
УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«25» мая 2026 г.
Протокол № 2

**Рабочая программа дисциплины (модуля)
«Язык программирования C++. Часть 2»**

Направление подготовки: 02.03.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Математика и компьютерные науки

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Срок освоения программы: 4 года

Год набора: 2026

**Москва
2026**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения	5
3. Тематический план	7
4. Содержание дисциплины (модуля)	7
5. Учебно-методическое обеспечение	8
6. Материально-техническое обеспечение	8
7. Методические и оценочные материалы	10

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Язык программирования C++. Часть 2» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по направлению 02.03.01 «Математика и компьютерные науки», профиль «Математика и компьютерные науки», утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Язык программирования C++. Часть 2» важно для освоения эффективного программирования системного и прикладного уровня, а также для понимания принципов работы с памятью и объектно-ориентированного подхода. Это позволяет создавать производительные и надежные приложения, востребованные в различных областях, от игр до встроенных систем.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 «Математика и компьютерные науки», профиль «Математика и компьютерные науки» и входит в обязательную часть Блока 1, как дисциплина по выбору.

Дисциплина (модуль) изучается на 1 курсе в 1 семестре.

Цель изучения дисциплины (модуля): формирование навыков эффективного программирования на языке C++ для создания высокопроизводительных и надежных программных решений.

Задачи изучения дисциплины (модуля):

- изучить основные концепции и модели облачных вычислений, включая сервисы IaaS, PaaS и SaaS;
- освоить принципы проектирования и развертывания облачных архитектур с учетом безопасности и масштабируемости;
- научиться применять инструменты и платформы облачных провайдеров для решения практических задач в разработке программного обеспечения.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- основы синтаксиса языка C++;
- фундаментальные типы данных;
- принципы работы с указателями и ссылками;
- механизмы работы с памятью;
- основы объектно-ориентированного и обобщённого программирования;
- стандартную библиотеку C++;
- принципы современного программирования на C++;
- методы модульного построения программ.

уметь:

- проектировать и реализовывать программы на языке C++;
- применять принципы модульности при разработке программ;
- использовать системы сборки для организации многофайловых проектов и управления зависимостями;
- применять возможности современных стандартов C++ для повышения выразительности и производительности кода.

владеть:

- навыками разработки и сопровождения проектов на языке C++;
- навыками выбора подходов и инструментов для решения различных задач;
- навыками анализа и отладки кода;

— практиками повышения качества программного обеспечения.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
ОПК-6.	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1.	Знает алгоритмы разработки, компьютерные программы, а также алгоритмы вычислительной математики
		ОПК-6.2.	Умеет разрабатывать математические программные продукты и комплексы с использованием современных технологий программирования
		ОПК-6.3.	Имеет практический опыт разработки интеллектуальных информационных систем для визуализации результатов исследований
ПК-3.	Способен использовать методы математического и алгоритмического моделирования при решении теоретических и прикладных задач	ПК-3.1.	Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач
		ПК-3.2.	Умеет разрабатывать и применять математические модели и алгоритмы для решения различных задач, анализируя полученные результаты
		ПК-3.3.	Имеет практический опыт использования методов математического и алгоритмического моделирования в реальных проектах или исследованиях
ПК-4.	Способен под руководством специалиста более высокой категории разрабатывать программное обеспечение для решения прикладных задач в сфере	ПК-4.1.	Знает основные принципы разработки программного обеспечения и методы решения прикладных задач
		ПК-4.2.	Умеет применять языки программирования и инструменты разработки для создания программного обеспечения под руководством более опытного специалиста

		ПК-4.3.	Имеет практический опыт участия в проектах по разработке программного обеспечения, работая в команде под руководством специалиста более высокой категории
--	--	---------	---

