

Міністерство освіти і науки України  
Львівський національний університет імені Івана Франка  
Факультет прикладної математики та інформатики  
Кафедра прикладної математики

## Дипломна робота

### **Блокчейн технологія Використання смарт-контрактів**

Студент групи ПМп-42:  
**Спасник Юрій Романович**,  
спеціальність 113-прикладна математика

Науковий керівник:  
професор  
**Дияк Іван Іванович**

Рецензент:

---

Львів–2023

# Зміст

|  |           |
|--|-----------|
| <b>Вступ</b>   | <b>3</b>  |
| <b>1 Робота блокчейну та смарт-контрактів</b>              | <b>4</b>  |
| 1.1 Вигляд блоку в блокчейні . . . . .                     | 5         |
| 1.2 Головні особливості блокчейн технологій . . . . .      | 7         |
| <b>2 Використання блокчейну</b>                            | <b>8</b>  |
| 2.1 Застосування блокчейну у повсякденному житті . . . . . | 8         |
| <b>3 Недоліки блокчейну та смарт-контрактів</b>            | <b>10</b> |
| 3.1 Недоліки блокчейну . . . . .                           | 10        |
| 3.2 Недоліки смарт-контрактів . . . . .                    | 11        |
| <b>4 Програмна реалізація</b>                              | <b>13</b> |
| 4.1 Створення конфігурації . . . . .                       | 13        |
| 4.2 Вибір конфігурації користувачем . . . . .              | 16        |
| 4.3 Внесення криптовалюти користувачем . . . . .           | 17        |
| 4.4 Отримання винагород користувачем . . . . .             | 18        |
| 4.5 Розрахунок винагороди для користувачів . . . . .       | 18        |
| <b>Висновок</b>  | <b>20</b> |
| <b>Список використаних джерел</b>                          | <b>21</b> |

# Вступ

Блокчейн - це розподілена база даних, яка містить впорядковані записи. Термін "блокчейн" перекладається з англійської як "ланцюжок блоків і ця технологія була запропонована в 2008 році Сатоші Накамото (ім'я, що використовується псевдонімом однієї людини або групи людей). Відкритий децентралізований код, доступний для всіх користувачів. Цифрові записи або транзакції об'єднуються в блоки, які потім криптографічно і хронологічно зв'язуються між собою, використовуючи складні математичні алгоритми.

У своїй дипломній роботі я збираюся розробити смарт-контракт за допомогою блокчейн технології. Смарт-контракт - це цифровий еквівалент звичайних контрактів, який може виконувати певні дії відповідно до умов, встановлених сторонами. Смарт-контракти забезпечують безпеку, оскільки вони є децентралізованими і не контролюються жодною окремою особою, або організацією. Після написання та публікації контракту, внесення змін до нього стає неможливим, а весь код контракту доступний для перевірки користувачами. Ці програми дозволяють безпечно обмінюватися криптовалютою, грошима, товарами та послугами.

## Розділ 1

# Робота блокчейну та смарт-контрактів

Технологія блокчейн використовує розподілену базу даних, щоб гарантувати безпеку, надійність і прозорість транзакцій. Основним принципом блокчейну є створення безперервного ланцюжка блоків, де кожен блок містить набір транзакцій або записів. Кожен новий блок у блокчейні має хеш (зашифрований код) попереднього блоку, що встановлює зв'язок між ними. Це унеможливорює зміну попередніх блоків без виявлення незаконної поведінки, оскільки будь-яка зміна в одному блоці потребує перерахунку хешів усіх наступних блоків. Це робить блокчейн надзвичайно стійким до підробки даних. Крім того, блокчейн працює в децентралізованому середовищі, де жодна центральна влада не контролює всю систему. Натомість мережа блокчейну складається з вузлів (комп'ютерів або серверів), які мають копію всієї бази даних. Кожен вузол має повноваження перевіряти, записувати та підтверджувати транзакції, що забезпечує децентралізацію системи. В блокчейні також використовується криптографія для забезпечення безпеки та конфіденційності. Кожна транзакція в блокчейні підписується криптографічним ключем, який гарантує її автентичність і не може бути змінений без

відповідного дозволу. Для захисту конфіденційності даних також можна використовувати різні протоколи шифрування.

