

**Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Национальный исследовательский университет «МЭИ»**

Направление подготовки/специальность: 09.04.01 Информатика и вычислительная техника

Наименование образовательной программы: Автоматизированные системы обработки информации и управления

Уровень образования: высшее образование - магистратура

Форма обучения: Очная

**Оценочные материалы
по дисциплине
Введение в технологию блокчейн**

**Москва
2021**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ РАЗРАБОТАЛ:

Преподаватель

(должность)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Оцоков Ш.А.
	Идентификатор	R1955ce2a-OtsokovShA-1e5b4243

(подпись)

Ш.А. Оцоков

(расшифровка
подписи)

СОГЛАСОВАНО:

Руководитель
образовательной
программы

(должность, ученая степень, ученое
звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Вишняков С.В.
	Идентификатор	R35b26072-VishniakovSV-02810d9

(подпись)

С.В.
Вишняков

(расшифровка
подписи)

Заведующий
выпускающей кафедры

(должность, ученая степень, ученое
звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Вишняков С.В.
	Идентификатор	R35b26072-VishniakovSV-02810d9

(подпись)

С.В.
Вишняков

(расшифровка
подписи)

ОБЩАЯ ЧАСТЬ

Оценочные материалы по дисциплине предназначены для оценки: достижения обучающимися запланированных результатов обучения по дисциплине, этапа формирования запланированных компетенций и уровня освоения дисциплины.

Оценочные материалы по дисциплине включают оценочные средства для проведения мероприятий текущего контроля успеваемости и промежуточной аттестации.

Формируемые у обучающегося компетенции:

1. ПК-2 Способен осуществлять проектирование вычислительных комплексов и систем, включая разработку аппаратного, программного обеспечения, системную интеграцию, ввод в эксплуатацию

ИД-2 Демонстрирует знание теории баз данных, включая перспективные технологии обработки больших данных

и включает:

для текущего контроля успеваемости:

Форма реализации: Защита задания

1. Защита лабораторной работы "Сложный смарт-контракт, оптимизация" (Лабораторная работа)
2. Защита цикла лабораторных работ по теме "Основы solidity" (Лабораторная работа)
3. Разработка сложного смартконтракта (Расчетно-графическая работа)

Форма реализации: Компьютерное задание

1. Блокчейн, транзакции (Контрольная работа)

Форма реализации: Письменная работа

1. Простой смарт-контракт (Контрольная работа)

БРС дисциплины

2 семестр

Раздел дисциплины	Веса контрольных мероприятий, %					
	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5
	Срок КМ:	3	7	11	14	16
Основы блокчейна						
Основы блокчейна		+				
Архитектура узла в сети блокчейна						
Архитектура узла в сети блокчейна		+				
Консенсус, криптовалюта и майнинг						
Консенсус, криптовалюта и майнинг			+			

Смарт-контракты в сети Ethereum на языке Solidity					
Смарт-контракты в сети Ethereum на языке Solidity		+	+		
Программирование на solidity					
Программирование на solidity			+		
Библиотека Мока и др для тестирования смарт-контрактов					
Библиотека Мока и др для тестирования смарт-контрактов				+	+
Взаимодействие со смарт-контрактом					
Взаимодействие со смарт-контрактом					+
Разработка смарт-контракта «Лотерея»					
Разработка смарт-контракта «Лотерея»					+
Разработка смарт-контракта «Инвестиционная площадка»					
Разработка смарт-контракта «Инвестиционная площадка»					+
Введение в DApp					
Введение в DApp					+
Вес КМ:	20	20	30	20	10

