

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ИНКЛЮЗИВНОГО ВЫСШЕГО
ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО -
ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»**

Факультет Прикладной математики и информатики
Кафедра Прикладной математики и информатики по областям

УТВЕРЖДАЮ

И.о. проректора по учебно-
методической работе
Хакимов Р.М.



«30»августа 2021г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ**

образовательная программа направления подготовки
09.04.03 Прикладная информатика
Блок Б1.О.07 «Дисциплины (модули)», обязательная часть

Профиль подготовки
Интеллектуальные биоинформационные технологии

Квалификация (степень) выпускника
Магистр

Форма обучения очная

Курс 1 семестр 1

Москва
2021

ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЙ РАЗДЕЛ

1.1. Цель и задачи изучения учебной дисциплины (модуля)

Цели и задачи изучения дисциплины «Современные технологии разработки программного обеспечения»: предоставление магистрантам знаний и умений в области проектирования, тестирования, отладки, внедрения и сопровождения программного обеспечения (ПО) и вычислительной техники с использованием современных CALS-технологий и CASE-средств.

1.2. Требования к результатам освоения дисциплины

Изучение данной дисциплины направлено на формирование следующих компетенций:

Код и содержание компетенции	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций
ОПК-2. Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	ОПК-2.1 Знает современные интеллектуальные технологии для решения профессиональных задач.
	ОПК-2.2 Умеет обосновывать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач.
ОПК-5.Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем	ОПК-5.1 Знает современное программное и аппаратное обеспечение информационных и автоматизированных систем.
	ОПК-5.2 Умеет модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач.
ОПК-8. Способен осуществлять эффективное управление разработкой программных средств и проектов	ОПК-8.1 Знает архитектуру информационных систем предприятий и организаций; методологии и технологии реинжиниринга, проектирования и аудита прикладных информационных систем различных классов; инструментальные средства поддержки технологии проектирования и аудита информационных систем и сервисов; методы оценки экономической эффективности и качества, управления надежностью и информационной безопасностью; особенности процессного подхода к управлению прикладными ИС; современные ИКТ в процессном управлении; системы управления качеством; концептуальное моделирование процессов управления знаниями; архитектуру систем управления знаниями; онтологии знаний; подсистемы сбора, фильтрации, накопления, доступа, генерации и распространения знаний.

	ОПК-8.2 Умеет выбирать методологию и технологию проектирования информационных систем; обосновывать архитектуру ИС; управлять проектами ИС на всех стадиях жизненного цикла, оценивать эффективность и качество проекта; применять современные методы управления проектами и сервисами ИС; использовать инновационные подходы к проектированию ИС; принимать решения по информатизации предприятий в условиях неопределенности; проводить реинжиниринг прикладных и информационных процессов; обосновывать архитектуру системы управления знаниями.
--	--

1.3. Место дисциплины (модуля) в структуре образовательной программы направления подготовки 09.04.03 «Прикладная информатика (уровень магистратуры)»

Учебная дисциплина «Современные технологии разработки программного обеспечения» относится к обязательной части блока Б.1. Изучение учебной дисциплины «Современные технологии разработки программного обеспечения» базируется на знаниях, умениях и навыках, полученных обучающимися при изучении дисциплин уровня бакалавриата.

Изучение учебной дисциплины «Современные технологии разработки программного обеспечения» необходимо для изучения дисциплин «Высокоуровневое программирование» и «Интеллектуальные информационные технологии (продвинутый уровень)».

2. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2.1. Объем дисциплины и виды учебной работы в соответствии с формами обучения

Объем дисциплины «Современные технологии разработки программного обеспечения» составляет 6 з.е./216 часов:

Вид учебной работы	Всего, часов	Очная форма
		Курс, часов
		1 курс, 1 сем.
Аудиторная работа обучающихся с преподавателем (по видам учебных занятий), всего в том числе:	48	48
Лекции	12	12
Практические занятия	36	36
Лабораторные занятия		
Самостоятельная работа обучающихся	132	132
Промежуточная аттестация (подготовка и сдача), всего:		
Контрольная работа		
Курсовая работа		
Зачет		
Экзамен	36	36
Итого: Общая трудоемкость учебной дисциплины (в часах, зачетных единицах)	216/6	216/6

Содержание дисциплины по темам (разделам)

