

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І
АРХІТЕКТУРИ**

Факультет автоматизації і інформаційних технологій

Кафедра управління проектами

ЗАТВЕРДЖУЮ

Завідувач кафедри

Сергій БУШУЄВ

“ ___ ” _____ 2023 року

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО АТЕСТАЦІЙНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА**

**Розробка та оптимізація автоматизованого торгового бота для
криптовалютного ринку**

Виконав студент групи: ІСТ-ШКТ-11м

Янушкевич Карина Євгенівна

(прізвище, ім'я та по батькові повністю)

Спеціальність: 126 “Інформаційні системи та технології”

Освітня програма: Штучний інтелект. Когнітивні технології

Керівник: Бушуєва Н. С.

(прізвище, ініціали,)

Доктор технічних наук, професор

науковий ступінь, вчене звання

Рецензент: _____

(прізвище, ініціали,)

_____ *науковий ступінь, вчене звання*

Київ 2023р

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І
АРХІТЕКТУРИ**

Факультет: Автоматизації і інформаційних технологій
Кафедра: Управління проектами
Освітній рівень: Магістр за освітньо-професійною програмою
Галузь знань: 12 Інформаційні технології
Спеціальність: 126 "Інформаційні системи та технології"
Освітня програма: Штучний інтелект. Когнітивні технології

ЗАТВЕРДЖУЮ
Завідувач кафедри
Сергій БУШУЄВ
„___” _____ 2023 року

**З А В Д А Н Н Я
ДО ВИКОНАННЯ АТЕСТАЦІЙНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА**

Янушкевич Карина Євгенівна

(прізвище, ім'я та по батькові студента)

1. Тема роботи:

«Розробка та оптимізація автоматизованого торгового бота для криптовалютного ринку»

затверджена наказом ректора КНУБА № 2385/2 від 11 жовтня 2023 року

2. Керівник роботи:

Д.т.н., проф. Бушуєва Наталія Сергіївна

(прізвище, ім'я та по батькові, науковий ступінь, вчене звання)

3. Строк подання студентом роботи до захисту:

За 5 днів до захисту

4. Зміст пояснювальної записки (перелік питань, які слід розробити):

Провести аналіз автоматизованих торгових систем на криптовалютному ринку, та

технічних індикаторів, для використання у торговому боті.

Розробити організаційну структуру.

Розробити торгову стратегію у TradingView за допомогою мови програмування Pine Script, провести зворотне тестування стратегії.

Провести аналіз результатів роботи автоматизованого торгового бота.

5. Графічний матеріал за розділами:

Рисунки технічних індикаторів. Організаційна структура.

Візуалізація платформ. Код торгової стратегії.

Результат зворотного тестування.

Результат роботи автоматизованого торгового бота.

6. Календарний план виконання роботи:

Види робіт та їх зміст	Дата виконання
Збір матеріалів обраного напрямку роботи	11.10.2023
Опрацювання та аналіз матеріалів роботи	15.10.2023
Вступ	20.10.2023
Розділ 1.	31.10.2023
Розділ 2.	10.11.2023
Розділ 3.	16.11.2023
Висновки	23.11.2023
Остаточне оформлення роботи	30.11.2023
Перевірка роботи на плагіат	01.12.2023
Попередній захист роботи на кафедрі	01.12.2023
Направлення роботи на рецензування	04.12.2023

7. Консультанти розділів атестаційної випускної роботи

Розділ	Прізвище, ініціали та посада консультанта	Перевірів	
		дата	підпис
Розділ 1.	---	---	---
Розділ 2.	----	---	---
Розділ 3.	-----	---	----

8. Дата видачі завдання _____

Зав. кафедри _____

(підпис)

Керівник _____

(підпис)

Студент _____

(підпис)

Сергій БУШУЄВ _____

(прізвище та ініціали)

Бушуєва Н.С. _____

(прізвище та ініціали)

Янушкевич К.Є. _____

(прізвище та ініціали)

РЕЗЮМЕ (summary) <i>до атестаційної випускної роботи студента:</i>		Янушкевич Карина Євнегівна	
<i>ЗВО</i>	Київський національний університет будівництва і архітектури		
<i>Тема</i>	Розробка та оптимізація автоматизованого торгового бота для криптовалютного ринку		
<i>Освітній ступінь</i>	Магістр за освітньо-професійною програмою навчання		
<i>Факультет</i>	Автоматизації і інформаційних технологій		
<i>Кафедра</i>	Управління проектами		
<i>Спеціальність</i>	126 «Інформаційні системи та технології»		
<i>Освітня програма</i>	Штучний інтелект. Когнітивні технології		
<i>Керівник</i>	Бушуєва Наталія Сергіївна		
<i>Обсяг роботи:</i>	<i>пояснювальна записка, стор.</i>	<i>розділів</i>	<i>слайдів презентації</i>
	73	3	14
<i>Розділ 1.</i>	<p>Проведено аналіз основних характеристик та функцій криптовалютних бірж, включаючи роль у забезпеченні ліквідності, а також вибір централізованої біржі Binance. Також розглянуті основи автоматизованої торгівлі на криптовалютному ринку, зокрема переваги, закони та правила в різних країнах. Наведено ключові принципи автоматизованої торгівлі, включаючи управління ризиками, вибір стратегій, використання API та зворотне тестування. Проведено аналіз прикладів мов програмування, таких як Pine Script, і типів технічних індикаторів для автоматизованих торгових стратегій.</p>		
<i>Розділ 2.</i>	<p>Було створено організаційну структуру проекту та розроблено план для ефективного створення торгового бота. Використання платформи TradingView для розробки стратегій, зокрема використання індикатора RSI. Розробка торгової стратегії з використанням мови програмування Pine Script. Проведено зворотне тестування стратегії, оцінено її ефективність та підібрано оптимальні налаштування..</p>		

<i>Розділ 3.</i>	Використання системи сповіщень TradingView для отримання повідомлень за умов, відповідних стратегії. Створення API ключа на Binance для взаємодії торгової системи з біржею. Підключено сигнал та стоп-лос у Finandy. Проведено аналіз отриманих результатів.
<i>Висновки по роботі:</i>	У роботі показан розвиток криптовалютного ринку та попит на автоматизовані торгові системи. Розроблено бота з успішною стратегією, підключено до Binance через Finandy для автоматизованих угод та вирішено проблему зі швидким виставленням стоп-лосу. Стратегія оптимізована для роботи з різними криптовалютними парами, що робить її універсальною. Проведений аналіз вказує на потребу в подальших дослідженнях для підвищення ефективності автоматизації торгових систем на криптовалютному ринку.
<p>Ключові слова: криптовалюти біржі, автоматизована торгівля, план проекту, Pine Script, управління ризиком, торговий бот.</p> <p>Keywords: cryptocurrency exchanges, automated trading, project plan, Pine Script, risk management, trading bot.</p>	

Укладач: _____

Керівник: _____

« ____ » _____ 2023 р.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1: ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЗОВАНОЇ ТОРГІВЛІ НА КРИПТОВАЛЮТНИХ РИНКАХ	9
1.1 ПОНЯТТЯ БІРЖИ НА КРИПТОВАЛЮТНИХ РИНКАХ	9
1.2 ВВЕДЕННЯ В АВТОМАТИЗОВАНУ ТОРГІВЛЮ.....	14
1.3 ОСНОВНІ ПРИНЦИПИ АВТОМАТИЗОВАНОЇ ТОРГІВЛІ	20
1.4 ТЕХНІЧНІ АСПЕКТИ АВТОМАТИЗОВАНОЇ ТОРГІВЛІ НА КРИПТОВАЛЮТНОМУ РИНКУ	26
1.5 ТИПИ ТЕХНІЧНИХ ІНДИКАТОРІВ В АВТОМАТИЗОВАНИХ ТОРГОВИХ СИСТЕМАХ	29
ВИСНОВКИ 1 РОЗДІЛУ	34
РОЗДІЛ 2: РОЗРОБКА ТА НАЛАШТУВАННЯ ТОРГОВОГО БОТА	35
2.1 ОРГАНІЗАЦІЙНА СТРУКТУРА ПРОЕКТУ	35
2.2 ВИКОРИСТАННЯ ПЛАТФОРМИ TRADINGVIEW	39
2.3 ОПИС ТЕХНІЧНИХ ІНДИКАТОРІВ У СТРАТЕГІЇ RSI	41
2.4 РОЗРОБКА СТРАТЕГІЇ НА PINE SCRIPT.....	43
2.5 ЗВОРОТНЕ ТЕСТУВАННЯ СТРАТЕГІЇ (BACKTEST)	56
ВИСНОВКИ 2 РОЗДІЛУ	61
РОЗДІЛ 3: ПІДКЛЮЧЕННЯ ТОРГОВОЇ СИСТЕМИ ДО БІРЖИ ТА УКЛАДАННЯ УГОД.....	61
3.1 СИСТЕМА СПОВІЩЕНЬ TRADINGVIEW	61
3.2 СТВОРЕННЯ VINANCE API.....	62
3.3 ПІДКЛЮЧЕННЯ СИГНАЛУ У FINANDY	65
3.4 ПІДКЛЮЧЕННЯ СТОП-ЛОСУ У FINANDY	68
3.5 РЕЗУЛЬТАТ УКЛАДАННЯ УГОД.....	70
ВИСНОВКИ 3 РОЗДІЛУ	74
ВИСНОВОК.....	75
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	76
ДОДАТКИ.....	78

ВСТУП

У сучасному світі, де цифрові технології стрімко розвиваються, фінансовий ринок не залишається осторонь цих змін, з'явилося безліч нових можливостей та інструментів торгівлі. Особливо це стосується криптовалютного ринку, який постійно пропонує нові можливості для трейдерів та розробників. На даний момент доступно багато варіацій автоматизованих торгових систем, написаних на різних мовах програмування та з використанням великої кількості інструментів, більшість з яких є закритими, та доступними лише невеликій кількості людей, які ці системи і розробляють. У вільному доступі все ж можливо знайти торгові системи, але без розуміння написання коду, та самої роботи системи можливо понести великі ризики з боку фінансів та безпеки даних. У цьому контексті, розробка власної автоматизованої торгової системи для криптовалютного ринку стає актуальною та перспективною темою.

Головною проблемою, яку повинна вирішити ця робота, є розробка та оптимізація автоматизованого торгового бота, спрямованого на криптовалютний ринок. Це включає в себе аналіз ринкових даних, виконання торгових операцій на основі обраної стратегії та використання спеціалізованих інструментів технічного аналізу.

Основною ціллю цього дослідження є створення ефективної, автоматизованої системи торгівлі, здатної самостійно аналізувати ринок та ухвалювати рішення про купівлю чи продаж криптовалюти без втручання з боку людини.

Для досягнення цієї мети були поставлені наступні завдання:

1. Розробка торгової стратегії, заснованої на RSI з додаванням SI та VO.
2. Використання графічної платформи TradingView та її вбудованого редактора Pine для написання та тестування стратегії.
3. Реалізація механізмів управління ризиками.
4. Бек-тестування розробленої стратегії.

5. Інтеграція торгової стратегії з API біржі Binance і фінансовим сервісом Finandy.
6. Успішне укладання угоди без втручання людини з моменту знаходження точки для входу, до закриття позиції та фіксування результату .

Актуальність цієї теми впливає зі стрімкого розвитку криптовалютного ринку та появи нового напрямку у біржовій торгівлі, а саме торгівлі із застосуванням автоматичних торгових систем, та великим попитом на них. Цей ринок характеризується високою волатильністю та швидкими змінами, що вимагає від трейдерів швидкого реагування та ефективного аналізу даних. Автоматизовані торгові боти можуть значно підвищити ефективність торгових операцій, мінімізувати помилки та забезпечити більш раціональне управління ризиками.

Для розробки торгового бота будуть використані сучасні програмні інструменти, зокрема, TradingView для візуалізації даних та тестування стратегій, а також інші необхідні інструменти для реалізації проекту включно з мовою програмування Pine script.