\$Общая часть/Для промежуточной аттестации\$

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

I. Оценочные средства для оценки запланированных результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Индекс компетенции	Индикатор	Запланированные результаты обучения по дисциплине	Контрольная точка
ПК-2	ИД-2ПК-2 Демонстрирует знание теории баз данных, включая перспективные технологии обработки больших данных	Знать: теоретические основы блокчейна понятийный аппарат блокчейна принципы функционирования блокчейна Уметь: разрабатывать структуру смарт-контракта, его поля и методы разрабатывать смарт-контракты взаимодействовать со смарт-контрактом	Блокчейн, транзакции (Контрольная работа) Защита цикла лабораторных работ по теме "Основы solidity" (Лабораторная работа) Простой смарт-контракт (Контрольная работа) Защита лабораторной работы "Сложный смарт-контракт, оптимизация" (Лабораторная работа) Разработка сложного смартконтракта (Расчетно-графическая работа)

II. Содержание оценочных средств. Шкала и критерии оценивания

КМ-1. Блокчейн, транзакции

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 20

Процедура проведения контрольного мероприятия: Проводится в компьютерном классе, время 45 мин

Краткое содержание задания:

Используются вопросы про терминологию блокчейна

Контрольные вопросы/задания:

Знать: понятийный аппарат блокчейна	<ol style="list-style-type: none">1. Как происходит подтверждение транзакции?2. Какие существуют алгоритмы консенсуса?3. За счёт чего достигается неизменность в блокчейне?4. Зачем нужен приватный блокчейн?5. Почему транзакции занимают время?
-------------------------------------	---

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-2. Защита цикла лабораторных работ по теме "Основы solidity"

Формы реализации: Защита задания

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 20

Процедура проведения контрольного мероприятия: Проводится в компьютерном классе, время 45 мин

Краткое содержание задания:

Задача (вариант №1)

Разработка смарт контракта «Сотрудник» с методами: вывод информации о сотруднике, внесение информации о сотруднике. Информация на ваш выбор.

Задача (вариант №2)

Реализовать смарт контракт «Перевозка» с методами добавить перевозку груза и получить информацию о перевозке. Содержание полей класса на ваш выбор
После разработки смарт-контракта протестировать его работы в среде remix.

Контрольные вопросы/задания:

<p>Знать: принципы функционирования блокчейна</p>	<p>1.Как можно генерировать псевдо случайные числа в смарт контракте? 2. Что такое событие где применяются и как создаются? 3.Какие вы знаете модификаторы доступа к переменной 4.Что такое газ?</p>
---	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-3. Простой смарт-контракт

Формы реализации: Письменная работа

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 30

Процедура проведения контрольного мероприятия: Проводится в компьютерном классе, время 45 мин

Краткое содержание задания:

Даются вопросы по синтаксису Soliditty и задания

Контрольные вопросы/задания:

<p>Знать: теоретические основы блокчейна</p>	<p>1.Приведите пример создания структуры? 2.Чем отличается отображение от массивов? 3.В чём недостатки использования массивов в смарт-контрактах?</p>
--	--

--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-4. Защита лабораторной работы "Сложный смарт-контракт, оптимизация"

Формы реализации: Защита задания

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 20

Процедура проведения контрольного мероприятия: Ориентирована на 1 ч. 30, проходит в лаборатории

Краткое содержание задания:

Задача (вариант №1)

Разработка смарт-контракта «Оплата сотрудникам». В смарт-контракте реализовать:

1. Конструктор, который сохраняет адрес создателя и общую сумму вознаграждения, которая будет делиться между сотрудником.
2. Функция регистрации, при вызове которой происходит сохранение информации о сотруднике в смарт-контракте, а именно: адрес, имя, возраст и процент вознаграждения, которое он получает.
3. Функция выплаты вознаграждения. Данную функцию может вызвать только создатель. При вызове данной функции вознаграждение делится в соответствии с процентом вознаграждения каждому зарегистрированному сотруднику.

Вознаграждение виртуальное, перевод криптовалюты не требуется.

Задача (вариант №2)

Реализовать смарт контракт «Игра палочки» таким образом, чтобы могли играть два игрока. В смарт-контракте есть функция для регистрация игрока, в которой записывается адрес игрока.

Изначально есть 10 палочек.