3. Тематический план

№п/ п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контр оль	Самостоятел ьная работа	
		Лекции	Семинары			
1	Основы программирования	4	8		12	Домашнее задание
2	Работа с памятью	6	8	4	20	Домашнее задание Контрольная работа
3	Модульное и обобщенное программирование	6	10	2	20	Домашнее задание Контрольная работа
4	Основы ООП	14	22	6	44	Домашнее задание Контрольная работа
	Зачет с оценкой			4		
Итого:		30	48	16	96	
Объем дисциплины (модуля) (в ак. ч.)		190				
Объем дисциплины (модуля) (в зач. ед.)		5				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Основы программирования	Структура программы. Типы данных. Переменные. Условия и циклы. Область действия и область видимости
2	Работа с памятью	Массивы. Функции-1. Указатели. Функции-2 Ссылки. Константность
3	Модульное и обобщенное программирование	Приведения типов в стиле C++ Шаблоны функций Модульное программирование
4	Основы ООП	Структуры Инкапсуляция, методы Конструкторы Перегрузка операций Шаблоны структур. Псевдонимы типов Функциональные объекты Новые стандарты C++

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Мэйерс, С. Эффективное использование C++. 55 верных способов улучшить структуру и код ваших программ : практическое руководство / С. Мэйерс ; пер. с англ. Н. Мухина. - 4-е изд. - Москва : ДМК Пресс, 2023. - 301 с. - ISBN 978-5-89818-324-0. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2102611>.

2. Уильямс, Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ : практическое руководство / Э. Уильямс ; пер. с англ. А. А. Слинкина. - 2-е изд. - Москва : ДМК Пресс, 2023. - 673 с. - ISBN 978-5-89818-319-6. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2102606>.

3. Павловская, Т. А. C/C++. Процедурное и объектно-ориентированное программирование : учебник для вузов / Т. А. Павловская. - Санкт-Петербург : Питер, 2021. - 496 с. - (Стандарт 3-го поколения), (Серия «Учебник для вузов»). - ISBN 978-5-4461-9722-4. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1857040>.

Дополнительная литература:

1. Павловская, Т. А. C/C++. Структурное и объектно-ориентированное программирование : практикум / Т. А. Павловская, Ю. А. Щупак. - Санкт-Петербург : Питер, 2021. - 352 с. - (Серия «Учебное пособие»). - ISBN 978-5-4461-9799-6. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1857042>.

2. Дэвидсон, Д. Красивый C++: 30 главных правил чистого, безопасного и быстрого кода : практическое руководство / Д. Дэвидсон, К. Грегори. - Санкт-Петербург : Питер, 2023. - 368 с. - (Для профессионалов). - ISBN 978-5-4461-2272-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2123398>.

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:
— столами и стульями;

- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		

КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Язык программирования C++. Часть 2» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, домашние задания, контрольные работы, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал, использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Контрольная работа – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы – получить специальные знания по одной или нескольким темам дисциплины и продемонстрировать навыки их практического применения.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов, планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Язык программирования C++. Часть 2»

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме *зачета с оценкой*, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
9	Отлично	Зачтено	
8	Отлично	Зачтено	
7	Хорошо	Зачтено	Студент обладает знаниями предмета

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
6	Хорошо	Зачтено	почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Язык программирования C++. Часть 2» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашнее задание	40%	6	Набор задач по темам недели
Контрольная работа	30%	3	Письменная работа с набором задач, которые нужно решить за ограниченное время
Зачет с оценкой	30%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

Формула расчёта итоговой оценки по дисциплине (модулю) «Язык программирования C++. Часть 2»: $\langle 0,4 \times \text{среднее за домашние задания} + 0,3 \times \text{среднее за контрольные работы} + 0,3 \times \text{зачет с оценкой} \rangle$.

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

1. Напишите программу, которая запрашивает у пользователя два целых числа и выводит их сумму, разность, произведение и частное. Обеспечьте проверку деления на ноль.
2. Напишите программу, которая определяет, является ли введённое число чётным или нечётным, а также положительным, отрицательным или нулём.
3. Напишите программу, которая по номеру месяца (целое число от 1 до 12) выводит количество дней в нём. Используйте оператор `switch`.
4. Напишите программу, которая выводит таблицу умножения от 1 до 10 в виде таблицы, используя вложенные циклы `for`.
5. Напишите функцию `int factorial(int n)`, вычисляющую факториал натурального числа с помощью цикла. Протестируйте её работу в функции `main()`.
6. Напишите программу, которая заполняет массив из 10 элементов случайными числами в диапазоне от 1 до 100 и находит максимальный и минимальный элементы массива.
7. Напишите функцию `bool isPrime(int n)`, которая возвращает `true`, если число является простым, и `false` — в противном случае. Протестируйте функцию для чисел от 2 до 30.
8. Напишите программу, которая переворачивает строку (массив символов), введённую пользователем, без использования стандартных библиотечных функций (`strlen`, `strcpy` и т.п.). Используйте указатели для обхода строки.
9. Создайте структуру `Student`, содержащую поля: имя (строка), возраст (целое число), средний балл (вещественное число). Заполните массив из трёх студентов и выведите на экран данные тех, у кого средний балл выше 4.0.
10. Напишите класс `Rectangle`, который содержит приватные поля `length` и `width`. Реализуйте публичные методы для:
 - Установки значений сторон (с проверкой на положительность),
 - Вычисления площади,
 - Вычисления периметра.Протестируйте работу класса в функции `main()`.