№ п/п	Наименование раздела (темы)	Содержание раздела (тематика занятий)	Формируемые компетенции (индекс)
1.	Раздел 1. Введение в технологии объектно-ориентированного программирования	Введение в технологии объектно-ориентированного программирования. Особенности технологии Java и C++. Структура приложений. Основные типы данных и операции. Приведение типов. Массивы. Реализация в C++ и Java	ОПК-2, ОПК-5, ОПК-8
2.	Раздел 2. Понятие класса, абстрактные классы и интерфейсы в Java и C++	Понятие класса, абстрактные классы и интерфейсы в Java и C++. Манипулирование памятью с помощью указателей в C++. Создание функций в C++. Структура классов в Java. Разработка классов в C++ и Java. Область видимости членов класса. Основные классы пакета java.lang	ОПК-2, ОПК-5, ОПК-8
3.	Раздел 3. Наследование в C++ и Java. Сложные структуры данных.	Наследование в C++ и Java. Реализация сложных структур данных в C++ и Java.	ОПК-2, ОПК-5, ОПК-8
4.	Раздел 4. Реализация функциональности сложных структур данных в C++ и Java	Реализация функциональности сложных структур данных в C++ (STL и основные методы) и Java (интерфейсы, классы и алгоритмы коллекций). Шаблоны. Программирование шаблонных функций в C++. Понятие исключительной ситуации и технологии обработка исключений	ОПК-2, ОПК-5, ОПК-8
5.	Раздел 5. Основные понятия XML. Схемы документов. Организация связи документа XML со структурами данных в C++ и Java	Основные понятия XML. Схемы документов. Организация связи документа XML со структурами данных в C++ и Java. Организация сетевого взаимодействия и взаимодействия с БД в Java.	ОПК-2, ОПК-5, ОПК-8
6.	Раздел 6. Проектирование информационных систем. Использование UML	Проектирование информационных систем. Использование UML Виды UML схем. Разработка UML диаграмм классов и БД учебного проекта.	ОПК-2, ОПК-5, ОПК-8

2.3. Разделы дисциплин и виды занятий

№ п/п	Наименование темы дисциплины	Лекционные занятия	Практические занятия	Самостоятельная работа	Всего часов	Формы текущего контроля успеваемости
1.	Введение в технологии объектно-ориентированного программирования	2	6	22	30	Устный опрос
2.	Понятие класса, абстрактные классы и интерфейсы в Java и C++	2	6	22	30	Устный опрос
3.	Наследование в C++ и Java. Сложные структуры данных	2	6	22	30	Устный опрос
4.	Реализация функциональности сложных структур данных в C++ и Java	2	6	22	30	Устный опрос
5.	Основные понятия XML. Схемы документов. Организация связи документа XML со структурами данных в C++ и Java	2	6	22	30	Устный опрос
6.	Проектирование информационных систем. Использование UML	2	6	22	30	Устный опрос
Экзамен		36				
Итого:		12	36	132	180	

2.4. Планы теоретических (лекционных) занятий

№	Наименование тем лекций	Кол-во часов в 1 семестре
1 семестр		
РАЗДЕЛ 1. Введение в технологии объектно-ориентированного программирования.		
1.	Среда разработки, поддержка ООП в среде разработки. Добавление новых классов с использованием визардов. Включение стандартных классов и собственных классов в выполняемые модули в Visual Studio.	2
РАЗДЕЛ 2. Понятие класса, абстрактные классы и интерфейсы в Java и C++.		
1.	Объявление классов и объектов в C++. Время жизни объектов. Конструкторы и деструкторы. Встроенные типы данных. Массивы. Массивы объектов. Объявление классов и создание объектов в Java. Ссылки и объекты в Java. Сборщик мусора. Время жизни ссылок и время жизни объектов в Java. Типы примитивы в Java. Массивы.	2
РАЗДЕЛ 3. Наследование в C++ и Java. Сложные структуры данных.		
1.	Особенности наследования в C++. Наследование и включение. Порядок создания сложных объектов.	2
РАЗДЕЛ 4. Реализация функциональности сложных структур данных в C++ и Java.		
1.	Шаблоны в C++. STL в C++. Обобщенные контейнеры и их свойства. Применение map для подсчета частоты различных слов в тексте.	2
РАЗДЕЛ 5. Основные понятия XML. Схемы документов. Организация связи документа XML со структурами данных в C++ и Java.		

