

Белорусский государственный университет



УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям

_____ О. Г. Прохоренко

«01» декабря 2022 г.

Регистрационный № УД – 11533/уч.

ВВЕДЕНИЕ В СПЕЦИАЛЬНОСТЬ

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:**

**1-31 03 07 Прикладная информатика (по направлениям)
направление специальности:**

**1-31 03 07-01 Прикладная информатика (программное обеспечение
компьютерных систем)**

2022 г.

Учебная программа составлена на основе образовательного стандарта высшего образования ОСВО 1-31 03 07-2021, типового учебного плана №G 31-1-030/пр-тип от 01.07.2021, учебных планов №G 31-1-216/уч. от 22.03.2022, №G 31-1-224/уч.ин. от 27.05.2022 г.

СОСТАВИТЕЛЬ:

А.Н. Вальвачев, доцент кафедры информационных систем управления Белорусского государственного университета, кандидат технических наук

РЕЦЕНЗЕНТЫ:

А.И. Кузьмич – заместитель директора научно-производственного предприятия “Горнэлектроникс”, кандидат технических наук

А.А. Воронов – ведущий научный сотрудник ОИПИ НАН Республики Беларусь, кандидат технических наук, доцент

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

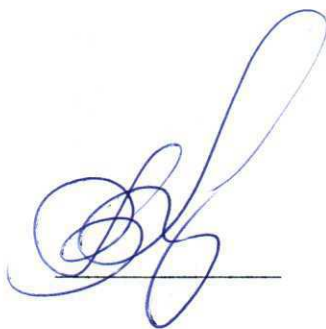
Кафедрой информационных систем управления Белорусского государственного университета

(протокол № 6 от 24.11.2022 г.)

Научно-методическим Советом БГУ

(протокол № 2 от 29.11.2022 г.)

Заведующий кафедрой



В.В. Краснопрошин

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная дисциплина «Введение в специальность» ориентирована на обучение студентов базовым знаниям, умениям и навыкам применения современных информационных технологий (далее – “ИТ”) для автоматизированного решения широкого спектра актуальных для Республики Беларусь и других стран задач из различных прикладных областей. В процессе обучения используются новейшие программные инструменты и методы, включая элементы искусственного интеллекта.

Основой для обучения тематике учебной дисциплины является курс информатики, изучаемый в средней школе и знания, полученные в первом семестре.

Цели и задачи учебной дисциплины

Цель преподавания учебной дисциплины «Введение в специальность» заключается в подготовке специалистов, способных ориентироваться в широком спектре актуальных задач и технологий для их решения. В результате обучения студент способен самостоятельно анализировать актуальные в настоящее время задачи, выбирать новейшие теоретические и практические механизмы решения и применять как индивидуально, так и при работе в территориально распределенной группе.

Задачи учебной дисциплины:

Для достижения цели в рамках дисциплины «Введение в специальность» решаются следующие задачи:

- изучение гибких методологий эффективной реализации проектов в составе территориально распределенной группе программистов;
- освоение методик разработки локальных консольных и графических приложений для различных прикладных направлений;
- изучение средств построения адаптивных web-приложений, инвариантных типу устройств;
- изучение методов выбора СУБД, релевантных требованиям проекта;
- приобретение навыков выбора облачных ресурсов, обеспечивающих оперативное развертывание и поддержку работы высоконагруженных систем;
- изучение методов построения систем поддержки принятия решений на основе искусственного интеллекта;
- изучение методов построения корпоративных систем;
- формирование навыков построения и применения средств защиты корпоративных программных и информационных ресурсов;
- изучение методов построения систем блокчейн для финансовых и других прикладных направлений.

При изложении курса акцент сделан на применение полученных знаний и навыков для решения актуальных для Республики Беларусь и других стран практических задач, включая: управление новыми видами организаций типа IoT, CPS, Smart Cities на основе искусственного интеллекта; принятие решений

на основе Big Data и Small Data; анализ данных средствами Data Science; оперативная оценка деятельности подразделений распределенных компаний; синтез безопасных ключей и паролей для персонала компании.

