

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Б1.В.ДВ.01.01 «Программирование на блокчейн»

Автор: Сенотов Дмитрий Игоревич, BANKEX Foundation School.

Код и наименование направления подготовки, профиля: 38.04.02 Менеджмент
«Прорывные финансовые и цифровые технологии»

Квалификация (степень) выпускника: Магистр

Форма обучения: очная

Цель освоения дисциплины:

Цель курса – разобраться в устройстве децентрализованных приложений с использованием технологии блокчейн.

В результате изучения дисциплины «Программирование на блокчейн» студенты должны:

- знать основы теоретической составляющей технологии блокчейн;
- понимать фундаментальные идеи и отличия различных блокчейнов;
- уметь создавать децентрализованные приложения с помощью языка программирования Solidity.

План курса:

Тема 1. Базовая теория блокчейна

Введение: что будет в курсе. Биткоин. Централизованный реестр. Цифровые подписи. Временные отметки. Система utxo. Децентрализованный реестр. P2P-сети. Как достичь консенсуса. Хеш-функции. Proof of work. Проблема двойных трат. Блоки и цепочки блоков. Дерево Меркла. Сложность майнинга. Награда за создание блока. Комиссии за транзакции. SEGWIT. Сеть Lightning.

Тема 2. Эфириум

Основные различия Эфириума и Биткоина. Отличие системы utxo от балансов. Базовая теория Эфириума. Виды узлов. Транзакции. Газ. Пользовательский аккаунт. Metamask. Основная сеть, тестовые сети. Faucet. Теория смарт-контрактов. Аккаунт смарт-контракта. Газ в смарт-контрактах. Создание контракта. Языки для написания смарт-контрактов (Solidity). Oracles. Bytecode, OPcode, ABI. Виртуальная машина Эфириума (EVM). Различные способы хранения данных. Stack-machine.

Тема 3. Другие платформы

Различные алгоритмы консенсуса. EOS. RAM, CPU. IOTA. Ripple. Разница между ними. Причины уязвимостей в IOTA. Централизация в Ripple. Различные применения смарт-контрактов. DGoods.

Тема 4. Solidity in Remix

Remix - онлайн среда разработки для Solidity.

Основы Solidity. Version pragma, import, комментарии. Переменные состояния.

Основные типы. Конструкторы. Функции, типы функций. Настройки Remix.

Выпуск смарт-контрактов в Remix. Вызов функций. Повторный запуск контракта. Разные виды вызова выполнения функций (вызов, отправка транзакций).

Подробности Solidity. Типы (struct, enum, mapping). Модификаторы view и pure. Видимость функций. Модификатор payable, fallback функции.

Продвинутые смарт-контракты. Свойства блока и транзакций. Обработка ошибок (assert, require, revert и exceptions). Модификаторы функций. Наследование, интерфейсы. События. Библиотеки. Calls, delegated calls.

Разбор существующих имплементаций.

Тема 5. Среды разработки

Среды разработки смарт-контрактов. Настройка среды. Структура проекта. Truffle framework. Тестовые фреймворки. Генерация ключей. Подпись/отправка транзакций. Библиотека web3. Компиляция. Тестирование. Mocha. Запуск контрактов с web3. Запуск контрактов с infura.

Пример приложения.

В результате освоения дисциплины у студентов должны быть сформированы:

Код компетенции	Наименование компетенции	Код этапа освоения компетенции	Наименование этапа освоения компетенции
ПК-4	способность управлять организациями, подразделениями, группами (командами) сотрудников, проектными сетями	ПК-4.1	способен управлять организациями, подразделениями, группами (командами) сотрудников, проектными сетями
ПК ОС-13	способность принимать управленческие решения на основе анализа структурированных и неструктурированных данных	ПК ОС-13.1	способен принимать управленческие решения на основе анализа структурированных и неструктурированных данных
ОК-1	способность к абстрактному мышлению, анализу и синтезу	ОК-1.1	обладает абстрактным мышлением, анализом и синтезом

Формы и методы текущего контроля и промежуточной аттестации:

В ходе реализации дисциплины используются следующие методы текущего контроля и успеваемости обучающихся:

Опрос, тест, диспут, итоговый зачет с оценкой.

Основная литература:

1. *Nakamoto S.* Bitcoin: A Peer-to-Peer Electronic Cash System. – 2008. URL: <https://bitcoin.org/bitcoin.pdf> (accessed 29 October 2018)
2. *Antonopoulos A. M.* Mastering Bitcoin: Programming the open blockchain. – "O'Reilly Media, Inc.", 2017.
3. *Narayanan A. et al.* Bitcoin and cryptocurrency technologies: a comprehensive introduction. – Princeton University Press, 2016.
4. *Buterin V. et al.* A next-generation smart contract and decentralized application platform //white paper. – 2014. URL: <https://github.com/ethereum/wiki/wiki/White-Paper> (accessed 29 October 2018)
5. *Wood G.* Ethereum: A secure decentralised generalised transaction ledger //Ethereum project yellow paper. – 2014. – T. 151. – C. 1-32. URL: <https://ethereum.github.io/yellowpaper/paper.pdf> (accessed 29 October 2018)

Antonopoulos A. M., Wood G. Mastering Ethereum – 2018. URL: <https://github.com/ethereumbook/ethereumbook> (accessed 29 October 2018)