На каждом ходу в смарт-контракт записывается текущее количество оставшихся палочек и игрок вводит количество палочек, которое хочет взять (делает ход). После каждого хода меняется очерёдность игроков (т.е. каждый игрок не может делать два последовательных хода) . После того как игрок взял палочки, общее кол-во палочек уменьшается. Создатель может вызвать функцию, которая должна возвращать одно из трёх значений («игра не началась», «игра идет», «игра завершена: победитель: { }»), вместо { } подставляется адрес победителя.)

После разработки смарт-контракта протестировать его работы в среде remix.

Контрольные вопросы/задания:

Уметь: разрабатывать смарт-контракты	1.Написать смартконтракт для добавления пользователей и выбора случайного пользователя
--------------------------------------	--

Описание шкалы оценивания:*Оценка: 5**Нижний порог выполнения задания в процентах: 70**Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно**Оценка: 4**Нижний порог выполнения задания в процентах: 60**Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач**Оценка: 3**Нижний порог выполнения задания в процентах: 50**Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено***КМ-5. Разработка сложного смартконтракта****Формы реализации:** Защита задания**Тип контрольного мероприятия:** Расчетно-графическая работа**Вес контрольного мероприятия в БРС:** 10**Процедура проведения контрольного мероприятия:** Ориентирована на несколько недель.**Краткое содержание задания:**

Темы для расчетно-графической работы

Варианты:

1. Разработка распределенного хранилища текстовых данных небольшой размерности в блокчейне
2. Разработка смартконтракта и веб-приложения для голосования.
3. Разработка смартконтракта и веб-приложения для оплаты в интернет-магазинах в криптовалюте (для иностранных покупателей).
4. Разработка тестовых смартконтрактов на основе приватного блокчейна
5. Разработка игры «Монополия» на основе смарт-контракта Ethereum в тестовой сети Ethereum.
6. Разработка смартконтракта и веб-приложения для взаимодействия с ним для оплаты коммунальных услуг.
7. Разработка платформы для хранения истории болезней Ethereum
8. Разработка смарт-контракта для крипто-ломбарда
9. Разработка собственного блокчейна на основе протокола MQTT
10. Исследование блокчейна HyperLadger и его возможностей
11. Изучение и тестирование платформы inno chain
12. Разработка веб-приложения с личным кабинетом для пожертвований в благотворительные организации

Контрольные вопросы/задания:

Уметь: взаимодействовать со смарт-контрактом	1.Продемонстрируете работу с вашим смартконтрактом в среде Remix в тестовой сети блокчейна?
Уметь: разрабатывать структуру	1.Напишите веб-приложение для взаимодействия со

смарт-контракта, его поля и методы	смартконтрактом 2.Протестировать смарт-контракт в среде Remix ethereum
------------------------------------	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

2 семестр

Форма промежуточной аттестации: Экзамен

Пример билета

Пример билета

Вопрос 1

Блокчейн достоинство технологии блокчейна, типы блокчейнов. Примеры применения блокчейна. Криптовалюта. Блокчейны Ethereum, Bitcoin.

Вопрос 2.

Электронная цифровая подпись этапы формирования и проверки подлинности подписи.

Задача

Реализовать смарт-контракт определяющий максимальную сумму, которую внесли в смартконтракт и адрес переводящего эту сумму.

Процедура проведения

Время, условия допуска, порядок пересдачи

1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ИД-2ПК-2 Демонстрирует знание теории баз данных, включая перспективные технологии обработки больших данных

Вопросы, задания

1. Блокчейн достоинство технологии блокчейна, типы блокчейнов. Примеры применения блокчейна. Криптовалюта. Блокчейны Ethereum, Bitcoin

2. Блок. Цепочка блоков. Примеры консенсусов. Транзакции. Комиссия за транзакцию.

Время подтверждения транзакции

3. Хэш функции. Требования к хэш-функциям. Примеры.