В итоге формируется теоретический и практический базис, необходимый студенту для упрощения и ускорения усвоения материала других учебных дисциплин, связанных с разработкой программных систем, а также при написании курсовых работ, дипломных проектов и самостоятельной работе в магистратуре и в качестве молодого специалиста.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием

Учебная дисциплина относится к компоненту учреждения высшего образования.

Программа составлена с учётом межпредметных **связей** с учебными дисциплинами.

Дисциплина «Введение в специальность» непосредственно связана с параллельно изучаемыми дисциплинами:

- «Разработка кросс-платформенных приложений», «Машинно-ориентированное программирование» модуля «Программирование»;
- «Алгебра и теория чисел», «Математический анализ» модуля «Высшая математика»;
- «Дискретная математика и математическая логика» модуля «Дискретная математика и алгоритмы».

Методы, излагаемые в указанных дисциплинах, используются для проектирования алгоритмов и программных приложений.

Сформированные при изучении дисциплины «Введение в специальность» компетенции являются основой для дальнейшего изучения дисциплин:

- «Промышленное программирование», «Технологии программирования для мобильных приложений» модуля «Программирование»;
- «Алгоритмы и структуры данных» модуля «Дискретная математика и алгоритмы»;
- «Модели данных и СУБД» модуля «Информационные системы».

Знания, полученные в учебной дисциплине, используются при изучении всех дисциплин специализации, при выполнении курсовых и дипломных работ, а также используются как инструментарий для моделирования и компьютерного решения задач ряда математических дисциплин, изучаемых на старших курсах.

Требования к компетенциям

Освоение учебной дисциплины «Введение в специальность» должно обеспечить формирование следующей **специализированной компетенции**:

СК-1. Осуществлять выбор профиля обучения, определять профессиональную склонности и пути дальнейшего развития в выбранной профессии в области информационных технологий.

В результате освоения учебной дисциплины студент должен:

знать:

- актуальные для Республики Беларусь прикладные задачи и современные подходы для их анализа и декомпозиции, обеспечивающие выбор релевантной операционной системы, языка программирования, IDE и СУБД;
- гибкие методики работы в территориально распределенной группе программистов и средства разработки консольных, графических, десктопных и web-приложений;
- методы и архитектуры для построения интеллектуальных корпоративных систем с интероперабельными модулями на основе многоагентного подхода;
- методы защиты от несанкционированного доступа к программным и информационным ресурсам.

уметь:

- выполнять декомпозицию задачи, выбирать эффективные средства для ее решения и распределять работу между участниками группы;
- использовать в процессе выполнения проектов в рамках методик Agile средства дистанционной работы Git, GitHub и языки C#, Python, HTML5-CSS3, а также Bootstrap;
- применять методы искусственного интеллекта для разработки корпоративных систем, минимизирующих участие человека;
- планировать и разрабатывать средства защиты информации.

владеть:

- методами самостоятельного комплексного решения задач;
- способами адаптации существующих решений к новым задачам;
- способами применения систем искусственного интеллекта для уменьшения рабочей нагрузки на корпоративных менеджеров;
- навыками применения интероперабельных программных модулей для решения различных прикладных задач;
- навыками имитационного моделирования.

Структура учебной дисциплины

Дисциплина изучается во 2-м семестре. Всего на изучение учебной дисциплины «Введение в специальность» отведено:

- для очной формы получения высшего образования – 90 часов, в том числе 34 аудиторных часа, из них: лекций – 34 часа.

Трудоёмкость учебной дисциплины составляет 3 зачётные единицы.

Форма текущей аттестации – зачет.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Разработка десктопных и веб-приложений

Тема 1.1. Актуальные задачи ИТ, средства и методы их решения

Актуальные проблемы информационного общества. ИТ как средство решения проблем. Перспективные направления развития ИТ. Схема Gartner. Классификация технологий ИТ.

Принципы разработки ПО в распределенной группе. Состав и роли участников группы. Методологии Agile. Контроль версий. Методика применения Git и GitHub.

Тема 1.2. Разработка приложений на C#

Обзор IDE MS VS, MS Core, MS Code. Интерфейс Visual Studio 2022. Методика разработки консольных приложений.

Методика разработки графических приложений на WinForms и WPF.

Тема 1.3. Разработка приложений на Python

Обзор IDE: PyCharm, Thonny, Anaconda. Интерфейс IDE Pycharm/Thonny. Типы данных, операторы Python. Подключение внешних библиотек. Этапы разработки прикладных систем.