Примерные задания для контрольной работы

Контрольная работа №1

1. Даны три целых числа. Найдите и выведите наибольшее из них, используя условные операторы.
2. Напишите программу, которая выводит все чётные числа от 1 до 50, каждое с новой строки.
3. Напишите функцию `double power(double base, int exp)`, которая возводит число `base` в целую степень `exp` (включая отрицательные степени). Использовать стандартную функцию `pow` запрещено.

4. Объявите массив из 5 целых чисел, заполните его с клавиатуры и найдите сумму всех положительных элементов.

Контрольная работа №2

1. Напишите программу, которая запрашивает у пользователя год и определяет, является ли он високосным. Високосным считается год, который делится на 4, но не делится на 100, либо делится на 400.

2. Напишите программу, которая проверяет, является ли введенная пользователем строка палиндромом (читается одинаково слева направо и справа налево). Строка состоит из строчных латинских букв, длина — не более 100 символов.

3. Напишите функцию `void sortArray(int arr[], int n)`, которая сортирует массив целых чисел по возрастанию методом пузырька. Протестируйте её работу: заполните массив, выведите до и после сортировки.

4. Создайте класс `Circle`, который имеет:

- Приватное поле `radius` (вещественное число),
- Конструктор по умолчанию (устанавливает радиус = 1.0),
- Конструктор с параметром,
- Методы для вычисления площади и длины окружности.

В функции `main()` создайте два объекта класса `Circle` и выведите их характеристики.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1	Укажите тип данных в C++, используемый для хранения целых чисел.	int	ОПК-6
2	Какой оператор в C++ используется для вывода данных на экран?	cout	ОПК-6
3	Укажите оператор, используемый для ввода данных с клавиатуры в C++.	cin	ОПК-6
4	Какой заголовочный файл необходимо подключить для использования <code>cout</code> и <code>cin</code> ?	<iostream>	ОПК-6
5	Укажите оператор ветвления, используемый для выбора одного из двух путей выполнения.	if-else	ОПК-6
6	Какой цикл в C++ выполняется хотя бы один раз, независимо от условия?	do-while	ОПК-6
7	Укажите оператор, который прерывает выполнение цикла в C++.	break	ОПК-6
8	Какой тип данных используется для хранения вещественных чисел в C++?	double	ОПК-6
9	Укажите способ объявления массива из 10 целых чисел.	int arr[10];	ОПК-6
10	Какой индекс имеет первый элемент массива в C++?	0	ОПК-6
11	Укажите функцию, с которой начинается выполнение любой программы на C++.	main	ОПК-6
12	Какой тип данных используется для хранения одиночного символа?	char	ОПК-6
13	Укажите ключевое слово для определения константы в C++.	const	ПК-4
14	Какой оператор используется для динамического выделения памяти в C++?	new	ПК-4
15	Укажите оператор, освобождающий динамически выделенную память.	delete	ПК-4
16	Укажите ключевое слово для объявления указателя в C++.	* (звездочка)	ПК-4

17	Какой ключевой элемент класса скрывает данные от прямого доступа?	private	ПК-4
18	Укажите принцип ООП, при котором один класс наследует свойства другого.	наследование	ПК-4
19	Укажите механизм, позволяющий функции иметь одно имя, но разные параметры.	перегрузка функций	ПК-3
20	Какой заголовочный файл в C++ используется для работы с динамическими массивами (std::vector)?	<vector>	ПК-3