Основний принцип роботи смарт-контрактів полягає в автоматичному забезпеченні виконання умов контракту без необхідності залучення посередників. Давайте розглянемо, як працює смарт-контракт:

- 1) Перше, що потрібно зробити це написати смарт-контракт, його можна створити на різних мовах програмування, таких як Solidity, Vyper, C++, Python та інші. У контракті визначаються умови, які повинні бути виконані для успішної роботи контракту.
- 2) Після написання смарт-контракту його потрібно розгорнути в блокчейн, де він буде доступний для використання. Контракт зберігається в блокчейні як транзакція.
- 3) Коли сторони укладають контракт, вони встановлюють умови, які потрібно виконати для його успішної роботи. Умови можуть бути різними, такі як виконання певних дій, передача певної суми грошей або активів.
- 4) Смарт-контракт автоматично перевірить задані умови та виконає свою роботу, якщо умови задовольняються.
- 5) Транзакції, пов'язані зі смарт-контрактами, записуються в блокчейн, що забезпечує прозорість та неможливість зміни інформації.
- 6) Після успішного виконання контракту обидві сторони отримують результат виконання договору. Наприклад, якщо договором передбачена передача певної суми грошей, обидві сторони отримують підтвердження про те, що гроші були передані.

## **1.1 Вигляд блоку в блокчейні**

Структура блоку в блокчейні може відрізнятися залежно від конкретної реалізації та типу блокчейну. Однак, загальна структура блоку включає в себе наступні компоненти:

1. Кожен блок містить посилання на хеш (унікальний ідентифікатор) попереднього блоку в ланцюжку. Це створює зв'язок між блоками і забезпечує послідовну структуру блокчейну.

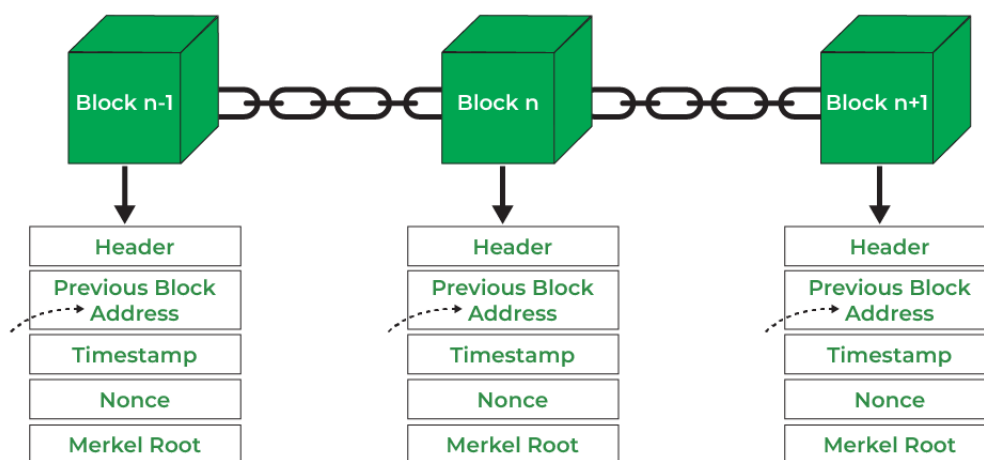
2. Кожен блок має власний хеш, який обчислюється на основі вмісту блоку. Хеш використовується для ідентифікації блоку та забезпечення його цілісності.

3. Складова блоку містить фактичну інформацію або транзакції, які включені в блок. Наприклад це можуть бути дані про перекази криптовалюти або взаємодії із смарт-контрактами.

4. Кожен блок має позначку часу, яка показує, коли блок був створений. Це дозволяє встановити хронологічний порядок блоків у ланцюжку.

5. Nonce - це випадкове число або значення, яке додається до блоку для отримання необхідного хешу. Використання nonce дозволяє регулювати складність обчислення хешу і забезпечує безпеку мережі.

Крім основних складових, блок може містити іншу інформацію, таку як номер блоку, дані про підписи або додаткові метадані, в залежності від конкретного використання блокчейну. Загалом, структура блоку включає хеш попереднього блоку, хеш поточного блоку, дані, відмітку часу та nonce. Див. [6]



## **1.2 Головні особливості блокчейн технологій**

Блокчейн є інноваційною технологією, яка має декілька головних особливостей:

- 1) Блокчейн працює в децентралізованому середовищі, де немає централізованого контролюючого органу. Натомість мережа складається з вузлів, які мають копії бази даних. Це дозволяє уникнути збоїв в одному вузлі і забезпечує стабільність і доступність мережі.
  - 2) Коли дані записані в блокчейн, вони стають практично незмінними. Зміна вже записаних блоків майже неможлива без співпраці більшості вузлів мережі. Це забезпечує цілісність і безпеку даних.
  - 3) Блокчейн забезпечує високий рівень прозорості, оскільки всі транзакції та записи є публічними і можуть бути переглянуті всіма учасниками мережі. Це сприяє довірі між сторонами та зменшує можливість фальсифікації або маніпуляцій з даними.
  - 4) Блокчейн використовує криптографічні методи для забезпечення безпеки транзакцій і даних. Кожна транзакція підписується цифровим підписом, щоб гарантувати її автентичність. Крім того, технологія блокчейн використовує хеш-функції та шифрування, щоб зробити дані стійкими до злому.
- Див. [3]

## Розділ 2

# Використання блокчейну

### 2.1 Застосування блокчейну у повсякденному житті

Блокчейн має потенціал надати людям низку переваг і змінити спосіб взаємодії та ведення бізнесу. Ось деякі з основних способів, як блокчейн може принести користь людям:

1) Технологія блокчейн, на якій базуються такі криптовалюти, як біткойн та ефіріум, відкриває можливості для безпечних і прямих фінансових транзакцій. Це означає, що люди можуть швидко переказувати кошти, оплачувати товари та послуги без необхідності залучати традиційні фінансові установи в якості посередників. Завдяки блокчейну можна забезпечити підвищену безпеку та захист даних від підробки. Інформація, що вноситься в блокчейн, захищається криптографічними методами і розподіляється між багатьма вузлами мережі. Це знижує ризик крадіжки, зміни або втрати даних, що особливо важливо для особистої інформації, медичних даних, фінансової звітності та інших конфіденційних даних.

2) Можливість автентифікації документів, продуктів та інших цифрових активів. Використовуючи унікальні хеші та криптографічні підписи,

ви можете перевірити походження та автентичність інформації. Це допомагає уникнути шахрайства, підробки та контрафакту. Блокчейн працює в децентралізованому середовищі, де немає центрального органу, який контролює всю систему. Це означає, що рішення та управління приймаються за спільною згодою учасників мережі, а не однією централізованою структурою. Децентралізована природа блокчейну забезпечує більшу прозорість, відкритість і відсутність єдиної точки вразливості в системі.

## Розділ 3

# Недоліки блокчейну та смарт-контрактів

### 3.1 Недоліки блокчейну

Хоча технологія блокчейн має потенціал для революції в багатьох галузях, вона також має свої недоліки та обмеження. Давайте розглянемо деякі з них.

Одним з головних недоліків блокчейну є його обмежена масштабованість, яка виникає через розподілену природу технології. Кожен вузол мережі зобов'язаний вести повний журнал всіх транзакцій, що призводить до значних обчислювальних і мережевих навантажень. Це обмежує швидкість обробки транзакцій і ускладнює розширення мережі, що робить її менш ефективною в масових додатках.

Ще одним недоліком блокчейну є його високе енергоспоживання. Процес майнінгу (видобування) криптовалют, зокрема біткоїна, вимагає значних обчислювальних ресурсів і споживає велику кількість електроенергії. Це негативно впливає на екологічну стійкість технології, оскільки призводить до збільшення викидів вуглекислого газу та споживання ресурсів. Велика

кількість обчислювальних потужностей, необхідних для підтримки мережі блокчейн, робить її менш екологічною і ставить питання про необхідність пошуку більш енергоефективних рішень.

Недоліком і перевагою є відкритість і прозорість транзакцій, коли всі дані доступні для перегляду всім учасникам мережі. Це може призвести до виникнення проблем з приватністю та конфіденційністю персональних даних. Хоча існують рішення, такі як конфіденційні блокчейни, які захищають певну інформацію за допомогою шифрування, вони все ще потребують подальших досліджень і вдосконалення. Важливо знайти баланс між прозорістю та конфіденційністю, щоб захистити персональні дані від несанкціонованого доступу та зловживань.

