечения ОПК6.ППК-Р2.2. У-3. Умеет анализировать значения полученных характеристик компьютерного программного обеспечения.
	ОПК6.ППК-Р2.3. Исправляет дефекты программного кода, зафиксированные в базе данных дефектов	ОПК6.ППК-Р2.3. 3-1. Знает типичные ошибки, возникающие при разработке компьютерного программного обеспечения, методы их диагностики и исправления. ОПК6.ППК-Р2.3. 3-1. Знает методы и приемы отладки программного кода ОПК6.ППК-Р2.3. У-1. Умеет воспроизводить дефекты программного кода, зафиксированные в базе данных дефектов. ОПК6.ППК-Р2.3. У-2. Умеет выяснять причины возникновения дефектов программного кода. ОПК6.ППК-Р2.3. У-3. Умеет вносить изменений в программный код для устранения выявленных дефектов.

	ОПК6.ППК-Р2.4. Выполняет рефакторинг и инспекцию программного кода	ОПК6.ППК-Р2.4. 3-1. Знает методы и средства рефакторинга и инспекции программного кода. ОПК6.ППК-Р2.4. 3-1. Знает нормативно-технические документы (стандарты и регламенты), регламентирующие требования к программному коду, порядок отражения изменений в системе управления версиями, порядок отражения результатов рефакторинга, оптимизации и инспекции в коллективной базе знаний. ОПК6.ППК-Р2.4. У-1. Умеет анализировать программный код на соответствие требованиям по читаемости и производительности. ОПК6.ППК-Р2.4. У-2. Умеет проводить инспекцию программного кода для поиска не обнаруженных на ранних стадиях разработки компьютерного программного обеспечения ошибок и критических мест. ОПК6.ППК-Р2.4. У-3. Умеет применять методы и средства рефакторинга и инспекции программного кода. ОПК6.ППК-Р2.4. У-4. Умеет публиковать результаты рефакторинга и инспекции в коллективной базе знаний.
ОПК6.ППК-Р7. Способен применять искусственный интеллект (ИИ) для генерации и отладки программного кода	ОПК6.ППК-Р7.1. Применяет ИИ-инструменты для генерации программного кода.	ОПК6.ППК-Р7.1. 3-1. Знает принципы работы современных генеративных ИИ-моделей для генерации кода. ОПК6.ППК-Р7.1. 3-2. Знает ограничения и риски использования ИИ-генерации (безопасность, качество кода, лицензирование). ОПК6.ППК-Р7.1. У-1. Умеет формулировать корректные текстовые запросы (промты) для генерации кода. ОПК6.ППК-Р7.1. У-2. Умеет интегрировать ИИ-

		инструменты в среду разработки.
	ОПК6.ППК-Р7.2. Использует ИИ для анализа и отладки кода.	ОПК6.ППК-Р7.2. 3-1. Знает методы ИИ-анализа кода. ОПК6.ППК-Р7.2. 3-2. Знает форматы и инструменты для автоматизированного тестирования с ИИ. ОПК6.ППК-Р7.2. У-1. Умеет настраивать ИИ-инструменты для поиска уязвимостей. ОПК6.ППК-Р7.2. У-2. Умеет интерпретировать рекомендации ИИ по исправлению кода.
	ОПК6.ППК-Р7.3. Оптимизирует код с помощью ИИ	ОПК6.ППК-Р7.3. 3-1. Знает методы ИИ-оптимизации. ОПК6.ППК-Р7.3. 3-2. Знает критерии качества кода, применяемые ИИ-системами. ОПК6.ППК-Р7.3. У-1. Умеет использовать ИИ для рефакторинга. ОПК6.ППК-Р7.3. У-2. Умеет проверять корректность оптимизаций, предложенных ИИ.
ОПК6.ИТК-4. Способен оптимизировать производительность программного обеспечения	ОПК6.ИТК-4.1. Выполняет мониторинг производительности программного обеспечения.	ОПК6.ИТК-4.1. 3-1. Знает методы и средства мониторинга производительности компьютерного программного обеспечения. ОПК6.ИТК-4.1. 3-2. Знает метрики производительности программного обеспечения. ОПК6.ИТК-4.1. 3-3. Знает современные инструменты мониторинга производительности программного обеспечения. ОПК6.ИТК-4.1. У-1. Умеет применять методы и средства мониторинга производительности компьютерного программного обеспечения. ОПК6.ИТК-4.1. У-2. Умеет интерпретировать диагностические данные мониторинга производительности

		компьютерного программного обеспечения. ОПК6.ИТК-4.1. У-3. Умеет определять "узкие места" программного обеспечения
	ОПК6.ИТК-4.2. Выполняет оптимизацию программного кода.	ОПК6.ИТК-4.2. 3-1. Знает методы и средства оптимизации производительности компьютерного программного обеспечения. ОПК6.ИТК-4.2. 3-2. Знает современные инструменты оптимизации производительности программного обеспечения. ОПК6.ИТК-4.2. У-1. Умеет оптимизировать программный код с использованием специализированных программных средств. ОПК6.ИТК-4.2. У-2. Умеет вырабатывать варианты оптимизации производительности компьютерного программного обеспечения. ОПК6.ИТК-4.2. У-3. Умеет проводить оценку и обоснование рекомендуемых решений по оптимизации производительности.