Ця робота має на меті не тільки розробити та оптимізувати торговий бот, а й дослідити потенціал автоматизації у криптовалютній торгівлі, і чи важливо шукати нові методи для розробки автоматизованих торгових систем.

РОЗДІЛ 1: ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЗОВАНОЇ ТОРГІВЛІ НА КРИПТОВАЛЮТНИХ РИНКАХ

1.1 ПОНЯТТЯ БІРЖІ НА КРИПТОВАЛЮТНИХ РИНКАХ

Криптовалютні біржі є критично важливою інфраструктурною основою ринків цифрових активів, забезпечуючи необхідну базу для здійснення транзакцій в екосистемі розподіленого реєстру. Ці платформи функціонують завдяки впровадженню механіки книги замовлень і алгоритмів узгодження, які полегшують обмін криптовалютами, будь то за допомогою транзакцій фіат-криптовалюта або криптовалюта-криптовалюта. Їхні операції підкреслюються складною взаємодією криптографічних протоколів безпеки, мережових затримок і механізмів консенсусу, які покликані забезпечити цілісність і безперервність торгівлі.

З технічної точки зору, ці біржі спроектовані таким чином, щоб забезпечити високу пропускну здатність і низьку затримку торгових операцій, при цьому особлива увага приділяється масштабованості, щоб впоратися зі зростаючим обсягом транзакцій. Надійні інтерфейси прикладного програмування (API) лежать в основі операційної ефективності цих платформ, забезпечуючи автоматизовану торгівлю, аналіз ринку та інтеграцію із зовнішніми сервісами.

Технічна інфраструктура криптовалютних бірж приділяє особливу увагу безпеці та продуктивності. Заходи безпеки різноманітні і включають протоколи шифрування, двофакторну автентифікацію та регулярний аудит. Продуктивність оцінюється за здатністю біржі виконувати замовлення з мінімальною затримкою, підтримувати безперебійну роботу і динаміку масштабуватися у відповідь на вимоги ринку.

У цій сфері безпеку неможливо переоцінити. Криптовалютні біржі застосовують багатогранний підхід до захисту активів, включаючи холодильні камери, гаманці з декількома підписами і складні системи спостереження, щоб зменшити ризик несанкціонованого доступу і компрометації активів. З посиленням регуляторного контролю комплаєнс став невід'ємною частиною дизайну бірж,

включаючи процеси “Знай свого клієнта” (KYC) і боротьби з відмиванням грошей (AML) для узгодження з глобальним фінансовим управлінням.

Щоб задовольнити потреби як початківців, так і досвідчених трейдерів, біржі включають в себе ряд торгових функцій, таких як спотова торгівля, маржинальна торгівля, ф'ючерси та безстрокові свопи. Користувацький досвід покращується завдяки інтуїтивно зрозумілим інтерфейсам, інформаційним панелям, що налаштовуються, та системам підтримки клієнтів, що швидко реагують на запити.

На ринку цифрових активів існує безліч моделей бірж, кожна з яких має свої технічні характеристики.

Централізовані біржі (CEX) характеризуються ієрархічною архітектурою, де центральна організація контролює операції. Вони розроблені для високої продуктивності, вміщують широкий спектр торговельної діяльності та пропонують підвищену ліквідність завдяки глибині портфеля замовлень і стратегіям маркетингу.

Прикладом централізованої криптовалютною біржі є Binance. Binance одна з найбільших і найвідоміших криптовалютних бірж у світі, що пропонує широкий спектр послуг, включаючи спотову торгівлю, торгівлю ф'ючерсами та опціонами, власний блокчейн (Binance Smart Chain) та екосистему, яка підтримує широкий спектр криптовалют і цифрових активів. Як централізована платформа, вона управляє торгівлею користувачів, зберігає їхні кошти та вимагає від користувачів пройти різні перевірки безпеки.

Децентралізовані біржі (DEX) працюють на основі розподіленого реєстру, укладаючи угоди безпосередньо між учасниками без посередників. Вони використовують смарт-контракти для забезпечення дотримання правил торгівлі та виконання транзакцій, що відображає прихильність до децентралізованого духу технології блокчейн.

Прикладом децентралізованої біржі (DEX) є Uniswap. Uniswap одна з найвідоміших DEX, що працює на блокчейні Ethereum і використовує систему автоматизованого маркет-мейкера (АММ) замість традиційної книги замовлень. У цій системі пули ліквідності використовуються для полегшення торгівлі, а ціни визначаються співвідношенням двох токенів у кожному пулі. Користувачі Uniswap торгують безпосередньо з власних гаманців без необхідності довіряти свої кошти посереднику



Рис. 1.1. Робота централізованих та децентралізованих бірж

Гібридні біржі прагнуть поєднати високу пропускну здатність і простоту використання, характерні для CEX, з моделлю DEX, що не передбачає довіри і орієнтована на безпеку. Ці системи часто будуються з багаторівневою архітектурою, забезпечуючи безперебійну торгівлю і намагаючись мінімізувати довіру до будь-якої окремої сторони.

Прикладом гібридної біржі є Qurrex. Qurrex інтегрує інфраструктуру традиційної централізованої біржі з мережевими технологіями, притаманними децентралізованим біржам. Такий підхід має на меті об'єднати надійність і продуктивність централізованих сервісів з підвищеною безпекою і прозорістю децентралізованих архітектур.

Гібридні біржі, такі як Qurrex, пропонують користувачам швидкість і простоту використання централізованих платформ, включаючи передові торгові інструменти і підтримку клієнтів, а також забезпечують переваги безпеки децентралізованих бірж, такі як зниження ризиків простою сервера і хакерських атак. Ці платформи призначені для того, щоб дати трейдерам контроль над своїми коштами, часто використовуючи гаманці з декількома підписами або подібні механізми, які гарантують, що активи трейдерів не зберігаються централізовано, забезпечуючи при цьому роботу централізованої системи узгодження ордерів.

Криптовалютні біржі є фундаментальною основою, на якій функціонує ринок цифрових активів, втілюючи в собі складне поєднання фінансових послуг і передових технологій. Ці платформи роблять більше, ніж просто сприяють обміну цифровими валютами; вони реалізують складні алгоритми для узгодження замовлень, впроваджують криптографічні заходи безпеки для захисту активів і забезпечують безперервну роботу глобального ринку в режимі реального часу. Як двигуни ліквідності та вектори інновацій, ці біржі підтримують цілісність ринку завдяки високочастотному виконанню торгів, стійкості до кіберзагроз та дотриманню регуляторного ландшафту, що постійно змінюється. Надійність технологічної інфраструктури біржі - від її здатності масштабуватися у відповідь на попит користувачів до здатності інтегрувати нові функції блокчейну - матиме вирішальне значення для підтримання траєкторії зростання криптовалютною економіки. Завдяки своїй технічній досконалості та адаптивності, криптовалютні біржі здатні орієнтуватися в складній взаємодії децентралізації, безпеки та дотримання нормативних вимог, що посилює їхню ключову роль у становленні торгівлі цифровими активами.

Обираючи криптовалютну біржу для автоматизованої торгівлі, важливо враховувати її основні характеристики, такі як обсяг ліквідності, вартість комісій, доступність API та інші фактори.

Ліквідність - у контексті криптовалютних ринків означає легкість, з якою цифровий актив можна купити або продати за стабільною ціною.

Високоліквідний ринок - це ринок, на якому активно торгують багато покупців і продавців, а актив можна швидко продати, не спричиняючи значних коливань ціни. Висока ліквідність має вирішальне значення для трейдерів, оскільки вона сприяє кращому визначенню ціни і дозволяє здійснювати операції у великих обсягах без істотного впливу на ринкову ціну.

API розшифровується як Application Programming Interface (інтерфейс прикладного програмування). Це набір протоколів, процедур та інструментів для створення програмних додатків, які визначають, як повинні взаємодіяти різні програмні компоненти. У випадку з криптовалютними біржами API використовуються для різних цілей.

API дозволяє підключати власне торгове програмне забезпечення або ботів до біржі, що дозволяє їм автоматизувати свої торгові стратегії шляхом розміщення замовлень, перевірки балансу та програмного управління торгівлею. За допомогою API біржі надають ринкові дані в режимі реального часу, історичні ціни, історію транзакцій та іншу важливу інформацію, яку трейдери та інші учасники ринку можуть використовувати для аналізу або для інтеграції зі своїми інструментами та додатками.

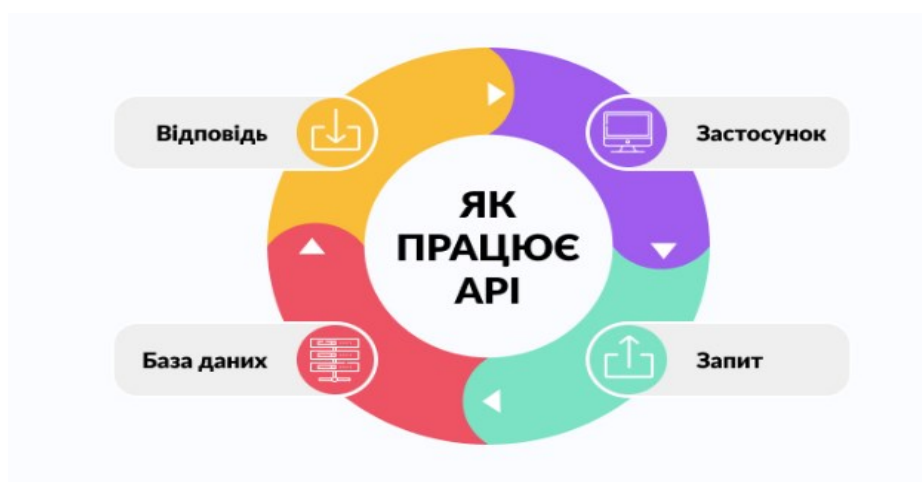


Рис. 1.2. Схема роботи API

Враховуючи всі пункти які потрібні для автоматизованої торгівлі було обрано централізовану криптовалютну біржу Binance на якій і будуть реалізовуватися угоди за допомогою торгівельного бота.

