План проведения лабораторных и практических занятий по дисциплине Блокчейн

1. Цель работ

- Изучение основ моделирования систем на основе технологии блокчейн.
- Пошаговое построение сети блокчейн в процессе выполнения работ.

2. Инструмент разработки

Моделирование сети блокчейн планируется производить с использованием языка программирования Python.

3. Описание работ

Первая работа является вступительной и нацелена на изучение основ языка программирования применительно к решению дальнейших задач. В последующих работах последовательно создаётся модель сети блокчейн. При этом подробно описываются основные составляющие данной технологии. С каждой следующей работой, модель дополняется новыми функциями и свойствами.

3.1. Вступительная работа.

Изучение основ написания и запуска кода на языке программирования Python. Теоретическая часть работы сопровождается примерами написания класса, выполняющего функции калькулятора. Рассматриваются простейшие операции: сложение, вычитание, умножение и деление двух чисел.

Основной задачей обучающегося в данной работе является дополнение кода класса, описывающего калькулятор, новой операцией по формуле в зависимости от варианта задания.

3.2. Основные работы

Последующие лабораторные и практические работы нацелены на изучение самой технологии блокчейн.

Темы работ:

- Хеширование;
- Дерево Меркла;
- Обобщённая структура блокчейн;
- Консенсус;
- Эллиптические кривые;
- Алгоритм ECDSA.

3.3. Пример реализации процедуры хеширования

В лабораторной работе, посвящённой теме хеширования, обучающемуся необходимо провести хеширование с использованием различных алгоритмов и сравнить полученные результаты.

В примере, приведенном ниже, производится преобразование текстовой строки символов «hello» при помощи библиотеки python hashlib с использованием двух алгоритмов хеширования SHA256 и MD5.

```
>>> from hashlib import sha256, md5
>>> text = 'hello'.encode('utf-8')
>>> sha256(text).hexdigest()
'2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824'
>>> md5(text).hexdigest()
'5d41402abc4b2a76b9719d911017c592'
```

Изменение любого символа в строке приводит к изменению результатов хеширования, как показано в следующем примере, где производится обработка предыдущей строки с искажением одного символа.

```
>>> text = 'hallo'.encode('utf-8')
>>> sha256(text).hexdigest()
'd3751d33f9cd5049c4af2b462735457e4d3baf130bcbb87f389e349fbaeb20b9'
>>> md5(data).hexdigest()
'598d4c200461b81522a3328565c25f7c'
```

Результаты, полученные на основе выполнения различных алгоритмов хеширования, должны стать предметом анализа с последующими выводами.

3.4. Эллиптические кривые

Практические работы на тему эллиптических кривых предполагают проведение обучающимися расчетов вручную в зависимости от варианта задания.

В качестве примера приводится сложение двух точек P=(12,12) и Q=(7,9) эллиптической кривой $y^2 = x^3 + 7x + 11$ над полем GF(23). Ниже представлено решение данной задачи:

$$l=(12-7)(12-9)^{-1} \bmod 23=19$$
$$x_R=19^2-12-7 \bmod 23=20$$
$$y_R=-12+19(12-20) \bmod 23=20$$
$$P+Q=R=(20,20)$$

Другим примером работы с эллиптическими кривыми является удвоение точки P=(12,11) на той же кривой:

$$l=(3\times12^2+7)\times(2\times11)^{-1} \bmod 23=21$$
$$x_R=21^2-2\times12 \bmod 23=3$$
$$y_R=-11+21\times(12-3) \bmod 23=17$$
$$2P=R=(3,17)$$

В дальнейшем требуется также проверить, принадлежат ли полученные точки данной эллиптической кривой.

3.5. Алгоритм ECDSA

Работы на тему применения алгоритма ECDSA предполагают использование эллиптических кривых в задачах цифровой подписи при построении системы на основе технологии блокчейн. Обучающиеся генерируют ключи на основе исходных данных в соответствии с вариантом задания. Далее производят подпись данных и подтверждают ее, используя алгоритм ECDSA.