

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ
575 ОТДЕЛЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ ОФИСА
ОБРАЗОВАТЕЛЬНЫХ ПРОГРАММ (М)

ОДОБРЕНО
УМС ИИКС Протокол №8/1/2025 от 25.08.2025 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ

Направление подготовки
(специальность)

[1] 01.03.02 Прикладная математика и информатика
[2] 09.03.04 Программная инженерия

Семестр	Трудоемкость, кред.	Общий объем курса, час.	Лекции, час.	Практич. занятия, час.	Лаборат. работы, час.	В форме практической подготовки/ В	СРС, час.	КСР, час.	Форма(ы) контроля, экз./зач./КР/КП
1	4	144	32	0	48		28-37	0	Э
2	3-4	108- 144	30	0	30		12-48	0	Э
Итого	7-8	252- 288	62	0	78	16	40-85	0	

АННОТАЦИЯ

Целью освоения учебной дисциплины «Введение в программирование» является подготовка программистов, способных писать программный код на языках C++ и Python с использованием таких подходов как объектно-ориентированное программирование (ООП), метапрограммирование и многопоточное программирование.

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целью освоения учебной дисциплины «Введение в программирование» является подготовка программистов, способных писать программный код на языках C++ и Python с использованием таких подходов как объектно-ориентированное программирование (ООП), метапрограммирование и многопоточное программирование.

2. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Освоение дисциплины предшествует изучению курсов и практикумов, которые требуют знаний, умений и навыков в области написания программного кода на C++ и Python, ООП, многопоточного и метапрограммирования. При изучении данной дисциплины используются базовые понятия из области информатики: переменные, циклы, условные операторы, функции, булева алгебра. Данная дисциплина используется как предшествующая для прочих курсов, в которых подразумевается умение студентов читать и писать программный код.

3. ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Универсальные и(или) общепрофессиональные компетенции:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
ОПК-1 [2] – Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	З-ОПК-1 [2] – Знать основные объекты дискретной математики и методы их описания и исследований; проблемы алгоритмической разрешимости задач и эффективной вычислимости чисел. У-ОПК-1 [2] – Уметь решать основные задачи математической логики; однозначно задавать объекты дискретной математики, приводить их к стандартным формам, выполнять эквивалентные преобразования; определять сложности алгоритмов, применение прямых и косвенных доказательств теорем, определение принадлежности функций к соответствующим классам В-ОПК-1 [2] – Владеть методами математической логики для решения задач формализации, анализа и синтеза логических схем, для нахождения инвариантов циклических и условных конструкций в информатике, для выполнения эквивалентных преобразований; методами применения логического подхода к решению сложных задач с помощью их декомпозиции.

ОПК-1 [1] – Способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности	З-ОПК-1 [1] – знать естественнонаучные методы познания окружающего мира, знать фундаментальный математический аппарат; У-ОПК-1 [1] – уметь применять естественнонаучные и математические методы исследования различных явлений, процессов и задач В-ОПК-1 [1] – владеть навыками исследования различных явлений и процессов с использованием естественнонаучного и математического подхода
ОПК-2 [2] – Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	З-ОПК-2 [2] – Знает принципы работы современных информационных технологий У-ОПК-2 [2] – Умеет использовать программные средства, в том числе отечественного производства, для решения задач профессиональной деятельности В-ОПК-2 [2] – Владеет программными средствами, в том числе отечественного производства, для решения задач профессиональной деятельности
ОПК-2 [1] – Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	З-ОПК-2 [1] – знать существующие математические методы и системы программирования необходимые для реализации алгоритмов решения прикладных задач У-ОПК-2 [1] – уметь использовать и адаптировать существующие математические методы и системы программирования необходимые для реализации алгоритмов решения прикладных задач В-ОПК-2 [1] – владеть навыками реализации математических алгоритмов для решения прикладных задач с использованием существующих систем программирования
ОПК-3 [1] – Способен применять и модифицировать математические модели для решения задач в области профессиональной деятельности	З-ОПК-3 [1] – знать принципы построения математических моделей физических явлений и процессов У-ОПК-3 [1] – уметь формулировать математические модели различных явлений и процессов на основе физических принципов и законов В-ОПК-3 [1] – владеть навыками построения математических моделей физических явлений и процессов
ОПК-6 [2] – Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	З-ОПК-6 [2] – Знать основы информатики и программирования У-ОПК-6 [2] – Уметь разрабатывать алгоритмы и программы; проектировать, конструировать и тестировать программные продукты В-ОПК-6 [2] – Владеть основами информатики и программирования
ОПК-8 [2] – Способен	З-ОПК-8 [2] – Знать способы осуществлять поиск,

осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	хранение, обработку и анализ информации из различных источников и баз данных У-ОПК-8 [2] – Уметь осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных; представлять информацию в требуемом формате с использованием информационных, компьютерных и сетевых технологий; В-ОПК-8 [2] – Владеть методами поиска, хранения, обработки и анализ информации из различных источников и баз данных
УКЦ-1 [1, 2] – Способен в цифровой среде использовать различные цифровые средства, позволяющие во взаимодействии с другими людьми достигать поставленных целей	З-УКЦ-1 [1, 2] – Знать: современные информационные технологии и цифровые средства коммуникации, в том числе отечественного производства, а также основные приемы и нормы социального взаимодействия и технологии межличностной и групповой коммуникации с использованием дистанционных технологий У-УКЦ-1 [1, 2] – Уметь: выбирать современные информационные технологии и цифровые средства коммуникации, в том числе отечественного производства, а также устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе и применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды с использованием дистанционных технологий В-УКЦ-1 [1, 2] – Владеть: навыками применения современных информационных технологий и цифровых средств коммуникации, в том числе отечественного производства, а также методами и приемами социального взаимодействия и работы в команде с использованием дистанционных технологий
УКЦ-2 [1, 2] – Способен искать нужные источники информации и данные, воспринимать, анализировать, запоминать и передавать информацию с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач	З-УКЦ-2 [1, 2] – Знать: методики сбора и обработки информации с использованием цифровых средств, а также актуальные российские и зарубежные источники информации в сфере профессиональной деятельности, принципы, методы и средства решения стандартных задач профессиональной деятельности с использованием цифровых средств и с учетом основных требований информационной безопасности У-УКЦ-2 [1, 2] – Уметь: применять методики поиска, сбора и обработки информации; с использованием цифровых средств, осуществлять критический анализ и синтез информации, полученной из разных источников, и решать стандартные задачи профессиональной деятельности с использованием цифровых средств и с учетом основных требований информационной безопасности В-УКЦ-2 [1, 2] – Владеть: методами поиска, сбора и обработки, критического анализа и синтеза информации с использованием цифровых средств для решения

	поставленных задач, навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций и библиографии по научно-исследовательской работе с использованием цифровых средств и с учетом требований информационной безопасности
УКЦ-3 [1, 2] – Способен ставить себе образовательные цели под возникающие жизненные задачи, подбирать способы решения и средства развития (в том числе с использованием цифровых средств) других необходимых компетенций	З-УКЦ-3 [1, 2] – Знать: основные приемы эффективного управления собственным временем, основные методики самоконтроля, саморазвития и самообразования на протяжении всей жизни с использованием цифровых средств У-УКЦ-3 [1, 2] – Уметь: эффективно планировать и контролировать собственное время, использовать методы саморегуляции, саморазвития и самообучения в течение всей жизни с использованием цифровых средств В-УКЦ-3 [1, 2] – Владеть: методами управления собственным временем, технологиями приобретения, использования и обновления социокультурных и профессиональных знаний, умений, и навыков; методиками саморазвития и самообразования в течение всей жизни с использованием цифровых средств

4. ВОСПИТАТЕЛЬНЫЙ ПОТЕНЦИАЛ ДИСЦИПЛИНЫ

Направления/цели воспитания	Задачи воспитания (код)	Воспитательный потенциал дисциплин
Профессиональное воспитание	Создание условий, обеспечивающих, формирование культуры решения изобретательских задач (В37)	1. Использование воспитательного потенциала дисциплин "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры написания и оформления программ, а также привития навыков командной работы за счет использования систем управления проектами и контроля версий. 2.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования культуры решения изобретательских задач, развития логического мышления, путем погружения студентов в научную и инновационную деятельность института и вовлечения в проектную