Методика моделирования крупномасштабных систем. Моделирование на Python сложных систем типа “умный город”, “киберфизическая система”, “интернет вещей” на основе библиотек NumPy, Matplotlib.

Тема 1.4. Разработка веб-приложений

Назначение, структура Internet. Терминология. Архитектура и принципы работы Internet. Протоколы HTTP, HTTPS, Rest API. Обзор IDE. Классификация web-приложений. Структура web-приложений. Этапы разработки. Backend. Frontend. HTML5, CSS3. XML, JSON. Основные паттерны. Адаптивная верстка. Bootstrap. Установка. Система сеток. Начальный шаблон страницы. Разработка лэндинг-сайта.

Назначение JavaScript. Типы данных, операторы. Модели DOM, BOM.

Сравнительный анализ IDE Vue.js, React, Angular. Методика выбора IDE в зависимости от свойств решаемых задач.

Тема 1.5. Системы управления базами данных

Понятие БД и СУБД. Модели данных. Классификация СУБД. Язык SQL. Сравнительный анализ СУБД SQL (MySQL, PostgreSQL), СУБД noSQL (MongoDB). Экосистема Elastic Stack. Принципы работа с БД в C# и Python.

Раздел 2. Основы искусственного интеллекта

Тема 2.1. Основные сведения и библиотеки

Мозг, нейроны, нейронные сети. Модель формального нейрона Маккаллока-Питтса. Искусственный интеллект (ИИ): понятие и классификация. Искусственные нейронные сети. Нейроморфные системы. Агенты.

Библиотеки OpenCV, SciPy, Seaborn, Scikit Learn, TensorFlow, Keras.

Тема 2.2. Методика машинного обучения

Сбор данных для обучения. Подготовка данных. Построение (обучение) модели. Тестирование модели. Интеграция модели в среду применения.

Тема 2.3. Интеллектуальные системы

Назначение. Классификация. Архитектура. Методика разработки. Актуальные области применения. Системы IBM Watson, AlphaGo Zero, GPT-3. Перспективные области применения.

Раздел 3. Корпоративные системы

Тема 3.1. Корпоративные системы OLAP-BI

Назначение. Системы поддержки принятия решений. Классификация. Типовая архитектура OLAP-BI. Методика построения.

Тема 3.2. Облачные сервисы

Назначение, терминология. Классификация сервисов. Выбор сервиса, релевантного проекту.

Дата-центры. Сравнительный обзор Windows Azure, Google Cloud Platform, Amazon Web Services.

Тема 3.3. Контейнерные системы

Назначение, терминология. Виртуальные машины. Микросервисы. Контейнеры. Технологии контейнеризации. Docker. Структура образа. Оркестрация. Kubernetes.

Тема 3.4. Высоконагруженные системы

Большие данные. Назначение. Терминология. Архитектура. Понятие “нагрузка”. Пошаговый процесс создания, контроля и поддержки устойчивой работы высоконагруженных систем.

Тема 3.5. Системы блокчейн

Назначение, терминология. Распределенный реестр. Транзакции, блоки, цепочки, контракты, умные контракты. Криптовалюта.

Типовая архитектура. Методика разработки систем блокчейн. Перспективные направления применения блокчейн.

Тема 3.6. Кибербезопасность

Терминология. Вектор атак OWASP Top-10. Методы защиты от кибератак. Контроль доступа. Шифрование, дешифрование. Надежные пароли. Хэширование, ключи, подписи. Защита от вирусов.