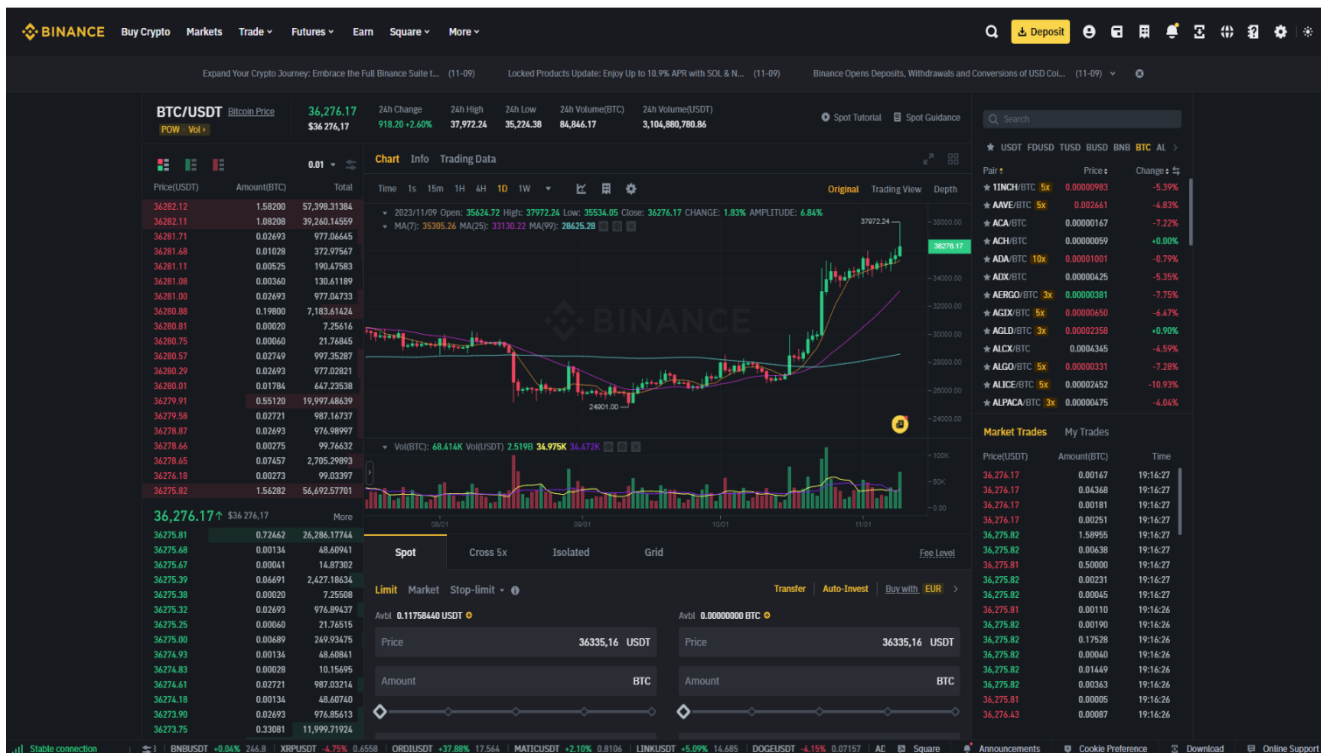


Рис. 1.3. Інтерфейс криптовалютної біржі Binance

1.2 ВВЕДЕННЯ В АВТОМАТИЗОВАНУ ТОРГІВЛЮ

У світі фінансів, що постійно розвивається, поява автоматизованих торгових систем стала значним поворотним моментом. Концепція автоматизованого трейдингу не була розроблена однією людиною, а скоріше є результатом безперервного технологічного прогресу і потреби в більш ефективній і менш емоційній торгівлі.

Шлях до автоматизованої торгівлі розпочався наприкінці 20-го століття, коли комп'ютери ставали все більш потужними і доступними. Трейдери і програмісти визнали потенціал використання цих машин для обробки величезних обсягів фінансових даних і точного виконання угод.

Початковою мотивацією автоматизованої торгівлі було усунення людських упереджень та емоцій з торгового процесу. Такі емоції, як страх і жадібність, часто призводили до ірраціональних торгових рішень, що призводило до значних втрат для багатьох інвесторів. Автоматизовані торгові системи мали на меті вирішити цю проблему, слідуючи заздалегідь визначеним алгоритмам і правилам, позбавленим людських почуттів.

З розвитком технологій зростали і можливості автоматизованих торгових систем. Вони еволюціонували від базових програм, які виконували заздалегідь визначені стратегії, до складних алгоритмів, здатних адаптуватися до мінливих ринкових умов. Машинне навчання та штучний інтелект були використані для створення торгових систем, які могли навчатися та вдосконалюватися з часом.

Автоматизована торгівля наразі стає все більш популярною на різних фінансових ринках, включаючи акції, сировинні товари, форекс і криптовалюти. Вона використовується індивідуальними трейдерами, інституційними інвесторами та фінансовими компаніями для управління портфелями, здійснення високочастотної торгівлі та реалізації складних торгових стратегій. Однак важливо зазначити, що автоматизована торгівля несе в собі власний набір ризиків і вимагає ретельної розробки, моніторингу та обслуговування для забезпечення її ефективності та прибутковості.

Використання автоматизованих торговельних систем поширилося на різні фінансові ринки, від акцій та облігацій до товарів та іноземної валюти. Трейдери та установи визнали переваги швидкості та точності, які пропонують ці системи, що призвело до їх широкого впровадження.

Однак розвиток автоматизованої торгівлі не обійшовся без проблем. З'явилися занепокоєння щодо волатильності ринку, раптових збоїв та алгоритмічних маніпуляцій, що спонукало регуляторні органи встановити правила та запобіжники для підтримки цілісності ринку.

Незважаючи на дебати та регуляторні перешкоди, автоматизована торгівля продовжує існувати і розвиватися. Вона стала фундаментальною складовою світових фінансових ринків, причому значна частина торгових операцій виконується машинами. Трейдери використовують цю технологію для укладання угод, точного управління портфелями та зменшення витрат.

Історія автоматизованого трейдингу відображає невпинний розвиток технологій та їхній вплив на світ фінансів. Вони назавжди змінили спосіб функціонування ринків, пропонуючи як можливості, так і виклики для інших учасників ринку.

Автоматизована торгівля криптовалютами, як і алгоритмічна торгівля на традиційних фінансових ринках, виникла як природний розвиток попередніх методів, які використовували трейдери. Історію автоматизованої торгівлі можна простежити з початку 1970-х років, коли на традиційних фінансових ринках з'явилися комп'ютеризовані торгові системи. Однак лише на початку 2000-х років, з появою електронних торгових платформ, алгоритмічна торгівля почала стрімко розвиватися.

Автоматизована криптовалютна торгівля, яку іноді називають автоматизованою торгівлею криптовалютою, - це практика використання комп'ютерних програм (криптовалютних ботів) для купівлі та продажу цифрових валют від чийогось імені. Ці програми призначені для реагування на ринкові зміни, щоб здійснювати торгівлю в оптимальний момент. Крім того, автоматична торгівля криптовалютами усуває елемент невизначеності та емоцій при купівлі та продажу криптовалют вручну.

Трейдери і розробники створюють конкретні торгові правила і стратегії, які програмуються в автоматизовану торговельну систему. Ці правила можуть базуватися на технічних індикаторах, фундаментальному аналізі, статистичних моделях або комбінації факторів. Тому однією з головних переваг автоматизованої торгівлі є швидкість і точність виставляти запити чи ордера за секунди або долі

секунди виходячи з торгової стратегії. Для самої людини такий процес займе набагато більше часу за який результат входу в позицію стає гіршим.

Хоча деякі новітні крипто-боти використовують смарт-контракти і працюють безпосередньо на блокчейні, більшість автоматизованих крипто-трейдингових платформ все ще використовують API.

Автоматизована торгівля криптовалютами, як правило, є законною в юрисдикціях, де торгівля криптовалютами дозволена. Однак регуляторне середовище є складним і сильно відрізняється в різних країнах. Наприклад:

Таблиця

1.1 Законодавство країн стосовно криптовалюної торгівлі

Країна	Відповідне законодавство або нормативно-правова база
Сполучені Штати	- Повинні відповідати правилам CFTC щодо торгівлі товарами.
	- Дотримуватися правил SEC, якщо токени вважаються цінними паперами.
	- Дотримуватися правил AML і KYC.
	- Також можуть застосовуватися закони окремих штатів.
Європейський Союз	- Повинна відповідати вимогам Директиви про ринки фінансових інструментів (MiFID II).
	- Повинні відповідати AMLD5 (5-й Директиві по боротьбі з відмиванням грошей).

	<ul style="list-style-type: none"> - Відповідність GDPR є обов'язковою для систем, що обробляють персональні дані.
Китай	<ul style="list-style-type: none"> - Торгівля криптовалютою та пов'язані з нею види діяльності здебільшого заборонені.
	<ul style="list-style-type: none"> - Фінансовим установам і платіжним компаніям заборонено надавати послуги, пов'язані з криптовалютою.
	<ul style="list-style-type: none"> - Просування, випуск і торгівля криптовалютами суворо регулюються.
Україна	<ul style="list-style-type: none"> - Криптовалюта є законною, нею можна торгувати та майнити.
	<ul style="list-style-type: none"> - Міністерство цифрової трансформації бере участь у створенні правового середовища для розвитку віртуальних активів і бізнесу в цій галузі.
	<ul style="list-style-type: none"> - Існують закони щодо протидії відмиванню грошей та фінансуванню тероризму, які також поширюються на криптоактиви поширюються і на криптоактиви (відповідно до рекомендацій FATF).
	<ul style="list-style-type: none"> - У 2021 році Україна прийняла закон, який легалізує та регулює криптовалюту.

Автоматизована торгівля в Україні, як і в багатьох інших країнах, набирає обертів завдяки підвищенню доступності технологій та зростаючому інтересу до ринку криптовалют. З огляду на позицію країни щодо цифрових інновацій, особливо в технологічному секторі, майбутнє автоматизованого трейдингу в Україні виглядає перспективною діяльністю.

Існує кілька типів ботів для торгівлі криптовалютами, кожен з яких відрізняється за характеристиками і функціоналом. Найпопулярнішими з них є арбітражні або сіткові торгові боти. Арбітражні боти використовують різницю в цінах на різних біржах, тоді як сіткова торгівля орієнтована на стратегію “купуй дешево, продавай дорого”.

Деякі автоматизовані криптоплатформи мають інші характеристики, наприклад, функція hodl. Це не просто торгівля, вона також дозволяє користувачам автоматично купувати і утримувати криптовалюту, купуючи її за низькими цінами. Користувач сам обирає криптовалюти, які він хоче, а бот допомагає йому в цьому.

Незалежно від місцезнаходження, трейдерам вкрай важливо забезпечити дотримання місцевих законів і нормативних актів, в тому числі тих, що стосуються фінансових ринків.

Виходячи з цього можна виділити наступні переваги та недоліки автоматизованої торгівлі криптовалютою. Важливо враховувати не тільки переваги а й недоліки цієї системи.

Таблиця

1.2 Переваги та недоліки автоматизованої криптовалютної торгівлі

Переваги	Недоліки
Ефективність і швидкість	Технічні збої
Беземоційне прийняття рішень	Відсутність інтуїції
Можливість зворотного тестування	Надмірна оптимізація

Операції на ринку 24/7	Вимоги до моніторингу
Послідовність	Регуляторні ризики
Диверсифікація	Обмеження стратегії
Зниження транзакційних витрат	Ризики безпеки
Масштабованість	Комплексне налаштування
Управління ризиками	Ринковий ризик

1.3 ОСНОВНІ ПРИНЦИПИ АВТОМАТИЗОВАНОЇ ТОРГІВЛІ

Алгоритмічна торгівля, часто відома як алго-трейдинг, використовує комп'ютерні програми, які слідують визначеному набору інструкцій (алгоритму) для укладання угод з метою отримання прибутку з такою швидкістю і частотою, яка неможлива для трейдера-людини. Визначені набори правил базуються на термінах, ціні, кількості або будь-якій математичній моделі. Крім можливостей отримання прибутку для трейдера, алгоритмічна торгівля робить ринки більш ліквідними, а торгівлю більш систематичною, виключаючи вплив людських емоцій на торговельну діяльність.

Алго-трейдинг використовується в багатьох формах торгівлі та інвестиційної діяльності в тому числі:

Середньо та довгострокові інвестори або фірми-покупці - пенсійні фонди, пайові інвестиційні фонди, страхові компанії - використовують алгоритмічну торгівлю для купівлі акцій у великих кількостях, коли вони не хочуть впливати на ціни акцій за допомогою дискретних інвестицій у великих обсягах.

Короткострокові трейдери та учасники на стороні продавця - маркет-мейкери (наприклад, брокерські контори), спекулянти та арбітражники - отримують вигоду від автоматизованого виконання угод; крім того, алго-трейдинг допомагає створити достатню ліквідність для продавців на ринку.

Систематичні трейдери - послідовники трендів, хедж-фонди або парні трейдери (нейтральна до ринку торгова стратегія, яка поєднує довгу позицію з

короткою в парі висококорельованих інструментів, таких як дві акції, біржові фонди (ETF), валюти, товари або опціони) - вважають, що набагато ефективніше запрограмувати свої торгові правила і дозволити програмі торгувати автоматично.

Алгоритмічна торгівля допомагає оптимізувати виконання угод, використовуючи складні математичні моделі для визначення оптимального часу, ціни та обсягу розміщення ордерів. Алгоритми призначені для розбиття великих ордерів на частини і виконання їх протягом певного часу, зменшуючи вплив на ринок і ризик руху ціни проти трейдера. Цей метод, відомий як середньозважена ціна за обсягом (VWAP), забезпечує виконання угод за ціною, близькою до середньої ціни за день, мінімізуючи таким чином прослизання ринку.