1.	Основы XML. Применения XML. Методы обработки XML (SAX и DOM). Средства работы с XML в C++ на основе MSXML. Создание документа. Проверка соответствия документа схеме. Применение XPath для обработки документа.	2
РАЗДЕЛ 6. Проектирование информационных систем. Использование UML.		
1.	Обработка ошибок и проектирование информационных систем. Обработка исключительных ситуаций. Исключения, защищенный код. Блоки catch и finally.	2

2.5. Планы практических (семинарских) занятий

№	Наименование практических занятий	Кол-во часов в 1 семестре
I семестр		
РАЗДЕЛ 1. Введение в технологии объектно-ориентированного программирования.		
1.	Среда разработки, поддержка ООП в среде разработки. Структура исполняемой программы в Java. Включение стандартных классов и собственных классов в выполняемые модули в Java. Особенности исполнения Java программ и Java машина.	6
РАЗДЕЛ 2. Понятие класса, абстрактные классы и интерфейсы в Java и C++.		
1.	Виртуальные методы в C++. Абстрактные классы в C++ и их конкретизация. Анализ примера с рисованием фигур и обобщенный код.	2
2.	Понятие интерфейса в Java. Реализация интерфейсов в классе. Абстрактные классы в Java и их конкретизация. Анализ примера с рисованием фигур и обобщенный код.	2
3.	Манипулирование памятью с помощью указателей в C++. Понятие фрагментации памяти. Содержание операции по перераспределению памяти.	2
РАЗДЕЛ 3. Наследование в C++ и Java. Сложные структуры данных.		
1.	Наследование в Java. Наследование и включение. Классы обертки, делегировании функциональности. Эффективность наследования на примере классов потоков ввода-вывода. Пакет java.io.	6
РАЗДЕЛ 4. Реализация функциональности сложных структур данных в C++ и Java.		
1.	Generic коллекции и алгоритмы в Java. Реализация операция над математическим множествами на основе коллекций Java. Основные классы пакета java.lang. Пакет java.util.	6
РАЗДЕЛ 5. Основные понятия XML. Схемы документов. Организация связи документа XML со структурами данных в C++ и Java.		
1.	Обработка XML документа на основе SAX, на примере сериализации коллекции объектов.	2
2.	Организация сетевого взаимодействия и взаимодействия с БД в Java. JDBC. Сокеты в Java.	2
3.	Пользовательский интерфейс в Java. Пакет javax.swing. Понятие Applet. Пакет java.applet. Архитектура апплета. Класс Applet.	2
РАЗДЕЛ 6. Проектирование информационных систем. Использование UML.		
1.	Основные типы диаграмм UML. Прецеденты использования системы. Диаграммы классов. Диаграммы состояний и диаграммы последовательностей.	2
2.	Применение каждого вида UML диаграмм. Цель проектирования информационной системы. Последовательная детализация проекта.	4

2.6. Планы лабораторных работ – не предусмотрено.

2.7. Планы самостоятельной работы обучающегося по дисциплине (модулю).

№	Название разделов и тем	Виды самостоятельной работы	Трудоемкость	Формируемые компетенции	Формы контроля
1.	Введение в технологии объектно-ориентированного программирования	CASE-средства проектирования информационных систем	22	ОПК-2, ОПК-5, ОПК-8	Устный опрос
2.	Понятие класса, абстрактные классы и интерфейсы в Java и C++	Разработка отдельных классов	22	ОПК-2, ОПК-5, ОПК-8	Устный опрос
3.	Наследование в C++ и Java. Сложные структуры данных	Организация обработки исключительных ситуаций в приложениях	22	ОПК-2, ОПК-5, ОПК-8	Устный опрос
4.	Реализация функциональности сложных структур данных в C++ и Java	Разработка приложений со сложными структурами данных с использованием методов STL и Java классов коллекций	22	ОПК-2, ОПК-5, ОПК-8	Устный опрос
5.	Основные понятия XML. Схемы документов. Организация связи документа XML со структурами данных в C++ и Java	Описание базы данных с использованием XML и реализации взаимодействия приложений с базой данных	22	ОПК-2, ОПК-5, ОПК-8	Устный опрос
6.	Проектирование информационных систем. Использование UML	Описание вариантов использования проектируемой системы с помощью UML диаграмм	22	ОПК-2, ОПК-5, ОПК-8	Устный опрос

3. ОСОБЕННОСТИ ОБУЧЕНИЯ ЛИЦ С ИНВАЛИДНОСТЬЮ И ОВЗ

Обучение лиц с ограниченными возможностями здоровья осуществляется с учетом индивидуальных психофизических особенностей, а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида.

Для получения обучающимися, имеющими ограниченные физические возможности, качественного образования должны выполняться следующие важные условия: обучающийся должен иметь возможность беспрепятственно посещать образовательное учреждение и использовать в своём обучении дистанционные образовательные технологии.