Примеры программ для защиты информации в системах различного типа.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Дневная форма получения образования с применением электронных средств обучения (ДО)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСП	Форма контроля знаний
		Лекции	Практические	Семинарские занятия	Лабораторные	Иное		
1	2	3	4	5	6	7	8	9
	Введение в специальность	34						
1	Разработка десктопных и веб-приложений	16						
1.1.	Актуальные задачи ИТ, средства и методы их решения	2						Экспресс-опрос
1.2.	Разработка приложений на С#	4						Тест
1.3.	Разработка приложений на Python	4						Экспресс-опрос. Электронный тест.
1.4.	Разработка веб-приложений	4						Электронный тест.
1.5.	Системы управления базами данных	2						Собеседование
2	Основы искусственного интеллекта	6						
2.1.	Основные сведения и библиотеки	2						Экспресс-опрос.
2.2.	Методы машинного обучения	2						Дискуссия
2.3.	Интеллектуальные системы	2						Коллоквиум по разделу 2
3	Корпоративные системы	12						
3.1.	Корпоративные системы OLAP-BI	2						Тест
3.2.	Облачные сервисы	2						Экспресс-опрос
3.3.	Контейнерные системы	2						Электронный тест
3.4.	Высоконагруженные системы	2						Собеседование
3.5.	Системы блокчейн	2						Экспресс-опрос
3.6.	Кибербезопасность	2						Контрольная работа по разделу 3.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Виссия, Х. Э. Принятие решений в информационном обществе : учебное пособие / Х. Э. Виссия, В. В. Краснопрошин, А. Н. Вальвачев. — Санкт-Петербург : Лань, 2022. — 228 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL.

2. Воронов, М. В. Системы искусственного интеллекта : учебник и практикум для студентов высших учебных заведений, обучающихся по ИТ и математическим направлениям / М. В. Воронов, В. И. Пименов, И. А. Небаев. - Москва : Юрайт, 2022. - 256 с.

3. Козадаев, К. В. Интеллектуальные информационные технологии = Intelligent Information Technologies : пособие для студентов учреждений высшего образования, обучающихся по специальности 1-31 04 04 "Аэрокосмические радиоэлектронные и информационные системы и технологии" / К. В. Козадаев; БГУ. - Минск: БГУ, 2020. - 194 с. <https://elib.bsu.by/handle/123456789/247567>.

4. Остроух, А. В. Теория проектирования распределенных информационных систем / А. В. Остроух, А. В. Помазанов. - Санкт-Петербург ; Москва ; Краснодар : Лань, 2019. - 92 с. <https://e.lanbook.com/book/116390>.

Перечень дополнительной литературы

1. Вайгенд, Андреас. BIG DATA. Вся технология в одной книге = DATA for the People: How to Make Our Post-Privacy Economy Work for You / Андреас Вайгенд ; [пер. с англ. С. Богданова]. - Москва : Бомбора, 2019. - 381 с.

2. Гифт, Н. Прагматичный ИИ. Машинное обучение и облачные технологии = Pragmatic AI. An Introduction to Cloud-Based Machine Learning / Ной Гифт ; [пер. с англ. И. Пальти]. - Санкт-Петербург [и др.] : Питер, 2019. - 301 с. - URL: <https://ibooks.ru/bookshelf/365271>.

3. Мэтиз, Э. Изучаем Python: программирование игр, визуализация данных, веб-приложения. 3-е изд. - Санкт-Петербург : Питер, 2021. - 512 с. - URL: <https://ibooks.ru/bookshelf/371712>.

4. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML 5 / Робин Никсон ; [пер. с англ. Н. Вильчинского]. - 4-е изд. - Санкт-Петербург [и др.] : Питер, 2018. - 766 с. - <https://ibooks.ru/bookshelf/352698>.

5. Фостер, Д. Генеративное глубокое обучение. Творческий потенциал нейронных сетей. - Санкт-Петербург : Питер, 2021. - 352 с. - ISBN 978-5-4461-1566-2. - URL: <https://ibooks.ru/bookshelf/371724>.

6. Шолле, Ф. Глубокое обучение на Python. — (Серия «Библиотека программиста»). - Санкт-Петербург : Питер, 2021. - 400 с. - ISBN 978-5-4461-0770-4. – URL: <https://ibooks.ru/bookshelf/376833>.

Электронные ресурсы

1. Образовательная платформа DotNetPerls [Электронный ресурс]. – Режим доступа: <https://dotnetperls.com> – Дата доступа: 02.06.2022.
2. Основы программирования. Курс CS50. [Электронный ресурс]. – Режим доступа: <https://www.youtube.com/playlist?list=PLawfWYMUziZqyUL5QDLVbe3j5BKWj42E5> Дата доступа: 02.06.2022.
3. Введение в искусственный интеллект с Python. [Электронный ресурс]. – Режим доступа: <https://www.youtube.com/watch?v=WhiEGkxYIbk> – Дата доступа: 02.06.2022.