Використання алгоритмічної торгівлі значно підвищило ліквідність ринку. Швидко обробляючи інформацію та виконуючи замовлення, алгоритми зменшують спреди між попитом і пропозицією - різницю між найвищою ціною, яку покупець готовий заплатити за актив, і найнижчою ціною, яку продавець готовий прийняти. Покращена ліквідність сприяє підвищенню ефективності ринку, оскільки учасники можуть відкривати та закривати позиції з мінімальними транзакційними витратами.

Алгоритми мінімізують витрати, пов'язані з торгівлею, завдяки ефективному розміщенню замовлень. Вони обходять необхідність людського втручання, таким чином усуваючи накладні витрати на робочу силу і пов'язані з цим витрати. Таке зниження транзакційних витрат особливо вигідне для стратегій високочастотної торгівлі (HFT), де трейдери прагнуть заробити на дуже малих розбіжностях в цінах при великих обсягах.

Алгоритмічна торгівля покращує управління ризиками завдяки моніторингу позицій у режимі реального часу та автоматичному розгортанню стратегій хеджування. Алгоритми можуть швидко ідентифікувати несприятливі рухи на ринку і реагувати на них шляхом коригування або виходу з позицій на основі попередньо визначених параметрів ризику. Ця можливість негайного реагування є життєво важливою для управління швидкою динамікою фінансових ринків.

Суворо дотримуючись заздалегідь встановлених правил торгівлі, алгоритми усувають психологічні упередження, які часто впливають на людей. Такий системний підхід забезпечує дисципліну, зберігаючи цілісність торгової стратегії навіть у нестабільних ринкових умовах. Він також дозволяє проводити бек-тестування, коли алгоритми перевіряються на історичних даних, щоб переконатися в їхній ефективності перед реальною реалізацією.

В основі функції алгоритмічної торгівлі лежить кілька стратегій, які включають в себе:

- Маркет-мейкінг.
- Статистичний арбітраж.
- Алгоритмічне виконання.
- Високочастотна торгівля (HFT).

Алгоритми маркет-мейкінгу призначені для забезпечення ліквідності ринків шляхом безперервного розміщення лімітних ордерів з обох боків книги заявок. Стратегія спрямована на отримання прибутку від спреду між попитом і пропозицією, забезпечуючи при цьому ліквідність ринку. Маркет-мейкери відіграють вирішальну роль у спрощенні торгівлі для інших учасників ринку, гарантуючи, що на певний цінний папір завжди є покупець і продавець. Завдання цих алгоритмів полягає в тому, щоб збалансувати запаси цінних паперів і управляти пов'язаними з ними ризиками, якщо ринок рухається всупереч їхнім позиціям.

Стратегія статистичний арбітраж ґрунтується на складних математичних моделях, які намагаються виявити розбіжності в цінах між пов'язаними фінансовими інструментами. Використовуючи моделі середнього розвороту, парну торгівлю та коінтеграцію, алгоритми статистичного арбітражу намагаються передбачити поведінку цінових розбіжностей і використати їх для отримання прибутку. Ці розбіжності можуть виникати через неправильну оцінку або тимчасові дисбаланси між активами, які історично рухалися разом. Алгоритми можуть швидко скористатися цими можливостями, які можуть бути непомітними або недоступними для людей через швидкість і можливості обробки даних.

Алгоритмічне виконання застосовують зазвичай інституційні інвестори яким, часто потрібно купувати або продавати великі партії цінних паперів. Безпосереднє розміщення таких великих замовлень може суттєво вплинути на ринок, що потенційно може призвести до несприятливих цін. Щоб пом'якшити цей вплив, використовуються алгоритмічні стратегії виконання, такі як Iceberg і VWAP (Volume Weighted Average Price - середньозважена ціна за обсягом). Вони розбивають великі замовлення на менші частини, які поступово випускаються на ринок, часто за алгоритмічно визначеним графіком, що має на меті мінімізувати вплив на ринок і вартість торгівлі.

Алгоритми HFT характеризуються швидкістю і здатністю виконувати численні угоди за частки секунди. Ця стратегія використовує переваги дуже короткострокової неефективності ринку і швидкості, з якою можуть діяти алгоритми, часто реагуючи на ринкову інформацію або ордери раніше, ніж інші трейдери. Стратегія вимагає складних технологій, як з точки зору програмного забезпечення (алгоритмів), так і з точки зору обладнання (серверів, підключення), оскільки успіх HFT часто залежить від затримки, або швидкості, з якою торгова система може реагувати на ринкові події.

Кожна з цих стратегій використовує потужність комп'ютерів і складні математичні моделі для здійснення торгівлі зі швидкістю і точністю, які виходять за межі людських можливостей. Однак вони також вимагають ретельного тестування, надійних систем управління ризиками та постійного моніторингу, щоб гарантувати, що вони працюють так, як очікується, і не створюють надмірних ризиків для торговельної організації або ринку в цілому. З розвитком ринків і появою нових технологій ці стратегії постійно вдосконалюються, щоб адаптуватися до нових ринкових умов і правил.

У своїй роботі я буду використовувати мультиіндикаторну стратегію. В алгоритмічній торгівлі це стратегія, яка покладається на використання декількох технічних індикаторів для прийняття рішень. Цей вид стратегії є формою технічного аналізу, який поєднує різні індикатори для генерації більш надійних торгових сигналів.

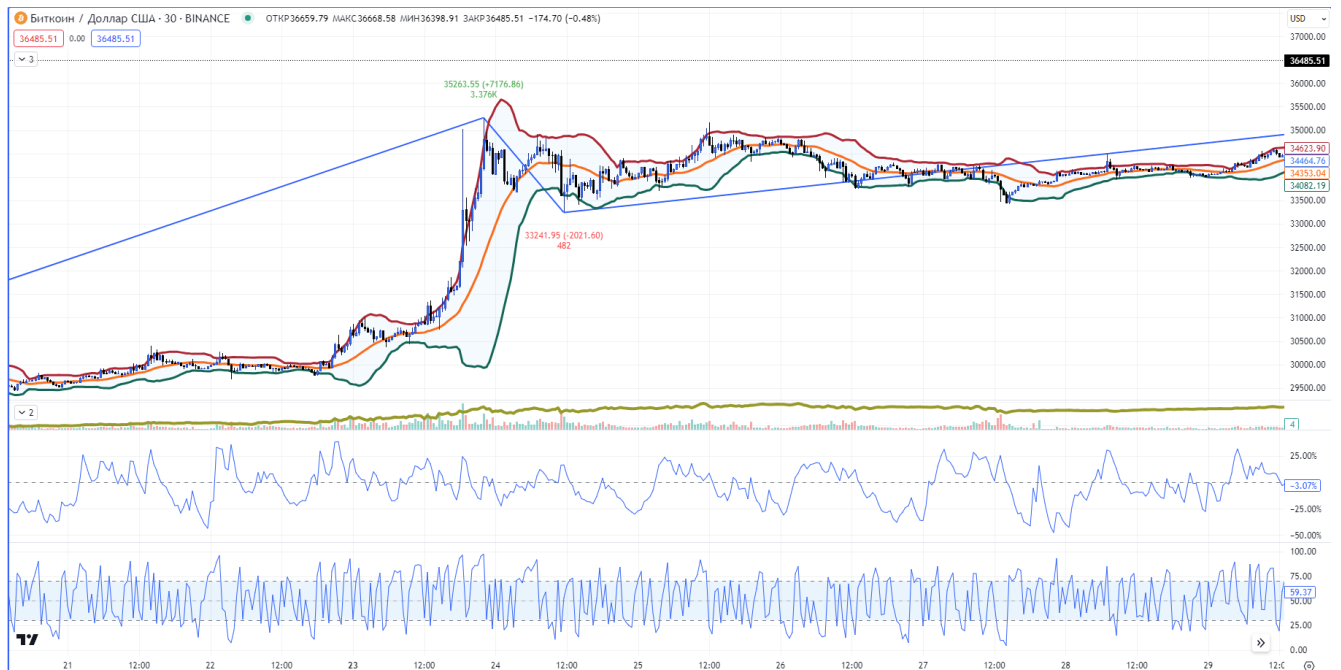


Рис. 1.4. Використання індикаторів на графіку

Алгоритмічна торгівля докорінно змінила динаміку ринку. Збільшення обсягів алгоритмічної торгівлі сприяє підвищенню складності та взаємопов'язаності ринку. Підвищуючи ефективність і прозорість, швидке виконання і великі обсяги ордерів можуть також сприяти волатильності ринку під час стресу, що потенційно може призвести до таких подій, як раптові обвали.

Отже, алгоритмічна торгівля відіграє ключову роль на сучасних фінансових ринках, характеризуючись здатністю підвищувати ліквідність, покращувати виконання, управляти ризиками та знижувати витрати. Постійний розвиток алгоритмічних методологій продовжує трансформувати торгові парадигми, що вимагає постійної оцінки та адаптації як з боку учасників ринку, так і з боку регуляторів.

Еволюція автоматизованих торгових систем, відомих як торгові боти, відіграла важливу роль у трансформації динаміки фінансових ринків. Ці алгоритмічні об'єкти здійснюють операції на основі заздалегідь встановленого набору правил, отриманих з технічного аналізу, кількісних показників і статистичних моделей, таким чином усуваючи латентність і емоційні упередження, пов'язані з торгівлею, яку здійснює людина.

По суті, торгові боти складаються зі складних математичних моделей, які обробляють ринкові дані в режимі реального часу, застосовуючи різноманітні статистичні методи та алгоритми машинного навчання для прогнозування руху цін і виявлення торгових можливостей. Технічна складність цих систем може варіюватися від простих тригерів на основі правил до просунутих систем штучного інтелекту, які адаптуються і навчаються на ринкових умовах за допомогою навчання з підкріпленням та інших адаптивних процесів.

Технічно архітектура торгового бота складається з декількох рівнів роботи. Модуль збору даних відповідає за отримання та обробку ринкових даних з бірж у режимі реального часу. Модуль аналізу використовує ці дані для обчислення різних торгових сигналів на основі заздалегідь визначених критеріїв, які можуть включати об'єднання декількох індикаторів, виявлення аномалій або розпізнавання шаблонів. Механізм прийняття рішень інтерпретує ці сигнали відповідно до правил торгової стратегії і визначає відповідну дію - ініціювати, модифікувати або вийти з угоди.

Алгоритми управління ризиками є невід'ємною частиною архітектури торгового бота, постійно відстежуючи несприятливі ринкові умови або потенційні загрози для торгового капіталу, а також динамічно регулюючи розміри угод і параметри стоп-лосс для захисту від значних збитків.

Модуль виконання перетворює результати роботи системи прийняття рішень на реальні ордери на купівлю або продаж, взаємодіючи з біржовими API для розміщення угод на ринку. Цей модуль є критично важливим для збереження переваги швидкого виконання, що є однією з головних переваг використання торгових ботів.

Безперервне зворотне тестування на історичних даних гарантує, що алгоритми залишаються актуальними та ефективними в умовах мінливого ринкового середовища. Крім того, форвардне тестування на змодельованих ринках дозволяє точно налаштувати параметри стратегії перед її розгортанням на реальних ринках.

Оскільки ці автоматизовані системи все більше вкорінюються в ринкові структури, їхній вплив на ліквідність, волатильність та ефективність ринку продовжує залишатися предметом технічного вивчення. Торгові боти не лише відображають передові фінансові технології, але й створюють нові виклики та міркування з точки зору регулювання ринку, етичних практик торгівлі та оцінки системних ризиків.

Отже, технічний розвиток торгових ботів є критичним моментом в еволюції фінансового ринку. Стаючи дедалі складнішими, ці системи вимагають не менш глибокого розуміння механізмів, що лежать в їхній основі, та ширших наслідків їхньої інтеграції в ринкову інфраструктуру.