4. Электронная цифровая подпись этапы формирования и проверки подлинности подписи

5. Протокол датирования времени

6. Работа с метамаском. Перевод криптовалюты, пополнение счёта. Создания нескольких счетов

7. Алгоритм hashcash

8. Алгоритм датирования без доверия

9. Смарт-контракт. Пример простейшего смарт-контракта

10. Типы данных в языке solidity. Простейшие операции с ними. Глобальная переменная msg

11. Типы функции в solidity. Привести примеры

12. Типы памяти для хранения переменных. События

13. Газ. Лимит газа. Оценка стоимости выполнения кода смарт-контракта

14. Конструкция языка solidity массивы статические и динамические, отображения.

Примеры

15. Тестирование смарт-контрактов

16. Смарт-контракт с фабрикой контрактов

17. Токены. Стандарт erc20

Материалы для проверки остаточных знаний

1.6 Тип блокчейна, предоставляющий доступ любому человеку ко всей цепочке транзакций?

Ответы:

- (1) публичный блокчейн
- (2) блокчейн консорциумов
- (3) частный блокчейн
- (4) все варианты, перечисленные выше

Верный ответ: (1) публичный блокчейн

2.7 Какой язык программирования используется для платформы Ethereum?

Ответы:

- (1) MetaMask
- (2) Solidity
- (3) Mist
- (4) JavaScript

Верный ответ: (2) Solidity

3.8 Какой браузер является связующим звеном между браузером и блокчейном?

Ответы:

- (1) Chrome
- (2) Mist
- (3) MetaMask
- (4) Opera

Верный ответ: (3) MetaMask

4.9. Какой из алгоритмов хэширования используется во многих блокчейнах?

Ответы:

- sha-512
- md5
- md2
- sha-256

Верный ответ: sha-256

5.10 Какой блокчейн поддерживает смарт-контракты?

Ответы:

1. (1) Биткойн
2. (2) Ethereum
3. (3) Hyperledger Fabric

Верный ответ: (2) Ethereum (3) Hyperledger Fabric

6.11 Что используется для доступа к кошельку?

Ответы:

- (1) Закрытый ключ
- (2) Открытый ключ
- (3) Пароль

Верный ответ: (1) Закрытый ключ

7.12 Что такое ERC20?

Ответы:

- (1) Стандарт токена
- (2) Кошелек
- (3) Смарт-контракт

Верный ответ: (1) Стандарт токена

8.13 "msg.data", "msg.sender", "msg.gas" есть?

Ответы:

- (1) Глобальная переменная
- (2) Локальная переменная
- (3) Переменная ремикса

Верный ответ: (1) Глобальная переменная

9.14. Вы можете развернуть смарт-контракт через Remix IDE

Ответы:

- (1) Правда
- (2) Ложь

Верный ответ: (1) Правда

10.15. Смарт-контракт - это

Ответы:

- (1) Децентрализованное приложение
- (2) Кошелек
- (3) Аппаратное обеспечение

Верный ответ: (1) Децентрализованное приложение

11.16 В чём особенность смарт-контракта Ethereum по сравнению с другими программами?

Ответы:

- (1) можно остановить смарт-контракт любому участнику сети
- (2) нельзя взломать (при условии правильного кода контракта)
- (3) можно изменить код
- (4) нельзя изменить код и изменить данные
- (5) нельзя изменить код, но можно изменить данные

Верный ответ: (5) нельзя изменить код, но можно изменить данные (2) нельзя взломать (при условии правильного кода контракта)

12.17 Может ли смарт-контракт взаимодействовать с другими смарт-контрактами?

Ответы:

- (1) Да
- (2) Нет

Верный ответ: (1) Да

13.18 Является ли Solidity языком с динамической или статической типизацией? (т.е. необходимо определить типы переменных)

Ответы:

- (1) с динамической
- (2) статической типизацией

Верный ответ: (2) статической типизацией

14.19 Какие переменные a, b локальные или глобальные contract A

```
{  
    uint a;  
    functions function foo()  
    {    uint b; }  
}
```

Ответы:

- (1) a - локальная, b - глобальная переменная

(2) a - глобальная , b - локальная переменная

Верный ответ: (2) a - глобальная , b - локальная переменная

15.20 Кто может читать частные и общедоступные переменные?