IV. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ

Объем дисциплины составляет 6зачетных единиц, 216часов.

Форма промежуточной аттестации: зачет с оценкой

4.1. Содержание дисциплины, структурированное по темам, с указанием видов учебных занятий и отведенного на них количества академических часов

№ п/п	Раздел дисциплины/тема	Семестр	Всего часов	Из них практическая подготовка обучающихся	Виды учебной работы, включая самостоятельную работу обучающихся, практическую подготовку и трудоемкость (в часах)				Форма текущего контроля успеваемости
					Контактная работа преподавателя с обучающимися			Самостоятельная работа	
					Лекция	Лабораторное занятие	Консультация		
1	2	3	4	5	6	7	8	9	10
1	Тема 1. Основы программирования	1	21,1		2	4	0,1	15	Сдача лабораторных заданий на ПК
2	Тема 2. Переменные, ввод/вывод	1	33,1		6	12	0,1	15	
3	Тема 3. Работа с файлами	1	33,1		6	12	0,1	15	
4	Тема 4. Условия и циклы	1	33,2		6	12	0,2	15	
5	Тема 5. Массивы и указатели	1	33,2		6	12	0,2	15	
6	Тема 6. Алгоритмы	1	33,2		6	12	0,2	15	
7	Тема 7. Оконные приложения	1	21,1		2	4	0,1	15	

4.2. План внеаудиторной самостоятельной работы обучающихся по дисциплине

Семестр	Название раздела, темы	Самостоятельная работа обучающихся			Оценочное средство	Учебно-методическое обеспечение самостоятельной работы
		Вид самостоятельной работы	Сроки выполнения	Трудоемкость (час.)		
1	Тема 1. Основы программирования	Работа с учебником, справочной литературой, первоисточниками, конспектом	1-2 нед.	15	Результат выполнения лабораторных заданий	
2	Тема 2. Переменные, ввод/вывод		3-4 нед.	15		
3	Тема 3. Работа с файлами		5-6 нед.	15		
4	Тема 4. Условия и циклы		7-9 нед.	15		
5	Тема 5. Массивы и указатели		10-12 нед.	15		
6	Тема 6. Алгоритмы		13-15 нед.	15		
7	Тема 7. Оконные приложения		16-17 нед.	15		
Общий объем самостоятельной работы по дисциплине (час)				105		

4.3.Содержание учебного материала

Тема 1.

Основы программирования на языке C++: IDE, проект, исходный код, компиляция, отладка, исполняемый файл, первое приложение.

Тема 2.

Текстовый ввод/вывод, форматирование вывода. Переменные и типы данных.

Тема 3.

Файловый ввод/вывод. Форматы файлов.

Тема 4.

Условия и циклы. Функции. Рекурсия.

Тема 5.

Указатели. Массивы. Передача массивов в функции.

Тема 6.

Примеры алгоритмы с использованием циклов, условий и массивов.

Тема 7.

Оконные приложения. Вывод графики.

4.3.1. Перечень семинарских, практических занятий и лабораторных работ

№ п/п	№ раздела и темы	Наименование семинаров, практических и лабораторных работ	Трудоемкость (час.)		Оценочные средства	Формируемые компетенции
			Всего часов	Из них практическая подготовка		
1	2	3	4	5	6	7
1	Тема 1-7	Лабораторные задания на ПК	68		Демонстрация работы программы и кода	

4.3.2. Перечень тем (вопросов), выносимых на самостоятельное изучение студентами в рамках самостоятельной работы (СРС)

№ п/п	Тема	Задание	Формируемая компетенция	ИДК
1	2	3	4	5
1	Тема 1-7	Осмысление материала лекций, воспроизведение и анализ лекционного кода	ОПК-6. ОПК6.ППК-Р1. ОПК6.ППК-Р2. ОПК6.ППК-Р7. ОПК6.ИТК-4.	ИДК _{ОПК-6.1.} ИДК _{ОПК-6.2.} ОПК6.ППК-Р1.1. ОПК6.ППК-Р1.2. ОПК6.ППК-Р1.3. ОПК6.ППК-Р1.4. ОПК6.ППК-Р1.5. ОПК6.ППК-Р1.7 ОПК6.ППК-

				<i>P2.1. ОПК6.ППК- P2.2. ОПК6.ППК- P2.3. ОПК6.ППК- P2.4. ОПК6.ППК- P7.1. ОПК6.ППК- P7.2. ОПК6.ППК- P7.3. ОПК6.ИТК-4.1. ОПК6.ИТК-4.2.</i>
--	--	--	--	--

4.4. Методические указания по организации самостоятельной работы студентов

Для успешного закрепления навыков программирования рекомендуется студентам во время самостоятельной работы воспроизводить и анализировать коды, изучаемые на лекции, модифицировать их по своему усмотрению.