1.4 ТЕХНІЧНІ АСПЕКТИ АВТОМАТИЗОВАНОЇ ТОРГІВЛІ НА КРИПТОВАЛЮТНОМУ РИНКУ

Торгові боти, по суті, є алгоритмічним програмним забезпеченням, призначеним для здійснення угод від імені трейдерів. Ці боти працюють на основі попередньо встановлених критеріїв, аналізуючи ринкові тенденції та виконуючи угоди у відповідні моменти. Основна функціональність цих ботів залежить від їхньої здатності безперешкодно взаємодіяти з біржовими платформами, що досягається завдяки використанню інтерфейсів API.

API слугують комунікаційним рівнем між торговими ботами та біржовими серверами. Вони відповідають за передачу торгових ордерів, отримання ринкових даних і управління обліковими записами користувачів. По суті, API - це канали, через які торгові боти отримують доступ до ринків.

Біржа Binance пропонує свій API, та я буду ще використовувати API, що надається Finandy для зв'язку з платформою TradingView.

Створення торгових ботів передбачає поєднання мов програмування і технологій, які підходять для обробки даних у режимі реального часу, алгоритмічної торгівлі та інтеграції з різними торговими платформами. Основні мови програмування та їх властивості наведенні нижче.

Python користується великою популярністю завдяки своїй простоті, зрозумілості та великій кількості бібліотек. Вона особливо добре підходить для аналізу даних і машинного навчання, які є критично важливими для розробки складних торгових алгоритмів.

JavaScript/Node.js ідеально підходить для розробки веб-ботів і додатків для торгівлі. JavaScript разом з Node.js дозволяє розробляти швидкі мережеві додатки.

Java підходить для високочастотних торгових ботів, які потребують швидкості та масштабованості.

C++ часто обирають для складних торгових алгоритмів, де швидкість виконання є критично важливою. У порівнянні з Python та Java вимагає більш глибоких знань для написання коду.

Мова програмування R особливо корисна для проведення складного аналізу даних і роботи з дослідницькими даними. В основному використовується для статистичних обчислень.

У своїй роботі я буду використовувати мову програмування Pine Script. Ця мова вбудована у платформу TradingView, та використовується для створення власних індикаторів технічного аналізу та торгових стратегій. Pine Script популярна мова серед трейдерів завдяки командам, які напряду стосуються сфери торгівлі. Однак це мова вимагає додаткових платформ для здійснення торгівлі, так як основна властивість аналіз і бек-тестування.



Рис. 1.5. Видяг редактора Pine Script

Автоматизована торгівля, пропонує багато переваг, такі як підвищена ефективність і можливість обробляти великі обсяги даних, але не слід виключати і ризики. Для трейдерів дуже важливо знати про ці ризики, щоб зменшити потенційні втрати.

До таких ризиків на сам перед слід віднести технічній збій, тож важливо щоб алгоритм був ретельно протестований. Помилки в програмному забезпеченні, проблеми з підключенням, це все може призвести до великих втрат.

Також важливим є конфіденційність та кібербезпека. Автоматизовані системи можуть бути вразливими до атак та злому, що може призвести до втрати всього капіталу. Важливо конфіденційну інформацію не розповсюджувати.

Застарілість моделі, чи програмного коду, може призвести також до втрат. Ринкові умови з часом змінюються, перестають працювати стратегії та покладатися на написану програму роками не розумно, тож завчасно треба робити оновлення програмного коду чи стратегії.

1.5 ТИПИ ТЕХНІЧНИХ ІНДИКАТОРІВ В АВТОМАТИЗОВАНИХ ТОРГОВИХ СИСТЕМАХ

Індикатори служать основою багатьох торгових алгоритмів, забезпечуючи системний підхід до аналізу ринку і прийняття рішень. Є різновиди індикаторів такі як трендові індикатори та осцилятори.

Трендові індикатори використовуються для визначення напрямку та сили тренду на ринку.

- Ковзаючі середні (МА)

Ковзаючі середні, включаючи прості ковзаючі середні (SMA) і експоненціальні ковзаючі середні (EMA), є трендовими індикаторами. Вони згладжують цінові дані за певний період, допомагаючи визначити напрямок руху ринку.

Автоматизовані торгові системи часто використовують ковзаючі середні для генерації сигналів, таких як купівля, коли короткострокова МА перетинає довгострокову МА, і продаж у зворотному сценарії. Простота і ефективність ковзних середніх роблять їх основним елементом в алгоритмах автоматизованої торгівлі.



Рис. 1.6. Трендовий індикатор МА

- Смоги Боллінджера

Смоги Боллінджера вимірюють волатильність ринку і дають уявлення про потенційні стани «перекупленості» або «перепроданості». Смоги складаються з SMA і двох ліній стандартного відхилення і адаптуються до волатильності ринку.

В автоматизованій торгівлі дотик або прорив цих смуг може спровокувати вхід або вихід з угоди, при цьому нижня смуга часто вказує на точку купівлі, а верхня - на точку продажу.



Рис. 1.7. Трендовий індикатор Смоги Боллінджера

- Відновлення Фібоначчі

Цей індикатор, заснований на послідовності Фібоначчі, використовується для визначення потенційних рівнів підтримки і опору. Автоматизовані торгові системи можуть використовувати ці рівні для встановлення стратегічних точок входу або виходу, вирівнюючи угоди з цими ключовими рівнями.



Рис. 1.8. Трендовий індикатор Відновлення Фібоначчі

Осцилятори зазвичай коливаються в межах певного діапазону (наприклад, від 0 до 100) і часто представлені у вигляді лінійного графіка під основним графіком ціни. Осцилятори, як правило, генерують чіткі сигнали, засновані на конкретних числових значеннях. Наприклад, RSI видає сигнал «купити», коли він перевищує 70, і сигнал «продати», коли опускається нижче 30. Ці чіткі критерії полегшують використання осциляторів в алгоритмічних торгових стратегіях.

- Індекс відносної сили (RSI)

RSI є осцилятором імпульсу, він має вирішальне значення для визначення швидкості руху цін. Він працює за шкалою від 0 до 100, сигналізуючи про перекупленість вище 70 і перепроданість нижче 30. В автоматизованих торгових системах RSI часто використовується для ініціювання ордерів на покупку, коли

ринок перепроданий, і ордерів на продаж, коли він перекуплений. Його прямий числовий вихід робить його особливо сприятливим для алгоритмічної торгівлі.



Рис. 1.9. Осцилятор RSI

- Ковзаюча середня конвергенція-дивергенція (MACD)

MACD, індикатор імпульсу, що слідує за трендом, показує взаємозв'язок між двома ковзними середніми ціни. В автоматизованих системах MACD використовується для виявлення потенційних можливостей для купівлі та продажу, в першу чергу через перетин лінії MACD та її сигнальної лінії. Крім того, розбіжність між MACD і ціною може вказувати на потенційні розвороти, що робить його цінним інструментом.



Рис. 1.10. Осцилятор MACD

- Осцилятори об'єму

Осцилятори об'єму - це категорія індикаторів, які вимірюють швидкість зміни об'єму. Вони використовуються разом з іншими індикаторами для підтвердження цінових тенденцій або сигналізації про потенційні розвороти. Значна зміна об'єму, особливо коли він розходиться з ціною, може надати цінну інформацію для прийняття рішень.



Рис. 1.11. Осцилятори об'єму

Ці індикатори, які використовуються для автоматизованої торгівлі, дозволяють здійснювати операції на основі конкретних, заздалегідь встановлених критеріїв. У стратегії автоматизованої торгівлі часто поєднують кілька індикаторів для підвищення точності сигналів і управління ризиками. Однак успіх цих стратегій значною мірою залежить від постійної оптимізації та адаптації до ринкових умов.

ВИСНОВКИ 1 РОЗДІЛУ

У першому розділі були пояснені основні характеристики та функції криптовалютних бірж, включаючи їх роль у забезпеченні ліквідності. Також акцентується увага на різновидах криптовалютних бірж. Перевага вибору централізованої криптовалютної біржі Binance.

Також були розглянуті основи автоматизованої торгівлі на криптовалютному ринку, включаючи її переваги, такі як здатність швидко реагувати на ринкові зміни, зменшення емоційного впливу на торгові рішення. Різкий розвиток систем привів до появи законів та правил, які відрізняються у різних країнах, але які потрібно дотримуватися при роботі з криптовалютною біржею.

Пояснено ключові принципи, які лежать в основі автоматизованої торгівлі, включаючи управління ризиками, вибір стратегій, налаштування параметрів торгівлі, використання API та важливість зворотнього тестування. Наведено приклади мов програмування, які використовуються для написання автоматизованих систем, включно з Pine Script, та основні типи технічних індикаторів, які використовуються в автоматизованих торгових стратегіях.

РОЗДІЛ 2: РОЗРОБКА ТА НАЛАШТУВАННЯ ТОРГОВОГО БОТА

2.1 ОРГАНІЗАЦІЙНА СТРУКТУРА ПРОЕКТУ

Організаційна структура проекту для розробки автоматизованої торгової системи-бота включає ієрархію та зв'язки між командами виглядає таким чином:

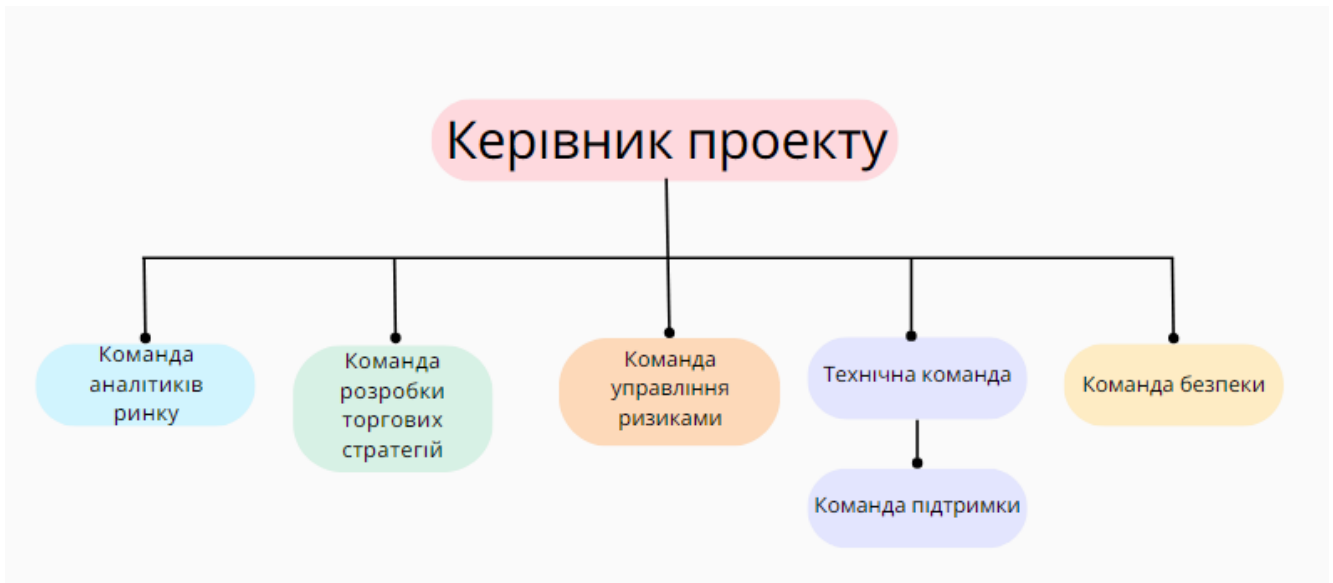


Рис. 2.1. Організаційна структура проекту

Комунікаційні зв'язки між командами побудовані таким чином, що інформаційні потоки течуть в обидві сторони, забезпечуючи двосторонній зв'язок для ефективного управління проектом. Команди взаємодіють одна з одною через регулярні зустрічі, спільні документи та проектні інструменти для співпраці, щоб забезпечити скоординовану роботу над проектом.