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Для диагностики компетенций в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: собеседование, экспресс-опрос, дискуссия.
2. Письменная форма: контрольная работа, коллоквиум, тесты.
3. Техническая форма: электронные тесты.

В качестве рекомендуемых технических средств диагностики используется сайты для дистанционной проверки знаний по языку Python, C#, HTML, CSS, Javascript: <https://kursy.guru/test/html/>, <https://kursy.guru/test/python/>, <https://kursy.guru/test/c-sharp/>, <https://kursy.guru/test/javascript/>

Формой текущей аттестации по дисциплине учебным планом предусмотрен зачет.

Точки контроля по текущей успеваемости формируются из расчета общего количества часов (зачетных единиц), выделенных на изучение дисциплины.

Критерием оценивания являются результаты экспресс-опросов, собеседований, дискуссий и выполнения заданий для самостоятельной работы обучающихся. Задания оцениваются с учетом следующих основных параметров:

- своевременное выполнение работы;
- полнота и правильность ответов на вопросы, заданные в ходе защиты работы.

Отметка «зачтено» выставляется студенту, который твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности.

Отметка «не зачтено» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в

изложении программного материала, допускающему в ответе или в решении задач грубые ошибки.

Рекомендуемая тематика контрольной работы и коллоквиума:

- 1) Коллоквиум по разделу 2 “Основы искусственного интеллекта”. Тема “Актуальные направления применения ИИ” (вопросы к коллоквиуму будут на портале <https://edufpmi.bsu.by/course/view.php?id=674>).
- 2) Контрольная работа по разделу 3 “Корпоративные системы”, тема “Моделирование систем OLAP-BI”.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используются следующие методы:

- **метод учебной дискуссии**, который предполагает участие студентов в целенаправленном обмене мнениями, идеями для предъявления и/или согласования существующих позиций по определенной проблеме. Использование метода обеспечивает появление нового уровня понимания изучаемой темы, применение знаний (теорий, концепций) при решении проблем, определение способов их решения.

- **метод группового обучения**, который представляет собой форму организации учебно-познавательной деятельности обучающихся, предполагающую функционирование разных типов малых групп, работающих как над общими, так и специфическими учебными заданиями.

В качестве технических средств для организации работы в рамках учебной дисциплины рекомендуется использовать Гарвардский курс CS50 (<https://habr.com/ru/company/vertdider/blog/403823/>).

- **практико-ориентированный подход**, который предполагает освоение содержания образования через решения практических задач; приобретение навыков эффективного выполнения разных видов профессиональной деятельности; ориентацию на генерирование идей, реализацию групповых студенческих проектов; использование процедур, способов оценивания, фиксирующих профессиональные компетенции.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные ресурсы: разместить на образовательном портале комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, методические указания к лабораторным занятиям, материалы текущего контроля и текущей аттестации,

позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательных стандартов высшего образования и учебно-программной документации, в т.ч. вопросы для подготовки к зачету и экзамену, задания, тесты, вопросы для самоконтроля и др., список рекомендуемой литературы, информационных ресурсов и др.).

При составлении заданий самостоятельной работы по учебной дисциплине необходимо предусмотреть возрастание их сложности: от заданий, формирующих достаточные знания по изученному учебному материалу на уровне узнавания, к заданиям, формирующим компетенции на уровне воспроизведения, и далее к заданиям, формирующим компетенции на уровне применения полученных знаний.

Таким образом, задания для самостоятельной работы по учебной дисциплине рекомендуется делить на три модуля:

- задания, формирующие достаточные знания по изученному учебному материалу на уровне узнавания;
- задания, формирующие компетенции на уровне воспроизведения;
- задания, формирующие компетенции на уровне применения полученных знаний.

Примерный перечень заданий

Задания на уровне узнавания

Задание 1. Сопоставить фрагменты блок-схем моделирования объектов различного типа на языках C# и Python, найти соответствия и отличия.

Задание 2. Сопоставить фрагменты моделирования воздействия на объект на языках C# и Python, найти соответствия и различия.

Задание 3. Сопоставить HTML-CSS тексты для лэндинг-сайтов из различных прикладных областей, найти соответствия и различия.