4.5. Примерная тематика курсовых работ

Выполнение курсовых работ не предусмотрено учебным планом

V.УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

а) перечень литературы

1. Технологии программирования. Технология программирования на C/C++: учебное пособие, Кривцов А.Н., Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2021, 274 с. (<https://e.lanbook.com/book/279680>)
2. Алгоритмизация и программирование. Основы программирования на C/C++: учебное пособие, Кривцов А.Н., Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2020, 202 с. (<https://e.lanbook.com/book/180057>)

б) периодические издания

в) список авторских методических разработок

г) базы данных, информационно-справочные и поисковые системы

1. Поисковые системы Google, Yandex.
2. Электронные ресурсы доступные по логину и паролю, предоставляемые Научной библиотекой ИГУ.

VI.МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1.Учебно-лабораторное оборудование:

Класс ЭВМ, аудитория 323А, оснащенная вычислительной техникой, специальным ПО и свободным доступом в сеть Internet.

6.2. Программное обеспечение:

Microsoft Visual Studio 2019/2022 Community

6.3. Технические и электронные средства:

Мультимедийный проектор, экран (по необходимости), меловая или маркерная доска.

VII. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

На лекциях происходит написание и анализ кода, а также разбор результатов его выполнения. Лабораторные работы проводятся с использованием ПЭВМ с последующей защитой, демонстрацией работы программ и их кода с пояснениями.

Применяемые интерактивные формы обучения:

- Гейм-модификация программирования (Решение алгоритмических задач в игре)
- IT-квесты (Поиск и устранение ошибок в коде, сборка фрагментов)

VIII. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Текущий контроль осуществляется по результатам работы студента на лабораторных работах, качеству написанного кода и данных к нему пояснений.

Промежуточная аттестация в виде зачета с оценкой принимается на основании выполнения студентом всех лабораторных заданий на программирование по всем темам курса. Примерный перечень лабораторных заданий и вариантов к ним:

Задание 1

Вывести в файл Out.txt первые 10 членов ряда и их сумму:

- 1) $1+2+3+4+\dots$
- 2) $1-2+3-4+\dots$
- 3) $1+1/2+1/3+1/4+\dots$
- 4) $1-1/2+1/3-1/4+\dots$

Задание 2

- 1) Вывести на экран 20 случайных целых чисел в диапазоне $[-10 \text{ до } 10]$ и их среднее
- 2) Вывести на экран 20 случайных целых чисел в диапазоне $[10 \text{ до } 20]$ и их среднее
- 3) Вывести на экран 20 случайных целых чисел в диапазоне $[-20 \text{ до } -10]$ и их среднее
- 4) Вывести на экран 20 случайных целых чисел в диапазоне $[-10 \text{ до } 20]$ и их среднее

Задание 3 (создание своей функции)

1) Создать функцию, возвращающую факториал числа, и вывести на экран результат вычисления $6!/(3!+4!)$

2) Создать функции, возвращающие гиперболические синус и косинус, и вывести таблицу их значений с аргументами от -5 до 5 и шагом 0.5 (как в задании 1)

3) Создать функцию, возвращающую гамма-функцию (использовать определение Эйлера и учитывать 1000000 членов ряда), и вывести на экран ее значения при аргументе от 0.5 до 6.0 с шагом 0.2

Задание 4 (массивы)

- 1) Считать с клавиатуры 5 целых чисел и вывести их на экран в обратном порядке

2) Сгенерировать 1000 раз случайное целое число в диапазоне [1..10] и вывести на экран количество

выпадений каждого из возможных

3) Сгенерировать 1000 раз пару случайных целых чисел в диапазоне [1..6] (игральный кубик) и вывести на экран количество каждой выпавшей случайной суммы ([2..12])

Пример тестовых заданий для проверки сформированности компетенции ОПК-6.

- 1) Название программы, преобразующий текст исходной программы в исполняемый код?
 - a) текстовый редактор
 - b) компилятор
 - c) трассер
 - d) препроцессор
- 2) Название процесса обнаружения и устранения ошибок в программном коде?
 - a) отладка
 - b) компиляция
 - c) линковка
 - d) трассировка
- 3) Название минимальной единицы логической информации?
 - a) бод
 - b) бит
 - c) байт
 - d) слово
- 4) Количество всевозможных значений, кодируемых одним байтом?
 - a) 2
 - b) 64
 - c) 256
 - d) 65536
- 5) Название переменной языка C++, содержащего адрес другой переменной?
 - a) ссылка
 - b) массив
 - c) структура
 - d) указатель
- 6) Как называется совокупность точно заданных правил решения некоторого класса задач или набор инструкций, описывающих порядок действий исполнителя для решения определённой задачи?
 - a) алгоритм
 - b) последовательность
 - c) компиляция
 - d) отладка

Разработчик:



доцент, Безлер И.В.

Программа составлена в соответствии с требованиями КРМ ТОП ИТ и ФГОС ВО по направлению и профилю подготовки **09.03.02 Информационные системы и технологии**.

Программа рассмотрена на заседании кафедры радиофизики и радиоэлектроники
«24» февраля 2025 г. протокол № 8

И.О. зав. кафедрой _____ Колесник С.Н.

Настоящая программа, не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.