		работу. 3.Использование воспитательного потенциала профильных дисциплин для формирования навыков цифровой гигиены, а также системности и гибкости мышления, посредством изучения методологических и технологических основ обеспечения информационной безопасности и кибербезопасности при выполнении и защите результатов учебных заданий и лабораторных работ по криптографическим методам защиты информации в компьютерных системах и сетях. 4.Использование воспитательного потенциала дисциплин "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры безопасного программирования посредством тематического акцентирования в содержании дисциплин и учебных заданий. 5.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования системного подхода по обеспечению информационной безопасности и кибербезопасности в различных сферах деятельности посредством исследования и перенятия опыта постановки и решения научно-практических задач организациями-партнерами.
Профессиональное воспитание	Создание условий, обеспечивающих, формирование навыков цифровой гигиены (В38)	1. Использование воспитательного потенциала дисциплин "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры написания и оформления программ, а также привития навыков командной работы за счет использования систем

		управления проектами и контроля версий. 2.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования культуры решения изобретательских задач, развития логического мышления, путем погружения студентов в научную и инновационную деятельность института и вовлечения в проектную работу. 3.Использование воспитательного потенциала профильных дисциплин для формирования навыков цифровой гигиены, а также системности и гибкости мышления, посредством изучения методологических и технологических основ обеспечения информационной безопасности и кибербезопасности при выполнении и защите результатов учебных заданий и лабораторных работ по криптографическим методам защиты информации в компьютерных системах и сетях. 4.Использование воспитательного потенциала дисциплин "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры безопасного программирования посредством тематического акцентирования в содержании дисциплин и учебных заданий. 5.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования системного подхода по обеспечению информационной безопасности и кибербезопасности в различных сферах деятельности посредством исследования и перенятия опыта постановки и решения научно-практических задач организациями-партнерами.
Профессиональное воспитание	Создание условий, обеспечивающих, формирование	1. Использование воспитательного потенциала дисциплин "Информатика (Основы

	ответственности за обеспечение кибербезопасности (B39)	программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры написания и оформления программ, а также привития навыков командной работы за счет использования систем управления проектами и контроля версий. 2.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования культуры решения изобретательских задач, развития логического мышления, путем погружения студентов в научную и инновационную деятельность института и вовлечения в проектную работу. 3.Использование воспитательного потенциала профильных дисциплин для формирования навыков цифровой гигиены, а также системности и гибкости мышления, посредством изучения методологических и технологических основ обеспечения информационной безопасности и кибербезопасности при выполнении и защите результатов учебных заданий и лабораторных работ по криптографическим методам защиты информации в компьютерных системах и сетях. 4.Использование воспитательного потенциала дисциплин "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры безопасного программирования посредством тематического акцентирования в содержании дисциплин и учебных заданий. 5.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования системного подхода по обеспечению
--	---	--

		информационной безопасности и кибербезопасности в различных сферах деятельности посредством исследования и перенятия опыта постановки и решения научно-практических задач организациями-партнерами.
Профессиональное воспитание	Создание условий, обеспечивающих, формирование профессионально значимых установок: не производить, не копировать и не использовать программные и технические средства, не приобретённые на законных основаниях; не нарушать признанные нормы авторского права; не нарушать тайны передачи сообщений, не практиковать вскрытие информационных систем и сетей передачи данных; соблюдать конфиденциальность доверенной информации (B40)	1. Использование воспитательного потенциала дисциплин "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и структуры данных)" для формирования культуры написания и оформления программ, а также привития навыков командной работы за счет использования систем управления проектами и контроля версий. 2.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования культуры решения изобретательских задач, развития логического мышления, путем погружения студентов в научную и инновационную деятельность института и вовлечения в проектную работу. 3.Использование воспитательного потенциала профильных дисциплин для формирования навыков цифровой гигиены, а также системности и гибкости мышления, посредством изучения методологических и технологических основ обеспечения информационной безопасности и кибербезопасности при выполнении и защите результатов учебных заданий и лабораторных работ по криптографическим методам защиты информации в компьютерных системах и сетях. 4.Использование воспитательного потенциала дисциплин " "Информатика (Основы программирования)", Программирование (Объектно-ориентированное программирование)", "Программирование (Алгоритмы и

		структуры данных)" для формирования культуры безопасного программирования посредством тематического акцентирования в содержании дисциплин и учебных заданий. 5.Использование воспитательного потенциала дисциплины "Проектная практика" для формирования системного подхода по обеспечению информационной безопасности и кибербезопасности в различных сферах деятельности посредством исследования и перенятия опыта постановки и решения научно-практических задач организациями-партнерами.
--	--	--

5. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Разделы учебной дисциплины, их объем, сроки изучения и формы контроля:

№ п.п	Наименование раздела учебной дисциплины	Недели	Лекции/ Практи. (семинары)/ Лабораторные работы, час.	Обязат. текущий контроль (форма*, неделя)	Максимальный балл за раздел**	Аттестация раздела (форма*, неделя)	Индикаторы освоения компетенции
	<i>1 Семестр</i>						
1	Первый раздел "Основы синтаксиса языка C++"	1-8	16/0/24		17	КИ-8	З-ОПК-1, У-ОПК-1, В-ОПК-1, З-ОПК-1, У-ОПК-1, В-ОПК-1, З-ОПК-2, У-ОПК-2, В-ОПК-2, З-ОПК-2, У-ОПК-2, В-ОПК-2, З-ОПК-3, У-ОПК-3, В-ОПК-3, З-ОПК-6, У-ОПК-6, В-ОПК-6,

							3-ОПК-8, У-ОПК-8, В-ОПК-8, 3-УКЦ-1, У-УКЦ-1, В-УКЦ-1, 3-УКЦ-2, У-УКЦ-2, В-УКЦ-2, 3-УКЦ-3, У-УКЦ-3, В-УКЦ-3
2	Второй раздел	9-16	16/0/24		33	КИ-16	В-УКЦ-2, 3-УКЦ-3, У-УКЦ-3, В-УКЦ-3, 3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-3, У-ОПК-3, В-ОПК-3, 3-ОПК-6, У-ОПК-6, В-ОПК-6, 3-ОПК-8, У-ОПК-8, В-ОПК-8, 3-УКЦ-1, У-УКЦ-1, В-УКЦ-1, 3-УКЦ-2, У-УКЦ-2
	<i>Итого за 1 Семестр</i>		32/0/48		50		
	Контрольные мероприятия за 1 Семестр				50	Э	3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-2, У-ОПК-2, В-ОПК-2,

							3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-3, У-ОПК-3, В-ОПК-3, 3-ОПК-6, У-ОПК-6, В-ОПК-6, 3-ОПК-8, У-ОПК-8, В-ОПК-8, 3-УКЦ-1, У-УКЦ-1, В-УКЦ-1, 3-УКЦ-2, У-УКЦ-2, В-УКЦ-2, 3-УКЦ-3, У-УКЦ-3, В-УКЦ-3
	2 Семестр						
1	Третий раздел	1-8	16/0/16		25	КИ-8	3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-3, У-ОПК-3, В-ОПК-3, 3-ОПК-6, У-ОПК-6, В-ОПК-6, 3-ОПК-8, У-ОПК-8, В-ОПК-8, 3-УКЦ-1, У-УКЦ-1, В-УКЦ-1, 3-УКЦ-2, У-УКЦ-2, В-УКЦ-2, 3-УКЦ-3, У-УКЦ-3, В-УКЦ-3

2	Четвертый раздел	9-15	14/0/14		25	КИ-15	3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-3, У-ОПК-3, В-ОПК-3, 3-ОПК-6, У-ОПК-6, В-ОПК-6, 3-ОПК-8, У-ОПК-8, В-ОПК-8, 3-УКЦ-1, У-УКЦ-1, В-УКЦ-1, 3-УКЦ-2, У-УКЦ-2, В-УКЦ-2, 3-УКЦ-3, У-УКЦ-3, В-УКЦ-3
	<i>Итого за 2 Семестр</i>		30/0/30		50		
	Контрольные мероприятия за 2 Семестр				50	Э	3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-1, У-ОПК-1, В-ОПК-1, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-2, У-ОПК-2, В-ОПК-2, 3-ОПК-3, У-ОПК-3, В-ОПК-3, 3-ОПК-6, У-ОПК-6, В-ОПК-6, 3-ОПК-8, У-ОПК-8, В-ОПК-8,

							3-УКЦ-1, У-УКЦ-1, В-УКЦ-1, 3-УКЦ-2, У-УКЦ-2, В-УКЦ-2, 3-УКЦ-3, У-УКЦ-3, В-УКЦ-3
--	--	--	--	--	--	--	---

* – сокращенное наименование формы контроля

** – сумма максимальных баллов должна быть равна 100 за семестр, включая зачет и (или) экзамен

Сокращение наименований форм текущего контроля и аттестации разделов:

Обозначение	Полное наименование
КИ	Контроль по итогам
Э	Экзамен

КАЛЕНДАРНЫЙ ПЛАН

Недели	Темы занятий / Содержание	Лек., час.	Пр./сем., час.	Лаб., час.
	<i>1 Семестр</i>	32	0	48
1-8	Первый раздел "Основы синтаксиса языка C++"	16	0	24
1 - 5	1.1 Синтаксис языка C++ 1.1 Синтаксис языка C++ Базовый синтаксис языка C++. Указатели, ссылки, арифметика указателей, const/mutable. Классы, структуры, инкапсуляция. Конструкторы, деструкторы. Базовые STL контейнеры: vector, stack, queue. Шаблонные классы, шаблонные функции. Итераторы.	Всего аудиторных часов		
		10	0	15
		Онлайн		
		0	0	0
6 - 8	1.2 Вспомогательные знания Системы контроля версий. Git.	Всего аудиторных часов		
		6	0	9
		Онлайн		
		0	0	0
9-16	Второй раздел	16	0	24
9 - 11	2.1 Библиотека STL Что такое Standard Template Library. Контейнеры в STL. Строки в C++, C-строки, std::string, std::string_view. Потоки ввода-вывода. STL Алгоритмы: sort, lower bound, upper bound, find, remove if, merge. Простые структуры: pair, optional. std::exception, обработка ошибок в C++	Всего аудиторных часов		
		6	0	9
		Онлайн		
		0	0	0
12 - 13	2.2 Современный C++ Move-семантика. std::function. Structured bindings. Lambda. Автоматическое выведение типов на этапе компиляции. Динамическое управление памятью. Умные указатели. RAII.	Всего аудиторных часов		
		4	0	6
		Онлайн		
		0	0	0
14 - 16	2.3 Метапрограммирование Выражения и функции, вычисляемые на этапе компиляции,	Всего аудиторных часов		
		6	0	9

	constexpr. Принципы подстановки значений в шаблоны, SFINAE. Выведение типов, decltype, declval. Работа с кортежами на этапе компиляции, std::tuple. Шаблоны с переменным количеством параметров (variadic templates). Условный оператор на этапе компиляции (constexpr if). Вспомогательные операции для манипуляции с типами во время компиляции, std::conditional, std::decay, std::enable_if, std::static_assert.	Онлайн		
		0	0	0
	2 Семестр	30	0	30
1-8	Третий раздел	16	0	16
1 - 4	3.1 Многопоточное программирование Базовая многопоточность: std::thread, std::mutex, std::shared_mutex, std::atomic. False sharing. Concurrent hash set. FixedQueue - очередь ограниченного размера с ожиданием в push/pop при условии заполненности/пустоты на основе std::mutex и std::condition_variable. Condition variable. Семафоры. Deadlock. Каналы. Singleton. Mutex. SpinLock. Atomic: Compare and Swap (CAS).	Всего аудиторных часов		
		8	0	8
		Онлайн		
		0	0	0
5 - 6	3.2 Системные вызовы и сеть Системные вызовы. Работа с диском и файловой системой. Инструменты дебага — gdb, coredumps, санитайзеры. Сетевая модель OSI. Сетевые протоколы. Сетевые системные вызовы. Инструменты дебага сети в linux.	Всего аудиторных часов		
		4	0	4
		Онлайн		
		0	0	0
7 - 8	3.3 Вспомогательные знания Компиляция c++ проектов: препроцессинг, компиляция, линковка. AST - abstract syntax tree. Объявления (declaration) и определения (definition). pragma once. ODR - one definition rule. forward declaration.	Всего аудиторных часов		
		4	0	4
		Онлайн		
		0	0	0
9-15	Четвертый раздел	14	0	14
9 - 15	4.1 Краткий курс по Python Простые типы. Управляющие конструкции. Структуры данных. Функции, ввод-вывод, строки, файлы. Неймспейсы, замыкания, декораторы. Итераторы. Генераторы. Классы, datetime. Исключения, контекстные менеджеры. Модули, пакеты, импорт, setup.py. Bindings. Subprocess, threading, multiprocessing. WEB, HTTP, API	Всего аудиторных часов		
		14	0	14
		Онлайн		
		0	0	0

Сокращенные наименования онлайн опций:

Обозначение	Полное наименование
ЭК	Электронный курс
ПМ	Полнотекстовый материал
ПЛ	Полнотекстовые лекции
ВМ	Видео-материалы
АМ	Аудио-материалы
Прз	Презентации
Т	Тесты
ЭСМ	Электронные справочные материалы
ИС	Интерактивный сайт

ТЕМЫ ЛАБОРАТОРНЫХ РАБОТ

Недели	Темы занятий / Содержание
	<i>1 Семестр</i>
1 - 2	Работа №1 Базовый синтаксис языка C++
3 - 4	Работа №2 Простые контейнеры
5 - 6	Работа №3 Шаблоны
7 - 8	Работа №4 Итераторы
9	Работа №5 Продвинутые контейнеры
10	Работа №6 Строки
11	Работа №7 Алгоритмы с итераторами
12	Работа №8 Работа с памятью 1
13	Работа №9 Работа с памятью 2
14	Работа №10 Объектно-ориентированное программирование
15	Работа №11 Обработка ошибок
	<i>2 Семестр</i>
1	Работа №12 Примитивы синхронизации: mutex, atomic, lock
2	Работа №13 Примитивы синхронизации: conditional variables atomic: shared_mutex
3	Работа №14 Пул потоков
4	Работа №15 Парсинг torrent-файла
5	Работа №16 Запрос списка пиров через протокол torrent-трекера
6	Работа №17 Работа с сетью, tcp-сокеты
7	Работа №18 Установление peer-to-peer соединения по протоколу torrent
8	Работа №19 Загрузка частей файлов по протоколу torrent
9 - 11	Работа №20 Базовый синтаксис python
12 - 13	Работа №21 Скачивание файлов по протоколу torrent
14 - 15	Работа №22 Создание command line interface для torrent-клиента

6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Дидактической основой выполнения лабораторного практикума по дисциплине является метод учебного проектирования, предусматривающий:

- получение студентами теоретических знаний в ходе лекций и при самостоятельной работе с материалами учебных пособий, размещенных в библиотеке университета;
- формирование навыков применения сведений теоретического и аналитического характера при решении конкретных задач в ходе проведения контрольно-измерительных мероприятий по разделам курса.

При проведении лабораторных занятий и самостоятельной работы студентов используется компьютерная обучающая система manytask. Компьютерная обучающая система выполняет следующие функции: предъявление лабораторных и домашних заданий и контроль сроков их выполнения, сбор и хранение результатов выполнения лабораторных и домашних заданий и их частичную проверку.

7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

Компетенция	Индикаторы освоения	Аттестационное мероприятие (КП 1)	Аттестационное мероприятие (КП 2)
ОПК-1	З-ОПК-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
ОПК-2	З-ОПК-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
ОПК-6	З-ОПК-6	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-6	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-6	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
ОПК-8	З-ОПК-8	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-8	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-8	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
УКЦ-1	З-УКЦ-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-УКЦ-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-УКЦ-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
УКЦ-2	З-УКЦ-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-УКЦ-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-УКЦ-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
УКЦ-3	З-УКЦ-3	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-УКЦ-3	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-УКЦ-3	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
ОПК-1	З-ОПК-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-1	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15

ОПК-2	З-ОПК-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-2	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
ОПК-3	З-ОПК-3	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	У-ОПК-3	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15
	В-ОПК-3	Э, КИ-8, КИ-16	Э, КИ-8, КИ-15

Шкалы оценки образовательных достижений

Шкала каждого контрольного мероприятия лежит в пределах от 0 до установленного максимального балла включительно. Итоговая аттестация по дисциплине оценивается по 100-балльной шкале и представляет собой сумму баллов, заработанных студентом при выполнении заданий в рамках текущего и промежуточного контроля.