Задания на уровне восприятия

Задание 1. Разработать код процедуры на языках C#, Python, реализующий паттерн моделирования объекта.

Задание 2. Разработать код процедуры на языках C#, Python, реализующий паттерн моделирования воздействия на объект.

Задание 3. Разработать код для построения лэндинг-сайта в Bootstrap для реализации лэндинг-паттерна.

Задания на уровне применения полученных знаний

Задание 1. Разработать подсистему, формирующую модель “Умного дома” на основе кода паттерна моделирования объекта.

Задание 2. Разработать подсистему, формирующую модель воздействия землетрясения на “Умный дом” на основе кода паттерна моделирования воздействия.

Задание 3. Разработать адаптивный сайт “Продажа автомобилей” на основе лэндинг-кода паттерна.

Примерный перечень вопросов к зачету

1. Классификация прикладных систем. Понятие паттерна. Топ-5 паттернов.
2. Контроль версий. Этапы работы в группе с Git - GitHub
3. Сравнительная характеристика архитектур программных систем
4. Специфика разработки приложений на C#+Console.
5. Специфика разработки графических приложений на C#+WinForms.
6. Специфика разработки графических приложений на C#+WPF
7. Сравнительная характеристика IDE для Python.
8. Сравнительная характеристика научных библиотек для Python.
9. Назначение и задачи Data Science.
10. Структура WEB. Протоколы HTTP/HTTPS. Языки HTML5, CSS3.
11. Адаптивные сайты. Назначение и методика применения Bootstrap.
12. Назначение и специфика JavaScript.
13. Сравнительная характеристика IDE для JavaScript.
14. Понятие базы данных, СУБД, модели данных. Классификация СУБД.
15. Назначение языка SQL. Основные операторы.
16. Нейрон. Модель нейрона МакКаллока-Питтса. Нейросеть.
17. Определение и классификация искусственного интеллекта.
18. Понятие Deep Learning. Этапы обучения искусственной нейросети.
19. Интеллектуальные системы (ИС). Топ-3 ИС.
20. Унифицированная структура ИС (mD, DM, mKD, mVis).
21. Многоагентные системы. Понятия агента. Типовые агенты ИС.
22. Интеллектуальные системы в медицине, бизнесе, экологии.
23. Корпоративные системы поддержки принятия решений. OLAP-BI.
24. Понятие мониторинга. Типовая архитектура систем мониторинга.
25. Понятие IoT, CPS, Smart Cities. Этапы их моделирования.
26. Понятие облака, типы облаков, классификация облачных услуг.
27. Сравнительная характеристика Топ-5 сервисов облачных вендоров.
28. Понятие виртуализации. Виртуальная машин. Средства виртуализации.
29. Понятие контейнеризации. Назначение Docker. Структура образа.
30. Понятие оркестрации. Назначение Kubernetes.
31. Понятие кибербезопасности. Состав вектора угроз OWASP Top 10.
32. Средства контроля доступа к ресурсам. Безопасный пароль.
33. Симметричные и ассиметричные методы шифрования. Типы ключа.
34. Понятие блокчейн, транзакция, блок, хэш.
35. Понятие умный контракт в блокчейн. Структура блока.

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ
ПО ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ
С ДРУГИМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Разработка кросс-платформенных приложений	Информационных систем управления	Нет	Изменений не требуется (протокол № 6 от 24.11.2022 г.)
Машинно-ориентированное программирование	Информационных систем управления	Нет	Изменений не требуется (протокол № 6 от 24.11.2022 г.)
Модели данных и СУБД	Информационных систем управления	Нет	Изменений не требуется (протокол № 6 от 24.11.2022 г.)
Промышленное программирование	Информационных систем управления	Нет	Изменений не требуется (протокол № 6 от 24.11.2022 г.)
Технологии программирования для мобильных приложений	Технологий программирования	Нет	Изменений не требуется (протокол № 6 от 24.11.2022 г.)

**ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ
ПО ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ
на ____ / ____ учебный год**

№№ ПП	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры информационных систем управления (протокол № ____ от _____ 200_ г.)

Заведующий кафедрой

_____ (степень, звание) _____ (подпись) _____ (И.О.Фамилия)

УТВЕРЖДАЮ
Декан факультета

_____ (степень, звание) _____ (подпись) _____ (И.О.Фамилия)