Керівник проекту:

- Знаходиться на вершині ієрархії та керує всім проектом.

Команда аналітиків ринку:

- Підпорядковується безпосередньо керівнику проекту.

- Постачає аналітичні дані для команди розробки торгових стратегій.

Технічна команда:

- Працює під керівництвом керівника проекту, виконуючи технічну реалізацію проекту.
- Взаємодіє з командою розробки торгових стратегій для інтеграції алгоритмів у торгову систему.

Команда розробки торгових стратегій:

- Працює під керівництвом керівника проекту і в тісній співпраці з командою аналітиків ринку.

Команда безпеки:

- Підзвітна керівнику проекту та співпрацює з технічною командою для забезпечення безпеки системи.

Команда підтримки:

- Надає підтримку користувачам та зв'язується з технічною командою для вирішення технічних проблем.

Команда управління ризиками:

- Підпорядковується керівнику проекту і взаємодіє з командою розробки торгових стратегій.

Цей проект є індивідуальним тому, основні ролі буду виконувати я.

Для зручності роботи, та послідовності процесу був розроблений план, поділений на етапи роботи (таблиця 2.1)

Таблиця

2.1 Етапи розробки автоматизованої торгової системи-бота

Етап роботи	Опис етапу	Завдання
1	Огляд та налаштування платформи TradingView для аналізу ринків і тестування стратегій.	Вивчення інтерфейсу TradingView. Налаштування графіків для аналізу. Знайомство з інструментами аналізу на платформі.
2	Детальний аналіз індикатора RSI (Relative Strength Index) та його застосування у торговій стратегії.	Визначення параметрів RSI. Аналіз сигналів RSI для входу та виходу з ринку. Інтеграція RSI у торгову стратегію.
3	Створення торгової стратегії за допомогою мови програмування Pine Script на платформі TradingView.	Основи Pine Script. Складання алгоритму торгової стратегії. Впровадження індикаторів у стратегію. Управління ризиками.
4	Проведення зворотного тестування (Backtest) розробленої стратегії для оцінки її ефективності.	Налаштування параметрів тестування. Аналіз результатів тестування. Оптимізація стратегії на основі отриманих даних.

5	Проведення аналізу система сповіщень TradingView.	Налаштування параметрів сповіщень у TradingView. Інтеграція сповіщень з торговими алгоритмами.
6	Створення Binance API ключів для автоматизації торгівлі та управління аккаунтом через сторонні додатки.	Реєстрація та налаштування API ключів на платформі Binance. Встановлення необхідних дозволів для торгових операцій.
7	Підключення сигналу від TradingView до Finandy для автоматичного виконання торгових операцій.	Отримання URL-адреси у Finandy. Налаштування умов прийому та обробки торгових сигналів.
8	Налаштування параметрів стоп-лосу в Finandy для управління ризиками.	Налаштування параметрів стоп-лосу. Тестування механізму закриття позицій.
9	Аналіз результатів укладення угод для оцінювання ефективності торгової стратегії бота.	Моніторинг торгових операцій. Аналіз прибутковості та збору статистики здійснених угод.

На цей проект було виділено фінансування у розмірі 10\$, для повного тестування укладання угод на криптовалютній біржі. Також вартість підписки на платформу TradingView Plus коштувала 30\$ на місяць. Загальні витрати на весь проект у мене повинні займати не більше 50\$.

Необхідне обладнання для реалізації:

- ПК чи ноутбук (для розробки та підключення системи).
- Телефон (для легкості моніторингу).
- Інтернет (для доступу у мережу, та оновлення даних).

2.2 ВИКОРИСТАННЯ ПЛАТФОРМИ TRADINGVIEW

TradingView - це сучасна веб-платформа для побудови графіків, яка широко використовується для технічного аналізу на фінансових ринках. Вона інтегрує поточні дані в реальному часі з різних класів активів, включаючи акції, форекс, криптовалюти та ф'ючерси. Платформа розроблена для підтримки широкого спектру інструментів технічного аналізу, з більш ніж 100 попередньо створеними індикаторами і більш ніж 50 інтелектуальними інструментами малювання.

Ключовою технічною особливістю TradingView є можливість алгоритмічної торгівлі та розробки стратегій. Платформа надає комплексне середовище бек-тестування, яке дозволяє ретельно тестувати і перевіряти торгові стратегії на основі історичних даних. На даний момент я маю підписку TradingView Plus, що дозволяє мені використовувати до 10 індикаторів на графіку та мати можливість бачити та використовувати 10 тисяч історичних барів, та 100 оповіщень по індикаторам. Кількість оповіщень важлива для мене, бо саме це буде слугувати сигналом для укладання угод.

TradingView обробляє ринкові дані в режимі реального часу, що має вирішальне значення для торгівлі на криптовалютному ринку. Інфраструктура платформи оптимізована для передачі даних з низькою затримкою, що забезпечує своєчасне оновлення графіків та індикаторів. Крім того, хмарна архітектура TradingView забезпечує доступ з різних пристроїв. Якщо встановити додаток на телефон, то це дає велику мобільність дій, та швидкий доступ відстеження позиції. Можна використовувати веб-додаток, щоб не встановлювати програму на інших ПК.



Рис. 2.2. Вигляд графічної платформи TradingView на ПК



Рис. 2.3. Вигляд додатку графічної платформи TradingView на телефоні

В основі TradingView вбудований редактор Pine Script, він є власною мовою програмування, орієнтованої на конкретну галузь. Pine Script призначений для кодування індикаторів технічного аналізу та автоматизованих торгових стратегій. Він забезпечує поєднання простоти для написання базових скриптів і надійності,

необхідної для складних алгоритмічних конструкцій. Мова підтримує змінні, функції та керуючі структури, що дозволяє писати складні логічні операції та математичні функції.

Для свого торгового бота я буду використовувати принцип торгівлі за допомогою стратегії. Для покращення результату буду поєднувати 3 індикатори за допомогою яких покращується шанс для вдалого укладання угоди.

2.3 ОПИС ТЕХНІЧНИХ ІНДИКАТОРІВ У СТРАТЕГІЇ RSI

Перед тим як приступати до розробки стратегії треба обрати основні технічні індикатори, за допомогою яких буде відбуватися аналіз.

Я обрала один основний індикатор на основі якого і буде будуватися стратегія, та два додаткових, для покращення та більшої результативності роботи.

1. Індекс відносної сили (RSI) – основний індикатор
2. Індекс стиснення (SI) – додатковий індикатор
3. Осцилятор об'єму (VO) – додатковий індикатор

Індекс відносної сили (RSI) - це осцилятор імпульсу, який зазвичай використовується в технічному аналізі для вимірювання швидкості і зміни цінних рухів.

RSI формула (1) коливається в діапазоні від нуля до 100 і використовується для визначення перекупленості або перепроданості активу, що торгується на ринку.

Формула RSI:

$$RSI = 100 - (100 / RS) (100 / (1 + RS)),$$

(1)

де RS - це середнє значення закриття за x днів, поділене на середнє значення за x днів, поділене на середнє значення закриття за x днів, стандартне значення для x становить 14 днів, хоча його можна змінювати для збільшення або зменшення чутливості RSI

Значення RSI вище 70 вважаються перекупленістю, що вказує на можливість розвороту або відкату, тому коли позначка буде проходити цю границю та вертати до неї у мене буде відкриватися коротка позиція. Значення нижче 30 вказує на перепроданість, що потенційно може вказувати на розворот цінового руху активу, і відкриття довгої позиції. Саме на основі цього індикатора і буде будуватися моя стратегія, тому і назва стратегії «RSI Strategy».



Рис. 2.4. Технічний індикатор RSI

Індекс стиснення (SI) - це індикатор, який використовується для оцінки волатильності та імпульсу ринку. Стиснення вказує на низьку волатильність і може вказувати на періоди підвищеної волатильності та потенційні стрибки цін.

Індекс стиснення розраховується шляхом аналізу відносин між Смугами Боллінджера і Каналом Кельтнера. Смуги Боллінджера побудовані з використанням стандартного відхилення ціни за певний період часу, тоді як Канал Кельтнера використовують середній діапазон істинності (ATR). Коли Смуги Боллінджера розташовані всередині Канала Кельтнера, це вказує на період низької волатильності, що може сигналізувати про можливий рух цін в майбутньому.

Осцилятор об'єму (VO) - це індикатор імпульсу в торгівлі, який вимірює різницю між двома середніми обсягу, щоб допомогти визначити тенденції в об'ємі. Він розраховується шляхом віднімання більш довгострокової ковзної середньої об'єму від більш короткострокової ковзної середньої, різниця потім зазвичай будується у вигляді гістограми або лінії для візуалізації взаємозв'язку між двома ковзними середніми.

Загальними налаштуваннями для ковзних середніх є 5-денний період для короткострокової ковзної середньої і 20-денний період для довгострокової.

Позитивне значення VO вказує на те, що короткостроковий обсяг вищий за довгостроковий середній, що свідчить про зростання торгової активності та потенційний зростаючий тренд, і навпаки, негативне значення вказує на те, що короткостроковий об'єм нижчий за довгостроковий середній, що може свідчити про падаючий настрій або зниження торгової активності.

VO часто використовується для підтвердження цінових тенденцій або для передбачення розворотів, якщо спостерігається розбіжність між ціновими та об'ємними трендами.

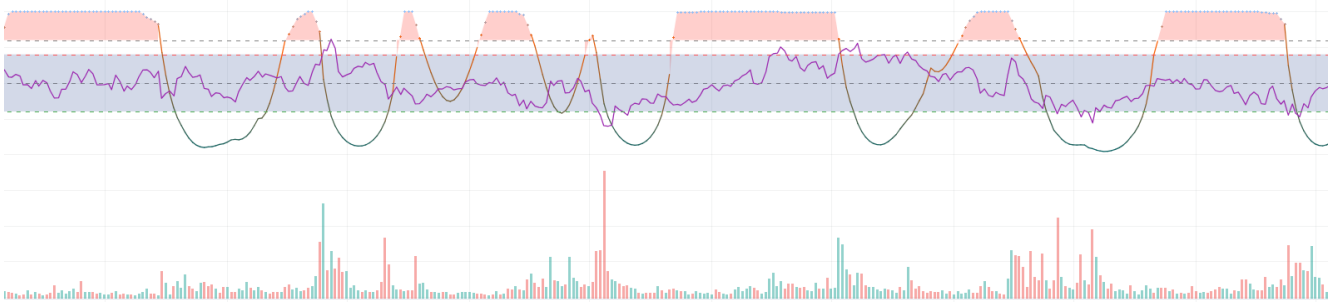


Рис. 2.5. Графічне відображення поєднання технічних індикаторів SI та VO

2.4 РОЗРОБКА СТРАТЕГІЇ НА PINE SCRIPT

Мова програмування Pine script розроблена спеціально для написання торгових стратегій, які в подальшому можна використовувати для автоматизованої системи. Тому більшість функцій та формул оптимізовані для людей, які мають знання у технічному аналізі ринку, що допоможуть розібратися з знаннями коду. Це для мене має важливу роль, тому, що маючи такі знання, я зможу самостійно розібратися з написанням стратегії. Також у відкритому доступі є допоміжна література, де можна знайти всі необхідні основні функції та як їх поєднувати.

Спочатку треба запустити саму платформу TradingView. Потім вже створити новий графік, та відкрити редактор Pine для написання самої стратегії.

Далі я буду наводити поетапні виконання коду, та описувати кожен частину.



```
1 //@version=5
2 strategy("RSI Strategy", overlay=true)
3
```

Рис. 2.6. Підключення стратегії RSI

Цей рядок коду в Pine Script використовується для створення торгової стратегії у TradingView.

Спершу ми прописуємо функцію `strategy`, яка використовується для вказівки, що подальший код є частиною торгової стратегії, а не простого індикатора.