Для обучения и контроля обучающихся с нарушениями координации движений предусмотрено проведение тестирования с использованием компьютера.

Во время аудиторных занятий обязательно использование средств обеспечения наглядности учебного материала с помощью мультимедийного проектора. Скорость изложения материала должна учитывать ограниченные физические возможности студентов.

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение дисциплины для организации самостоятельной работы студентов (содержит перечень основной литературы, дополнительной литературы, программного обеспечения и Интернет-ресурсы).

В распоряжении преподавателей и обучающихся имеется основное необходимое материально-техническое оборудование, Интернет-ресурсы, доступ к полнотекстовым электронным базам, книжный фонд библиотеки Московского государственного гуманитарно-экономического университета.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Перечень основной литературы

1. Казарин, О. В. Надежность и безопасность программного обеспечения: учебное пособие для бакалавриата и магистратуры / О. В. Казарин, И. Б. Шубинский. — Москва: Издательство Юрайт, 2019. — 342 с. — (Бакалавр и магистр. Модуль). — ISBN 978-5-534-05142-1. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/441287>
2. Казарин, О. В. Программно-аппаратные средства защиты информации. Защита программного обеспечения: учебник и практикум для вузов / О. В. Казарин, А. С. Забаурин. — Москва: Издательство Юрайт, 2019. — 312 с. — (Специалист). — ISBN 978-5-9916-9043-0. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/437163>
3. Астапчук, В. А. Корпоративные информационные системы: требования при проектировании: учебное пособие для вузов / В. А. Астапчук, П. В. Терещенко. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 113 с. — (Университеты России). — ISBN 978-5-534-08546-4. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/425572>

5.2 Перечень дополнительной литературы

1. Разработка высоконадежных интегрированных информационных систем управления предприятием/Капулин Д.В., Царев Р.Ю., Дрозд О.В. и др. - Краснояр: СФУ, 2015. - 184 с.: ISBN 978-5-7638-3227-3 - Текст: электронный. - URL: <https://znanium.com/catalog/product/549904>
2. Методологические основы построения защищенных автоматизированных систем: Монография / Душкин А.В. - Воронеж: Научная книга, 2016. - 76 с. ISBN 978-5-4446-0902-6 - Текст: электронный. - URL: <https://znanium.com/catalog/product/923295>

5.3 Программное обеспечение

1. Сетевой компьютерный класс, оснащенный современной техникой
2. Офисный программный пакет (например, Microsoft Office 2003 или более поздних версий).
3. Web-браузер Mozilla Firefox или Google Chrome
4. Экран для проектора

5.4 Электронные ресурсы

1. Электронная библиотека «Знаниум»: <https://znanium.com/>
2. Электронная библиотека «Юрайт»: <https://urait.ru/>
3. Научная электронная библиотека «Elibrary.ru»: <https://www.elibrary.ru/defaultx.asp>

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий	Перечень оборудования и технических средств обучения
1.	Лекционная аудитория	Персональный компьютер, мультимедийный проектор
2.	Компьютерный класс	Персональные компьютеры (IBM PC-совместимые) под управлением ОС Microsoft Windows, компьютерная сеть, доступ в сеть Интернет

7. ОЦЕНКА КОМПЕТЕНЦИЙ ПО ИЗУЧАЕМОЙ ДИСЦИПЛИНЕ

№	Критерии оценки			
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»
ЗНАТЬ				
1	Студент не способен самостоятельно выделять современные тенденции развития информатики и вычислительной техники, компьютерных технологий. Не знает основы создания информационных систем и использование новых информационных технологий обработки информации; жизненный цикл программного обеспечения; объектно-ориентированное программирование; теории и методы классификации ПО; элементы теории сложности.	Студент усвоил основное содержание материала дисциплины, но имеет пробелы в усвоении материала. Имеет несистематизированные знания о современных тенденциях развития информатики и вычислительной техники, компьютерных технологий; основах создания информационных систем и использование новых информационных технологий обработки информации; объектно-ориентированном программировании.	Студент способен самостоятельно выделять главные положения в изученном материале. Знает современные тенденции развития информатики и вычислительной техники, компьютерных технологий; основы создания информационных систем и использование новых информационных технологий обработки информации; жизненный цикл программного обеспечения; объектно-ориентированное программирование.	Студент знает, понимает, выделяет главные положения в изученном материале и способен дать краткую характеристику основным идеям проработанного материала дисциплины. Показывает глубокое знание и понимание современных тенденций развития информатики и вычислительной техники, компьютерных технологий; основ создания информационных систем и использования новых информационных технологий обработки информации; жизненного цикла программного обеспечения; объектно-ориентированного программирования; теории и методов классификации ПО; элементов теории сложности.
УМЕТЬ				
2	Студент не умеет применять математические методы, физические законы и вычислительную технику для	Студент испытывает затруднения при применении математических методов, физических законов и	Студент умеет применять математические методы, физические законы и вычислительную технику для	Студент умеет применять математические методы, физические законы и вычислительную технику для