Итоговая оценка выставляется в соответствии со следующей шкалой:

Сумма баллов	Оценка по 4-ех балльной шкале	Оценка ECTS	Требования к уровню освоению учебной дисциплины
90-100	5 – <i>«отлично»</i>	A	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, использует в ответе материал монографической литературы.
85-89	4 – <i>«хорошо»</i>	B	Оценка «хорошо» выставляется студенту, если он твёрдо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос.
75-84		C	
70-74		D	
65-69	3 – <i>«удовлетворительно»</i>	E	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала.
60-64			
Ниже 60	2 – <i>«неудовлетворительно»</i>	F	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОСНОВНАЯ ЛИТЕРАТУРА:

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА:

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ:

Специальное программное обеспечение не требуется

LMS И ИНТЕРНЕТ-РЕСУРСЫ:

<https://online.mephi.ru/>

<http://library.mephi.ru/>

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Специальное материально-техническое обеспечение не требуется

10. УЧЕБНО-МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ СТУДЕНТОВ

Программа изучения курса «Введение в программирование» включает в себя помимо лекционных занятий выполнение 11-ти домашних работ в первом семестре и 11 домашних работ во втором семестре, а также 4 контрольных работ в первом семестре и 4 контрольных работ во втором семестре. Аттестация по теме ставится в случае, если решение доведено до конца, а количество допущенных в процессе него ошибок не превышает заранее оговоренного.

На восьмой неделе полусеместровый контроль проставляется на основании результатов контрольно-измерительных мероприятий, проведенных к данному времени.

В конце первого семестра студент сдает зачёт с оценкой по курсу, в конце второго семестра – экзамен.

Текущий контроль успеваемости в первом семестре осуществляется в соответствии со следующей таблицей:

№ разд,

№ мер. Наименование контрольного мероприятия Срок проведения, нед. Срок проверки, нед. Максимальный балл

1	1.1 Домашняя работа №1		
	Базовый синтаксис языка C++	3	3
1	1.1 Домашняя работа №2		
	Простые контейнеры	4	4
1	1.1 Домашняя работа №3		
	Шаблоны	5	5
1	1.1 Домашняя работа №4		

3	Итераторы	6	6
7	1.1 Контрольная работа №1 Итераторы	6	6
1	2.1 Домашняя работа №5 Продвинутые контейнеры	7	7
3	2.1 Домашняя работа №6 Строки	8	8
7	1.1, 2.2 Контрольная работа №2 TNumber	9	9
1	2.1 Домашняя работа №7 Алгоритмы с итераторами	12	12
1	2.2 Домашняя работа №8 Работа с памятью 1	13	13
2	2.2 Домашняя работа №9 Работа с памятью 2	13	13
8	2.1, 2.2 Контрольная работа №3 Умные указатели	13	13
8	2.2 Контрольная работа №4 Объектно-ориентированное программирование	14	14
3	1.1, 2.1, 2.2 Домашняя работа №10 Объектно-ориентированное программирование	15	15
3	1.1, 2.1, 2.2 Домашняя работа №11 Обработка ошибок	15	15

Текущая аттестация проводится в соответствии с таблицей:

№	Наименование мероприятия	Срок проведения, нед.	Максимальный балл
Примечание			
1	Контроль по итогам освоения раздела 1	8	17
2	Контроль по итогам освоения раздела 2	16	33
3	Зачет с оценкой	Зачетная сессия	50 при
условии успешного прохождения всех текущих форм контроля			
	Итого 100		

В конце семестра проводится зачёт с оценкой, к которому допускаются студенты, успешно выполнившие задания по всем разделам семестра. Зачёт включает в себя теоретический вопрос, практическую задачу и дополнительные задания.

Задания для проведения текущего контроля и промежуточной аттестации

- Реализовать буферизованный строковый поток над `std::cin`.
- Реализовать `deque` поверх кольцевого буфера
- Реализовать `FlatMap`: реализацию аналога `std::map` поверх `std::vector`
- Реализовать шаблонный класс `InserterIterator`, который обладает интерфейсом `output iterator`'а и позволяет последовательно вставлять значения в данный контейнер, начиная с данной позиции
- Реализовать `intrusive list`
- Реализовать шаблонный класс `MapperIterator`, который представляет собой итератор, ссылающийся на другой итератор, и позволяющий получить результат применения какой-то заданной функции к значению, на которое ссылается внутренний итератор.
- Реализовать шаблонный класс `WindowAverageIterator`, который позволяет пробегать по диапазону значений и вычислять среднее значение в окне заданного размера

Текущий контроль успеваемости во втором семестре осуществляется в соответствии со следующей таблицей:

№ разд,	№ мер. Наименование контрольного мероприятия	Срок проведения, нед.
Срок проверки, нед.	Максимальный балл	
	3.1 Домашняя работа №1	
	Примитивы синхронизации: <code>mutex</code> , <code>atomic</code> , <code>lock</code>	2
2	2	
	3.1 Домашняя работа №2	
	Примитивы синхронизации: <code>conditional variables</code>	4
4	1	
	3.1 Домашняя работа №3	
	<code>atomic</code> : <code>shared_mutex</code>	5
5	1	
	3.1 Контрольная работа №1	
	Пул потоков	4
4	8	
	3.1 Домашняя работа №4	
	Пул потоков	6
6	1	
	3.2 Домашняя работа №5	
	Системные вызовы	7
7	2	
	3.2 Домашняя работа №6	
	Сеть	8
8	2	
	3.3 Контрольная работа №2	

	Сеть и системные вызовы	8
8	8	
	4.1 Домашняя работа №6	
	Основы python. Функции, структуры данных. Ввод-вывод и файлы	9
9	1	
	4.1 Домашняя работа №7	
	Классы, итераторы, генераторы	11
11	1	
	4.1 Домашняя работа №6	
	Исключения и контекстные менеджеры	12
12	1	
	4.1 Контрольная работа №3	
	Основы python.	12
12	7	
	4.1 Домашняя работа №7	
	Параллельная работа в python. Сеть	14
14	1	
	4.1 Контрольная работа №6	
	Параллельная работа в python. Сеть	15
15	7	
	3.1, 3.2, 3.3, 4.1 Домашняя работа №8	
	Взаимодействие C++ и python3 на основе torrent клиент-сервера.	15
15	7	

Текущая аттестация проводится в соответствии с таблицей:

№	Наименование мероприятия	Срок проведения, нед.
Максимальный балл	Примечание	
1	Контроль по итогам освоения раздела 1	8
25		
2	Контроль по итогам освоения раздела 2	15
25		
3	Экзамен	Экзаменационная сессия
		50 при
	условии успешного прохождения всех текущих форм контроля	
	Итого 100	

В конце семестра проводится экзамен, к которому допускаются студенты, успешно выполнившие задания по всем разделам семестра. Экзамен включает в себя теоретический вопрос, практическую задачу и дополнительные задания.

Задания для проведения текущего контроля и промежуточной аттестации

- Реализовать пул потоков
- Реализовать фрагменты сетевых протоколов TCP / UDP / HTTP
- Реализовать иерархию наследования нескольких классов
- Реализовать простое клиент серверное взаимодействие между разными подпроцессами внутри одного приложения

11. УЧЕБНО-МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРЕПОДАВАТЕЛЕЙ

Программа изучения курса «Введение в программирование» включает в себя помимо лекционных занятий выполнение 11-ти домашних работ в первом семестре и 11 домашних работ во втором семестре, а также 4 контрольных работ в первом семестре и 4 контрольных работ во втором семестре. Аттестация по теме ставится в случае, если решение доведено до конца, а количество допущенных в процессе него ошибок не превышает заранее оговоренного.

На восьмой неделе полусеместровый контроль проставляется на основании результатов контрольно-измерительных мероприятий, проведенных к данному времени.

В конце первого семестра студент сдает зачёт с оценкой по курсу, в конце второго семестра – экзамен.

Задания для проведения текущего контроля и промежуточной аттестации в первом семестре

- Реализовать буферизованный строковый поток над `std::cin`.
- Реализовать `deque` поверх кольцевого буфера
- Реализовать `FlatMap`: реализацию аналога `std::map` поверх `std::vector`
- Реализовать шаблонный класс `InsertIterator`, который обладает интерфейсом `output iterator`'а и позволяет последовательно вставлять значения в данный контейнер, начиная с данной позиции
- Реализовать `intrusive list`
- Реализовать шаблонный класс `MapperIterator`, который представляет собой итератор, ссылающийся на другой итератор, и позволяющий получить результат применения какой-то заданной функции к значению, на которое ссылается внутренний итератор.
- Реализовать шаблонный класс `WindowAverageIterator`, который позволяет пробегать по диапазону значений и вычислять среднее значение в окне заданного размера

Задания для проведения текущего контроля и промежуточной аттестации во втором семестре

- Реализовать пул потоков
- Реализовать фрагменты сетевых протоколов TCP / UDP / HTTP
- Реализовать иерархию наследования нескольких классов
- Реализовать простое клиент серверное взаимодействие между разными подпроцессами внутри одного приложения

Автор(ы):

Таракчян Левон Суменович

Колобашкина Любовь Викторовна