У цьому випадку, стратегія названа «RSI Strategy». Назва з'являється у вікні інформації про стратегію на графіку та у списку стратегій користувача. Також у вікні інформації про стратегію назву можна змінити, як це зробила я на просто «Strategy».

Параметр `overlay=true` вказує, що стратегія має бути відображена безпосередньо на основному графіку ціни активу. Якщо `overlay` встановлено у `false`, стратегія відображатиметься у власному окремому вікні під графіком.

Цей рядок коду важливий, оскільки він визначає основу для стратегії. Всі подальші вказівки, щодо входу в позиції, виходу з них, встановлення стоп-лосів та інші елементи стратегії, будуть побудовані в контексті цієї створеної стратегії.

Наступним кроком є написання фрагменту коду, який буде використаний для створення індикаторів у стратегії.

```
Strategy  ✎ ☆ 🔍 4.0 ▾
1 //@version=5
2 strategy("RSI Strategy", overlay=true)
3
4 //Inputs
5 string conftype = input.string("All","Entry Method", options = ["RSI", "RSI+SI", "RSI+VO", "All"], group = "Position Initiation Settings")
6 length = input(14, title = "RSI Lenght", group = "RSI Settings")
7 overSold = input.float(30, minval = 0, maxval = 100, title = "Oversold Zone", group = "RSI Settings")
8 overBought = input.float(70, minval = 0, maxval = 100, title = "Overbought Zone", group = "RSI Settings")
9 RSI_Safe_Zone = input(40, title = "RSI Safe Zone", group = "RSI Settings")
10 price = input(close, title = "RSI Source", group = "RSI Settings")
11 sq_conv = input(50, title = "Convergence Factor", group = "Squeeze Index Settings")
12 sq_length = input.int(20, minval = 1, title = "Squeeze Index Lenght", group = "Squeeze Index Settings")
13 sq_src = input(close, title = "Squeeze Index Source", group = "Squeeze Index Settings")
14 shortlen = input.int(5, minval=1, title = "Volume Oscillator Fast MA", group = "Volume Oscillator Settings")
15 longlen = input.int(10, minval=1, title = "Volume Oscillator Slow MA", group = "Volume Oscillator Settings")
16 StopLoss = input.float(1, minval = 0.01, title = "Stop Loss (%)", group = "Risk Managment") / 100
17 profit = input.float(1, minval = 0.01, title = "Take Profit (%)", group = "Risk Managment") / 100
18
```

Рис. 2.7. Підключення налаштувань стратегій RSI RSI+SI RSI+VO та All

Лінія коду `string conftype = input.string(...)` створює введення для вибору методу входу у позицію. За допомогою чого ми можемо обирати у меню налаштувань індикатори, якими користуватися. Для зручності я поділила на RSI, RSI+SI, RSI+VO та All. Це було зроблено, для аналізу кращого результату, та виключення одного з індикаторів у разі потреби.

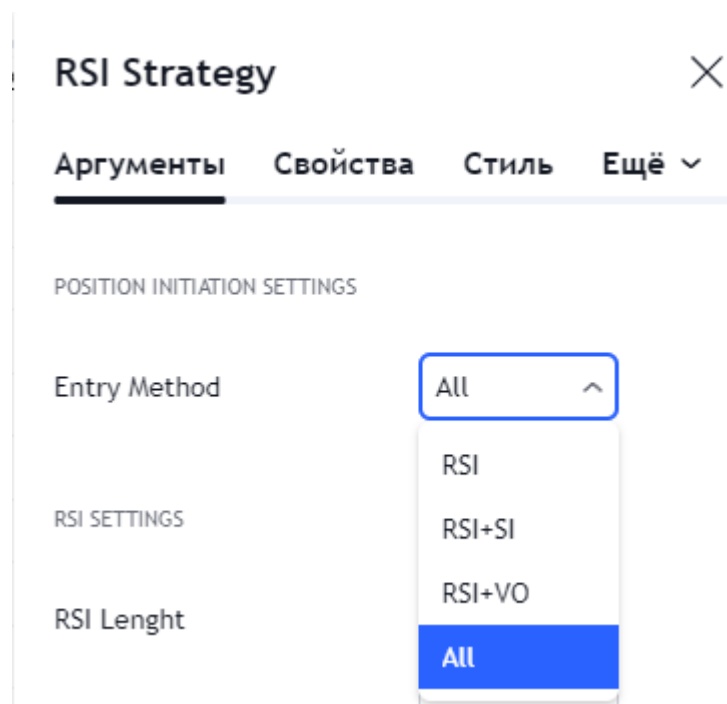


Рис. 2.8. Меню вибора індикаторів

Додаємо основні значення для індикатора RSI. `Length = input(14, ...)` Встановлює довжину для індикатора RSI (Relative Strength Index). У стандартному випадку це - 14 періодів. `OverSold = input.float(30, ...)` та `overBought = input.float(70, ...)` Визначають зони перепроданості та перекупленості для RSI. Тут вони

встановлені як 30 і 70 відповідно це зазвичай стандартні значення, але у подальшому їх можливо змінювати. `RSI_Safe_Zone = input(40, ...)` визначає «безпечну зону» для RSI, у цьому випадку стандартне значення - 40. `Price = input(close, ...)` визначає джерело ціни для розрахунку RSI, зазвичай використовується поточна ціна закриття.

Відповідно додаємо основні значення для SI та VO.

Параметри `sq_conv = input(50, ...)`, `sq_length = input.int(20, ...)`, `sq_src = input(close, ...)` налаштовують індекс стиснення (SI), включаючи коефіцієнт конвергенції, довжину і джерело ціни.

Також встановлюємо основні параметри для VO швидкої (Fast MA) та повільної (Slow MA) рухомих середніх у налаштуваннях осцилятора об'єму відповідно `shortlen = input.int(5, ...)`, `longlen = input.int(10, ...)`.

Розмір стоп-лосу та прибутку у відсотках відповідно `StopLoss = input.float(1, ...)`, `profit = input.float(1, ...)`.

У всіх параметрах я встановлювала стандартні значення для індикаторів, але за допомогою меню налаштування їх можна корегувати під свої потреби. Тож остаточне меню виглядає так:

Section	Parameter	Value
POSITION INITIATION SETTINGS	Entry Method	RSI
	RSI Length	14
	Oversold Zone	30
	Overbought Zone	70
	RSI Safe Zone	40
	Данные для RSI	закр
SQUEEZE INDEX SETTINGS	Convergence Factor	50
	Squeeze Index Length	20
	Squeeze Index Source	закр
VOLUME OSCILATOR SETTINGS	Volume Oscilator Fast MA	5
	Volume Oscilator Slow MA	10

Рис. 2.9. Меню налаштування індикатора RSI, SI та VO

RISK MANAGMENT

Stop Loss (%)

Take Profit (%)

Рис. 2.10. Меню налаштування Стоп-лосу та Тейк-профіту

```
//Style
col_0 = input(█#ffeb3b, 'Gradient'
, inline = 'inline0'
, group = 'Style')

col_1 = input(█#ff5d00, ''
, inline = 'inline0'
, group = 'Style')

col_2 = input(█#ff1100, ''
, inline = 'inline0'
, group = 'Style')
```

Рис. 2.11. Частина коду яка відповідає за параметри кольору індикатора

Створюємо індикатор стиснення та додаємо основні значення для побудови. Цей показник, що використовується для аналізу волатильності на фінансових ринках.

```
//Squeeze index
var max = 0.
var min = 0.
max := nz(math.max(sq_src, max - (max - sq_src) / sq_conv), sq_src)
min := nz(math.min(sq_src, min + (sq_src - min) / sq_conv), sq_src)
diff = math.log(max - min)
psi = -50 * ta.correlation(diff, bar_index, sq_length) + 50
```

Рис. 2.12. Частина коду яка відповідає за SI

Var використовується для збереження значення змінних між різними барами на графіку. Це означає, що їхні значення будуть оновлюватися, але не повністю перезаписуватися з кожним новим баром.

$Max := nz(\text{math.max}(sq_src, max - (max - sq_src) / sq_conv), sq_src)$
оновлюється значення max. Функція math.max порівнює поточне значення sq_src (джерело для розрахунку індексу стиснення) з попереднім максимальним значенням, відкоригованим на фактор конвергенції sq_conv. Функція nz (non-zero) використовується для запобігання нульових значень, забезпечуючи, що max буде хоча б рівне sq_src.

$Min := nz(\text{math.min}(sq_src, min + (sq_src - min) / sq_conv), sq_src)$ аналогічно оновлює min. Використовуючи math.min, визначається мінімальне значення між поточним sq_src та попереднім мінімальним значенням, скоригованим на фактор конвергенції.

$Diff = \text{math.log}(max - min)$ розраховує логарифм різниці між max і min. Це вимірює відстань між поточними максимальними та мінімальними значеннями sq_src, додаючи індикатор волатильності.

Показник PSI, який є кореляцією між diff та bar_index (індексом поточного бару) за останні sq_length барів розраховується як $psi = -50 * \text{ta.correlation}(diff, bar_index, sq_length) + 50$. Результат масштабується так, щоб коливатися навколо 50.

Індекс стиснення використовується для виявлення періодів низької волатильності, які часто передують періодам високої волатильності.

Наступним кроком додаємо індикатор VO, який є інструментом технічного аналізу, що використовується для вимірювання тенденцій у торговельному об'ємі.

```

//Volume Oscillator
var cumVol = 0.
cumVol += nz(volume)
if barstate.islast and cumVol == 0
|   runtime.error("No volume is provided by the data vendor.")
short = ta.ema(volume, shortlen)
long = ta.ema(volume, longlen)
osc = 100 * (short - long) / long

```

Рис. 2.13. Частина коду яка відповідає за VO

Також додаємо змінну `var`, яка зберігатиме своє значення між різними оновленнями барів на графіку. `Var cumVol = 0`, кумулятивний об'єм з початковим значенням 0.

Додаємо до кумулятивного об'єму поточний торговий об'єм, та використовуємо функцію `nz` для запобігання нульових значень, замінюючи їх на 0.

Перевіряємо, чи є поточний бар останнім у серії `barstate.islast` і чи кумулятивний об'єм дорівнює нулю. Якщо це так, це означає, що дані об'єму відсутні.

Якщо об'єм відсутній, виводиться помилка, повідомляючи, що постачальник даних не надає інформацію про об'єм.

`Short = ta.ema(volume, shortlen)` та `long = ta.ema(volume, longlen)` розраховують короткотермінову (`short`) та довготермінову (`long`) експоненційні рухомі середні (ЕМА) для об'єму. Значення `shortlen` та `longlen` визначають періоди для цих рухомих середніх.

`I osc = 100 * (short - long) / long`, розраховує сам об'ємний осцилятор `osc`. Він представляє собою відсоткове відношення між різницею короткотермінової та довготермінової ЕМА об'єму до довготермінової ЕМА.

Цей об'ємний осцилятор може використовуватися для виявлення тенденцій у торговельному об'ємі. Наприклад, значний ріст осцилятора може вказувати на підвищений інтерес до активу, тоді як спад може вказувати на зменшення

інтересу.

Тепер потрібно додати «Захисту від тренду» він аналізує високі (high) та низькі (low) значення цін, а також середній діапазон істинного руху (ATR) для визначення потенційних трендів.

```
//Trend Protection
curhi = request.security(syminfo.tickerid, timeframe = "1D", expression = high)
prehi = request.security(syminfo.tickerid, timeframe = "1D", expression = high[1])
curlo = request.security(syminfo.tickerid, timeframe = "1D", expression = low)
prelo = request.security(syminfo.tickerid, timeframe = "1D", expression = low[1])
datr = request.security(syminfo.tickerid, timeframe = "1D", expression = ta.atr(14))

lol = curhi > prehi and ((prehi - prelo) > datr) or ((curhi - curlo) > datr)
sol = curlo < prelo and ((prehi - prelo) > datr) or ((curhi - curlo) > datr)
lolo = not lol
solo = not sol
```

Рис. 2.14. Частина коду яка відповідає за реалізацію «Захисту від тренду»

Змінна `curhi` (поточний максимум) отримує високе значення ціни (high) активу для поточного дня (1D). Змінна `prehi` (попередній максимум) отримує високе значення ціни активу для попереднього дня.