	решения практических задач; не умеет программировать ни на одном из алгоритмических языков; не умеет применять алгоритмы поиска информации при разработке ПО.	вычислительной техники для решения практических задач; при применении алгоритмов поиска информации при разработке ПО.	решения практических задач; умеет применять алгоритмы поиска информации при разработке ПО	решения практических задач; умеет программировать на одном из алгоритмических языков; умеет применять алгоритмы поиска информации при разработке ПО.
ВЛАДЕТЬ				
3	Студент не владеет методами проведения объектно-ориентированной декомпозиции задачи в соответствии с заданными требованиями; не владеет способами реализации заданной спецификации (архитектуры) программной системы на языках программирования; не владеет навыками оценки качества спецификации (архитектуры) программной системы и ее кода.	Студент владеет основными методами проведения объектно-ориентированной декомпозиции задачи в соответствии с заданными требованиями.	Студент владеет основными методами проведения объектно-ориентированной декомпозиции задачи в соответствии с заданными требованиями; владеет способами реализации заданной спецификации (архитектуры) программной системы на языках программирования.	Студент владеет знаниями всего изученного материала; владеет методами проведения объектно-ориентированной декомпозиции задачи в соответствии с заданными требованиями; владеет способами реализации заданной спецификации (архитектуры) программной системы на языках программирования; владеет навыками оценки качества спецификации (архитектуры) программной системы и ее кода.
	Компетенции или их части не сформированы.	Компетенции или их части сформированы на базовом уровне.	Компетенции или их части сформированы на среднем уровне.	Компетенции или их части сформированы на высоком уровне.

8. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Интерактивные образовательные технологии, используемые в аудиторных занятиях и самостоятельной работе обучающихся – не предусмотрены.

9. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

9.1. Организация входного, текущего и промежуточного контроля обучения

Входное тестирование – не предусмотрено.

Текущий контроль – устный опрос.

Промежуточная аттестация – экзамен.

9.2. Тематика рефератов, проектов, творческих заданий, эссе и т.п.

Не предусмотрены.

9.3. Курсовая работа

Не предусмотрено.

9.4. Вопросы к зачету

9.5. Вопросы к экзамену

1. Основы анализа и проектирования информационных систем.
2. Назначение и виды моделей информационных систем.
3. CASE средства проектирования информационных систем.
4. Использование UML при проектировании информационной системы. Виды и назначение диаграмм.
5. Основные понятия и принципы ООП. Понятие класса и объекта. Состояние и поведение. Свойства и методы.
6. Интерфейс и реализация. Создание, уничтожение и время жизни объектов.
7. Наследование. Роль наследования в ООП. Синтаксическое описание наследования. Перекрытие имен. Соотношение понятий предок/потомок и тип/подтип. Примеры наследования.
8. Структура конструктора и деструктора. Виды конструкторов в C++.
9. Инкапсуляция. Роль инкапсуляции в ООП. Два взгляда на инкапсуляцию. Инкапсуляция и скрытие реализации.
10. Полиморфизм. Роль полиморфизма в ООП. Виды полиморфизма. Чистый полиморфизм, его роль и реализация. Механизм вызова виртуальных методов.
11. Шаблонный полиморфизм. Примеры применения чистого и шаблонного полиморфизма.
12. Структурированная обработка исключений (СОИ). Цели создания системы структурированной обработки исключений. Синтаксические конструкции порождения и перехвата исключений. Практика применения СОИ.
13. Понятие контейнерных классов. Свойства множеств, списков, таблиц, массивов. Реализация в STL.
14. XML как язык описания данных. Синтаксис и основные элементы XML. Пространства имен в XML документе.
15. Методы выборки данных из XML-документа в DOM-представлении. Синтаксис и применение XPath.
16. XML Схема. Элементы SDL. Ссылки на схемы в XML документах.
17. Контроль типов в XML. Примеры определения новых типов данных и проверки (валидации) XML документов.