Серед викликів, що стоять на шляху впровадження технології блокчейн, є правові та регуляторні питання. Ці проблеми включають міжнародні обмеження, відсутність стандартів і нерозуміння правового статусу блокчейн транзакцій, що ускладнює використання технології в більшості країн. Через недостатній розвиток нормативно-правової бази у сфері блокчейну спостерігається нестабільність і невизначеність, що перешкоджає його широкому впровадженню та інтеграції в різні галузі. Для забезпечення сталого розвитку технології блокчейн необхідна подальша робота над правовими рішеннями та створенням чіткої нормативно-правової бази. Див. [3]

## **3.2 Недоліки смарт-контрактів**

Серед переваг смарт-контрактів, заснованих на технології блокчейн, є і недоліки, які необхідно враховувати:

Смарт-контракти, створені за допомогою програмного коду, вимагають особливої уваги, оскільки навіть найменші помилки можуть мати серйозні наслідки. Вразливості в коді можуть призвести до можливості крадіжки коштів або некоректного виконання транзакцій, що може вплинути на надій-

ність і безпеку смарт-контрактів. Тому важливо приділяти належну увагу проектуванню та аудиту смарт-контрактів, а також встановлювати механізми виявлення та виправлення помилок для забезпечення їх безперервної та безпечної роботи.

Недоліком смарт-контрактів також можна вважати їхню незмінність. Це означає, що в разі помилки в угоді, або непередбачуваних обставин може бути складно або навіть неможливо внести зміни або скасувати контракт без взаємодії сторін і внесення змін до блокчейну. Така ситуація може спричинити юридичні проблеми та призвести до конфліктів між сторонами, оскільки виникає необхідність вирішувати суперечки щодо виконання смарт-контрактів на основі даних, які вже записані в розподіленій системі.

Деякі смарт-контракти можуть потребувати зовнішніх джерел даних, таких як ціни на акції або погодні умови, для прийняття рішень або виконання умов контракту. Однак це несе ризик невірного або неправильного використання інформації з цих джерел, що може призвести до некоректного виконання контракту.

Оскільки смарт-контракти є відносно новим явищем, відсутність єдиних стандартів та чіткого правового регулювання ускладнює їх використання. Це може створити плутанину та правову невизначеність у сфері смарт-контрактів, а також ускладнити визнання їхньої юридичної сили та вирішення спорів, що виникають у зв'язку з їхнім використанням. Необхідна подальша робота над створенням відповідних стандартів і розробкою правової бази для ефективного впровадження смарт-контрактів. Див. [7]

## Розділ 4

# Програмна реалізація

У своїй дипломній роботі я розробив смарт-контракт для зберігання криптовалют та отримання пасивного доходу. Враховуючи низьку процентну ставку за зберігання фіатних валют в сучасних банках, я вирішив використати свої знання в розробці смарт-контрактів з використанням мови C++ для програмування на блокчейні EOS.

### 4.1 Створення конфігурації

Власник конфігурації встановлює параметри, включаючи вимогу до типу криптовалюти, яку користувачі повинні вносити на депозит, і визначає тривалість періоду, протягом якого ці умови будуть діяти.

Перше за все потрібно створити запис у таблиці з пустою конфігурацією. Для цього викликаємо функцію **"createpool"** у смарт-контракті.



Enter Data

pool\_owner

jurassicwaxx

SUBMIT TRANSACTION

Параметри досить прості, потрібно просто вписати власника конфігурації. Далі підписуємо транзакцію і результат можемо побачити у таблиці під назвою "**custompools**"

| pool_id | pool_owner   | reward_tokens_contract | reward                     | total_staked_tokens        | min_stake_amount           | staked_toke |
|---------|--------------|------------------------|----------------------------|----------------------------|----------------------------|-------------|
| 1       | jurassicwaxx | none                   | 0.0000000000000000<br>NONE | 0.0000000000000000<br>NONE | 0.0000000000000000<br>NONE | none        |

Після успішного створення таблиці тепер потрібно заповнити всі порожні поля, щоб програма працювала належним чином. Для цього давайте крок за кроком пройдемося по всіх параметрах, щоб зрозуміти, як їх правильно заповнювати.

- 1) **pool\_owner** - визначає власника конфігурації. Тільки власник має можливість змінювати свою конфігурацію.
- 2) **reward\_token\_contract** - смарт-контракт токену(криптовалюти), яку буде отримувати користувач як винагороду.
- 3) **reward** - Це кількість криптовалюти, яка буде розподілена між користувачами протягом періоду.
- 4) **total\_staked\_tokens** - загальна к-сть токенів(криптовалюти) яку внесли користувачі в дану конфігурацію.
- 5) **min\_stake\_amount** - мінімальна кількість токенів (криптовалюти), яку потрібно внести однією транзакцією користувачем.
- 6) **staked\_tokens\_contract** - смарт-контракт токену(криптовалюти) яку потрібно вносити в дану конфігурацію.
- 7) **start\_time** - час, коли конфігурація стає активною і користувачі можуть вносити свою криптовалюту.
- 8) **end\_time** - час, коли закінчується конфігурація. Після цього ніхто не може вносити криптовалюту, тільки забрати яку вони до того внесли.
- 9) **available** - це є bool змінна, яка вказує чи є дана конфігурація активною. Після того як власник завершує налаштування він активує конфігурацію.