Ответы:

(1) частные переменные могут быть прочитаны только функциями внутри смарт-контракта

 публичные переменные могут быть прочитаны кем угодно

(2) публичные переменные могут быть прочитаны кем угодно, частные переменные могут быть прочитаны только создателем смарт-контракта

(3) частные переменные и глобальные переменные могут быть прочитаны только функциями внутри смарт-контракта

Верный ответ: (1) частные переменные могут быть прочитаны только функциями

внутри смарт-контракта публичные переменные могут быть прочитаны кем угодно

16.21 Как тогда поступать с личными данными?

Ответы:

(1) **Вы либо не помещаете личные данные в блокчейн, либо размещаете хеши**

(2) **Можно размещать личные данные в блокчейн, т.к. он защищён**

Верный ответ: (1) Вы либо не помещаете личные данные в блокчейн, либо размещаете хеши

17.22 Как добавить данные в массив, объявленный как переменная состояния?

Ответы:

(1) **`uint[] a; function add(uint newEntry) external { add.push(a); }`**

(2) **`uint[] a; function add(uint newEntry) external { a.add(newEntry) }`**

Верный ответ: (1) `uint[] a; function add(uint newEntry) external { add.push(a); }`

18.5 Какие характеристики присущи технологии блокчейн?

Ответы:

(1) защищенность методами шифрования данных

(2) состоит из цепочки блоков в хронологическом порядке

(3) централизованная технология

(4) имеет распределенную облачную платформу

Верный ответ: (1) защищенность методами шифрования данных (2) состоит из цепочки блоков в хронологическом порядке (4) имеет распределенную облачную платформу

19.23 Как добавить данные в отображение, объявленное как переменная состояния?

Ответы:

(1) **`mapping(address => bool) a; function add(address addr) external { a[addr] = true; }`**

(2) **`mapping(address => bool) a; function add(address addr) external { a.push(addr); }`**

Верный ответ: (1) `mapping(address => bool) a; function add(address addr) external { a[addr] = true; }`

20.4 Для чего реестр сохраняется в блокчейне?

Ответы:

(1) **Сопоставление между владельцем и объектом**

(2) **Идентификация принадлежащих объектам**

(3) **Идентификация собственников**

Верный ответ: (1) Сопоставление между владельцем и объектом

21.2 Из чего состоит блок в блокчейне?

Ответы:

- (1) транзакции
- (2) из хэшей
- (3) отметок времени
- (4) из всех этих пунктов

Верный ответ: (4) из всех этих пунктов

22.1 Что такое блокчейн?

Ответы:

- (1) публичный счет, на котором учитываются все транзакции за всю историю его существования
- (2) технология подключения к единой сети
- (3) децентрализованная технология, защищенная надежными методами шифрования
- (4) транзакции, обрабатываемые единым центром

Верный ответ: (3) децентрализованная технология, защищенная надежными методами шифрования

23.3 Где находится центральный сервер биткойнов?

Ответы:

- (1) Вашингтон
- (2) Неизвестное местоположение
- (3) Лондон
- (4) Ни один из этих

Верный ответ: (4) Ни один из этих

24.24 Как выдать ошибку если a не равен b , с сообщением об ошибке `My error message`?

Ответы:

- (1) `require(a != b, 'My error message')`
- (2) `require(a == b, 'My error message')`

Верный ответ: (2) `require(a == b, 'My error message')`

II. Описание шкалы оценивания

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Работа выполнена в рамках "продвинутого" уровня. Ответы даны верно, четко сформулированные особенности практических решений

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Работа выполнена в рамках "базового" уровня. Большинство ответов даны верно. В части материала есть незначительные недостатки

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Работа выполнена в рамках "порогового" уровня. Основная часть задания выполнена верно. на вопросы углубленного уровня

III. Правила выставления итоговой оценки по курсу

Оценка выставляется по совокупности оценок по отдельным вопросам и задаче