Для встановлення значень "start\_time" та "end\_time" потрібно викликати функцію "poolsetup"



Enter Data

owner  
jurassicwaxx

pool\_id  
1

start\_time  
1685444340

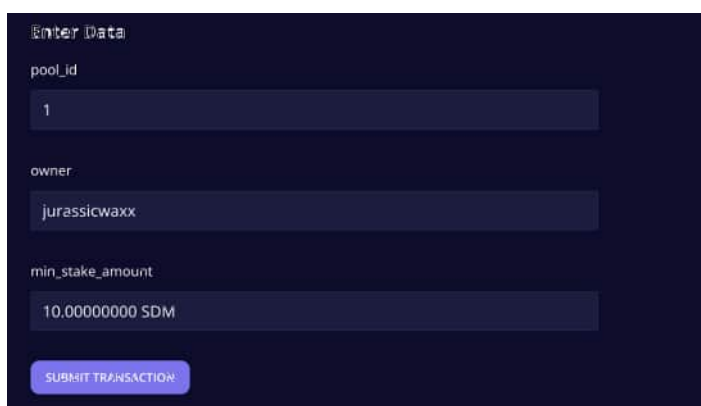
end\_time  
1685944340

SUBMIT TRANSACTION

Відповідно до вимог, час для параметрів слід вказувати як кількість секунд, що пройшли з початку епохи.

Далі власнику конфігурації потрібно ініціалізувати reward\_token\_contract. Для цього потрібно переказати будь-яку суму криптовалюти на наш смарт-контракт і в повідомленні вказати "set reward token:pool\_id". Після цього контракт автоматично змінить дані в таблиці та поверне власнику його надіслані кошти. Теж саме потрібно зробити із staked\_tokens\_contract, відповідно з іншим повідомлення до платежу "set staking token:pool\_id".

Після цього можна ініціалізувати min\_stake\_amount. Викликаємо функцію "setminstake"



Enter Data

pool\_id  
1

owner  
jurassicwaxx

min\_stake\_amount  
10,00000000 SDM

SUBMIT TRANSACTION



| pool_id | pool_owner  | reward_tokens_contract | reward                     | total_staked_tokens        | min_stake_amount           | staked_tok |
|---------|-------------|------------------------|----------------------------|----------------------------|----------------------------|------------|
| 20      | mr5xo.wam   | none                   | 0.0000000000000000<br>NONE | 0.0000000000000000<br>NONE | 0.0000000000000000<br>NONE | none       |
| 21      | gpi3..c.wam | waxpntgames            | 1892254.45457935<br>WNG    | 0.00000000 ELEMENT         | 10.00000000<br>ELEMENT     | wngtoken.g |
| 22      | gpi3..c.wam | t.taco                 | 130585.5605 SHING          | 53000.00000000 WNG         | 1000.00000000 WNG          | waxpntgam  |
| 23      | gpi3..c.wam | eosio.token            | 367.74499904 WAX           | 363132.00000000 WNG        | 1000.00000000 WNG          | waxpntgam  |
| 24      | gpi3..c.wam | waxpntgames            | 4904091.35235470<br>WNG    | 2501412.0000<br>WAXWNG     | 10.0000 WAXWNG             | swap.taco  |
| 25      | supercallou | none                   | 0.0000000000000000<br>NONE | 0.0000000000000000<br>NONE | 0.0000000000000000<br>NONE | none       |

Ось приклад різних конфігурацій. Для користувача є достатньо великий вибір, далі залишається тільки вибрати і внести свою криптовалюту.

### 4.3 Внесення криптовалюти користувачем

Тепер користувач приймає головне рішення - він обирає, де йому вигідніше внести рахунок криптовалютою. Після вибору користувачеві необхідно перевести криптовалюту на депозит. Для здійснення переказу користувач повинен вказати повідомлення **"stake:pool\_id"**, щоб смарт-контракт зрозумів, яку конфігурацію вибрав користувач. Після того, як користувач внесе криптовалюту, результати будуть записані в таблиці **"participants"** і **"userinfo"** відповідно.

| # | pool_id | participants   |
|---|---------|--|
| 1 | 22      | ["3o3to.wam", "rnexs.wam"]   |
| 2 | 23      | ["n.hl2.c.wam", "wngtoken.gm", "rnexs.wam", "sa3bu.wam", "vkuru.wam", "vvrva.wam"] |
| 3 | 24      | ["vkuru.wam", "rnexs.wam"]   |

| # | pool_id | time       | staked_tokens      | collected_rewards |
|---|---------|------------|--------------------|-------------------|
| 1 | 22      | 1685451603 | 50000.00000000 WNG | 1286.9138 SHING   |

## 4.4 Отримання винагород користувачем

Після успішного внесення криптовалюти, користувачу залишається чекати своєї виплати, яка відбувається раз в  $n$  годин (встановлюється адміном контракту). На попередньому знімку видно, що є **"collected\_rewards"** і воно є не порожнім, отже, користувач може забрати цю суму собі на гаманець, виконавши функцію **"collect"**:



The image shows a dark-themed web interface with a form titled "Enter Data". There are two input fields: the first is labeled "user" and contains the text "jurassicwaxx"; the second is labeled "pool\_id" and contains the number "1". Below the fields is a blue button with the text "SUBMIT TRANSACTION".

## 4.5 Розрахунок винагороди для користувачів

Для розрахункової формули я враховував такі фактори як:

- 1) **user\_staked\_token** - загальна к-сть криптовалюти яку вніс користувач
- 2) **total\_staked\_tokens** - загальна к-сть криптовалюти яку внесли всі користувачі.
- 3) **time\_now** - дата на момент розрахунку по формулі.
- 4) **last\_time\_claim** - дата від останнього розрахунку по формулі.
- 5) **total\_pool\_rewards** - загальна к-сть криптовалюти яку вніс власник конфігурації для винагород користувачів.
- 6) **end\_pool\_time** - дата закінчення конфігурації.
- 7) **start\_pool\_time** - дата початку конфігурації.

Використаймо ці всі дані у такій формулі для розрахунку:

$$reward\_per\_one\_second = \frac{total\_pool\_rewards}{end\_pool\_time - start\_pool\_time}$$

$$user\_staked\_time = (time\_now - last\_time\_claim)$$

$$user\_reward = \frac{user\_staked\_token}{total\_staked\_tokens} * \\ *reward\_per\_one\_second * user\_staked\_time$$

Детальніше ознайомитися із кодом можна за посиланням:

<https://github.com/kpm-lnu/student-applications/tree/develop/2022-2023/PMP-42/coursework/Yura%20Spasnyk>

# Висновок

Технологія блокчейн демонструє великий потенціал і перспективи в різних галузях, включаючи фінанси, логістику та управління ланцюгами поставок. Впровадження блокчейну може допомогти підвищити ефективність, прозорість і безпеку транзакцій та операцій. Однак при впровадженні блокчейну необхідно враховувати недоліки та виклики, такі як масштабованість, енергоспоживання, конфіденційність та правове регулювання.

Для успішного впровадження блокчейну необхідно продовжувати дослідження та розробку стандартів, які забезпечать правовий статус смарт-контрактів та сумісність між різними блокчейнами. Крім того, важливо знайти шляхи покращення масштабованості та енергоефективності блокчейну для забезпечення його широкого використання.

Загалом, технологія блокчейн має потенціал стати основою для інноваційних рішень і перетворень у різних галузях. Її успіх залежатиме від постійної роботи над вдосконаленням та вирішенням проблем, а також від усвідомлення та прийняття її потенціалу в суспільстві.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Когут Ю., Технології блокчейн та криптовалюта: ризики та кібербезпека / Ю. Когут // - 2022.
- [2] Кравченко П., Блокчейн і Децентралізовані Системи / П.Кравченко, Б.Скрябін, О.Дубініна, О.Курбатов // - Харків, - 2019.
- [3] Andreas M. A., Mastering Bitcoin: Programming the Open Blockchain / M. A. Andreas // - London, - 2017.
- [4] Lantz L., Mastering Blockchain. 1st Ed. / L. Lantz, D. Cawrey // - 2017.
- [5] Raval S., Decentralized Applications: Harnessing Bitcoin's Blockchain technology / S. Raval // - London, - 2016.
- [6] Tapscott D., Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World / D Tapscott, A. Tapscott // - 2018.
- [7] Wei-Meng L., Beginning Ethereum Smart Contracts Programming / L. Wei-Meng // - 2019.