Аналогічно, `curlo` (поточний мінімум) та `prelo` (попередній мінімум) отримують низькі значення ціни для поточного і попереднього днів відповідно.

`Datr` - це середній діапазон істинного руху (ATR) за останні 14 днів.

Логічна змінна `lol` (Long Open Logic) стає true, якщо поточний максимум вищий за попередній і якщо різниця між попередніми максимумом і мінімумом більша за ATR, або якщо різниця між поточними максимумом і мінімумом більша за ATR.

Аналогічно, `sol` (Short Open Logic) стає true, якщо поточний мінімум нижчий за попередній і виконуються ті ж умови щодо ATR.

Змінні `lolo` та `solo` є протилежними до `lol` та `sol` відповідно. Вони використовуються для визначення, коли не варто входити в довгу (`lolo`) або коротку (`solo`) позицію.

Цей код допомагає визначати, коли ціна активу показує певний тренд (наприклад, послідовне зростання або зниження високих/низьких значень) у порівнянні з попереднім днем, а також перевіряє волатильність за допомогою ATR для підтвердження сили тренду.

```
//Functions
vrsi = ta.rsi(price, length)
co = ta.crossover(vrsi, overSold) and vrsi <= 0 + RSI_Safe_Zone
cu = ta.crossunder(vrsi, overBought) and vrsi >= 100 - RSI_Safe_Zone
sc = psi < 80
voc = osc > 0
```

Рис. 2.15. Індикатори для генерації сигналів для визначення можливих точок входу або виходу

Кожна функція в цьому коді використовує різні індикатори для генерації сигналів для визначення можливих точок входу або виходу.

Спочатку іде розрахунок індикатора відносної сили індексу (RSI), де price - це вхідний параметр, який вказує на ціну активу, а length - це період для розрахунку RSI.

Після чого функція co визначає сигнал на покупку, який виникає, коли RSI перетинає вгору рівень перепроданості (overSold) і знаходиться в межах «Безпечної зони». «Безпечна зона» (RSI_Safe_Zone) використовується для додаткового фільтрування сигналів.

Аналогічно, cu визначає сигнал на продаж, який виникає, коли RSI перетинає вниз рівень перекупленості (overBought) і також знаходиться в межах визначеної «Безпечної зони».

Функція sc використовується для визначення, чи індекс стиснення (psi) є меншим за 80. Це може вказувати на потенційну волатильність або зміну тренду.

Функція voc перевіряє, чи об'ємний осцилятор (osc) має позитивне значення. Позитивні значення osc можуть вказувати на висхідний тренд в об'ємі торгів.

Для обмеження збитків, та довгого перебування у позиції, треба додати систему стоп-лосу, а також для вчасної фіксації профіту. Так як стратегія RSI будується на вході у зоні перекуленості, перепроданості, така система ризиків необхідна для запобігання знаходження у неволатильних та флетових ситуаціях на ринку.

```
//Risks
lstop = strategy.position_avg_price * (1 - StopLoss)
sstop = strategy.position_avg_price * (1 + StopLoss)

//Take Profits
lprofit = strategy.position_avg_price * (1 + profit)
sprofit = strategy.position_avg_price * (1 - profit)
```

Рис. 2.16. Додавання ризику та тейкпрофіту

Lstop визначає рівень стоп-лосу для довгих позицій. Strategy.position_avg_price вказує на середню ціну поточної позиції. Цей рівень стоп-лосу розраховується як відсоток (StopLoss) нижче середньої ціни позиції.

Схожим чином, для коротких позицій стоп-лос встановлюється як відсоток вище середньої ціни позиції. Це забезпечує захист від збитків, обмежуючи потенційні втрати, якщо ринок рухається проти позиції.

Lprofit встановлюється рівень для реалізації прибутку на довгих позиціях. Це робиться шляхом збільшення середньої ціни позиції на заданий відсоток (profit). Для коротких позицій рівень реалізації прибутку sprofit встановлюється як відсоток нижче середньої ціни позиції.

Ці рівні допомагають автоматизувати процес управління ризиками і прибутками у стратегії, дозволяючи заздалегідь визначити, при яких умовах буде закрита позиція для зменшення збитків або фіксація прибутку.

```

//Position Initiation Method
bool is_long = switch conftype
{
    "RSI" => co
    "RSI+SI" => co and sc
    "RSI+VO" => co and voc
    "All" => co and sc and voc
}
bool is_short = switch conftype
{
    "RSI" => cu
    "RSI+SI" => cu and sc
    "RSI+VO" => cu and voc
    "All" => cu and sc and voc
}

```

Рис. 2.17. Визначення позиції за допомогою торгових сигналів

Цей фрагмент коду визначає логіку визначення позицій для купівлі (long) та продажу (short) у залежності від обраного методу позиції (conftype). Це робиться за допомогою комбінацій різних торгових сигналів, які були визначені раніше в кодї.

Рядок `bool is_long` використовує конструкцію `switch` для встановлення логічної змінної `is_long`, яка визначає, чи потрібно відкривати довгу позицію. Значення `is_long` залежить від обраного методу ініціації (conftype).

- RSI: Встановлює `is_long` в `true`, якщо сигнал на покупку (co) є активним.
- RSI+SI: Вимагає активації сигналу на покупку (co) та сигналу індексу стиснення (sc) для встановлення `is_long` в `true`.
- RSI+VO: Потребує активації сигналу на покупку (co) та позитивного об'ємного осцилятора (voc) для встановлення `is_long` в `true`.
- All: Вимагає активації всіх трьох сигналів (co, sc, voc) для встановлення `is_long` в `true`.

Аналогічно `bool is_short`, цей рядок використовує `switch` для встановлення логічної змінної `is_short`, яка визначає, чи потрібно відкривати коротку позицію.

- RSI: Встановлює `is_short` в `true`, якщо сигнал на продаж (cu) є активним.

- RSI+SI: Потребує активації сигналу на продаж (cu) та сигналу індексу стиснення (sc) для встановлення is_short в true.
- RSI+VO: Вимагає активації сигналу на продаж (cu) та позитивного об'ємного осцилятора (voc) для встановлення is_short в true.
- All: Вимагає активації всіх трьох сигналів (cu, sc, voc) для встановлення is_short в true.

Кожен із цих методів використовує різні комбінації сигналів для визначення, коли відкривати довгі або короткі позиції. Це дозволяє гнучко налаштовувати стратегію під різні умови ринку.

```
//Executions
if (not na(vrsi))
|   if is_long and (lol or sol)
|   |   strategy.entry(id = "L", direction = strategy.long, comment = "L")
|   if is_short and (lol or sol)
|   |   strategy.entry(id = "S", direction = strategy.short, comment = "S")

//Stops and Takes
if strategy.position_size > 0
|   strategy.exit(id = "LX", stop = lstop, limit = lprofit)
if strategy.position_size < 0
|   strategy.exit(id = "SX", stop = sstop, limit = sprofit)

//plot(strategy.equity, title="equity", color=color.red, linewidth=2, style=plot.style_areabr)
```

Рис. 2.18. Виконання та реалізація торгових операцій

Реалізація логіки виконання торгових операцій, встановлення стоп-лосів і тейк-профіту а також для відображення кривої прибутковості стратегії.

If (not na(vrsi)) перевіряє, чи індекс відносної сили (RSI) не є відсутнім (na). Це означає, що операції будуть виконуватися тільки тоді, коли індекс RSI доступний.

If is_long and (lol or sol) перевіряє, чи має місце умова для входу в довгу позицію (is_long) та чи активований хоча б один з додаткових сигналів (lol або sol). Якщо умови виконуються, стратегія реєструє входження в довгу позицію.

Тоді виконується вхід у довгу позицію з ідентифікатором L та коментарем L.

Аналогічно, перевіряє, чи має місце умова для входу в коротку позицію (`is_short`) та чи активований хоча б один з сигналів (`lol` або `sol`). Якщо умови виконуються, стратегія реєструє входження в коротку позицію. Виконує вхід у коротку позицію з ідентифікатором `S` та коментарем `S`.

If `strategy.position_size > 0` перевіряє, чи є відкрита довга позиція. Та встановлює стоп-лос (`lstop`) та тейк-профіт (`lprofit`) для довгої позиції (`LX`).

If `strategy.position_size < 0` перевіряє, чи є відкрита коротка позиція. Та встановлює стоп-лос (`sstop`) та ціль для прибутку (`sprofit`) для короткої позиції (`SX`).

Закоментована лінія коду призначена для відображення кривої прибутковості стратегії на графіку. Це дозволяє візуально оцінити ефективність стратегії протягом часу.



Рис. 2.19. Гафічне використання стратегії на графіку

Код дозволяє автоматизувати процес входу в позиції та встановлення стоп-лосів і тейк-профіту, забезпечуючи управління ризиками та реалізацію прибутку.

```

//Plots
css1 = color.from_gradient(psi, 0, 80, col_0, col_1)
css2 = color.from_gradient(psi, 80, 100, css1, col_2)
plot_0 = plot(psi, 'PSI', psi > 80 ? na : css2)
plot(psi, 'Dots', psi > 80 ? css2 : na, style = plot.style_cross)

plot_1 = plot(80, display = display.none, editable = false)

fill(plot_0, plot_1, psi < 80 ? na : color.new(#ff1100, 80))

plot(vrsi, color = color.purple, display = display.pane)
//
hline(50, color = #787B86, title="Zero")
hline(80)
hb = hline(overBought, color = color.red, linewidth = 1)
hs = hline(overSold, color = color.green, linewidth = 1)
fill(hb, hs, color = color.rgb(65, 89, 156, 77))

```

Рис. 2.20. Фрагмент коду, який відповідає за візуалізацію елементів

Також треба додати візуалізацію різних елементів на графіку, зокрема індексу стиснення (PSI), індексу відносної сили (RSI), а також різних рівнів і ліній, для зручності використання, та розуміння де що знаходиться.

Для візуалізації коли PSI знаходиться між 0 та 80 колір змінюється col_0` до col_1, і від css1 до col_2, коли PSI знаходиться між 80 та 100. Якщо PSI більше 80, лінія не відображається, а створюється додатковий візуал у вигляді хрестиків. Також створюється додаткова лінія на 80, яка не відображається, але заповнює рівень між лінією та простором, коли позначка вище 80.

Також для візуалізації лінії RSI, використовується фіолетовий колір. Для зручності доданий нейтральний рівень 50. Та візуалізація зон перекупленості та перепродавності, які заповнюються для зручності.

Візуальний вигляд індикаторів показаний на рис та рис у розділі 2.2 опис технічних індикаторів у стратегії RSI.

2.5 ЗВОРОТНЕ ТЕСТУВАННЯ СТРАТЕГІЇ (BACKTEST)

Зворотне тестування стратегії, відоме також як backtest, - це процес, за допомогою якого можна оцінити ефективність своїх торгових стратегій, використовуючи історичні дані.

Ідея полягає в тому, щоб перевірити, як би стратегія працювала в минулому, щоб розуміти її потенційну ефективність у майбутньому. Цей процес дозволяє виявляти і відпрацьовувати потенційні слабкості стратегії, а також адаптуватися до змінюваних ринкових умов.

Платформа TradingView дозволяє проводити зворотне тестування стратегій. TradingView має вбудовані інструменти для цього, включаючи редактор Pine Script, який дозволяє писати та тестувати свої власні торгові скрипти.

Під час зворотного тестування на TradingView, можна використовувати різні індикатори та історичні дані, щоб оцінити, як стратегія б вела себе в різних ринкових сценаріях. Це дозволяє визначити потенційну прибутковість та ризики, пов'язані з їхніми стратегіями, перш ніж використовувати їх у реальній торгівлі.

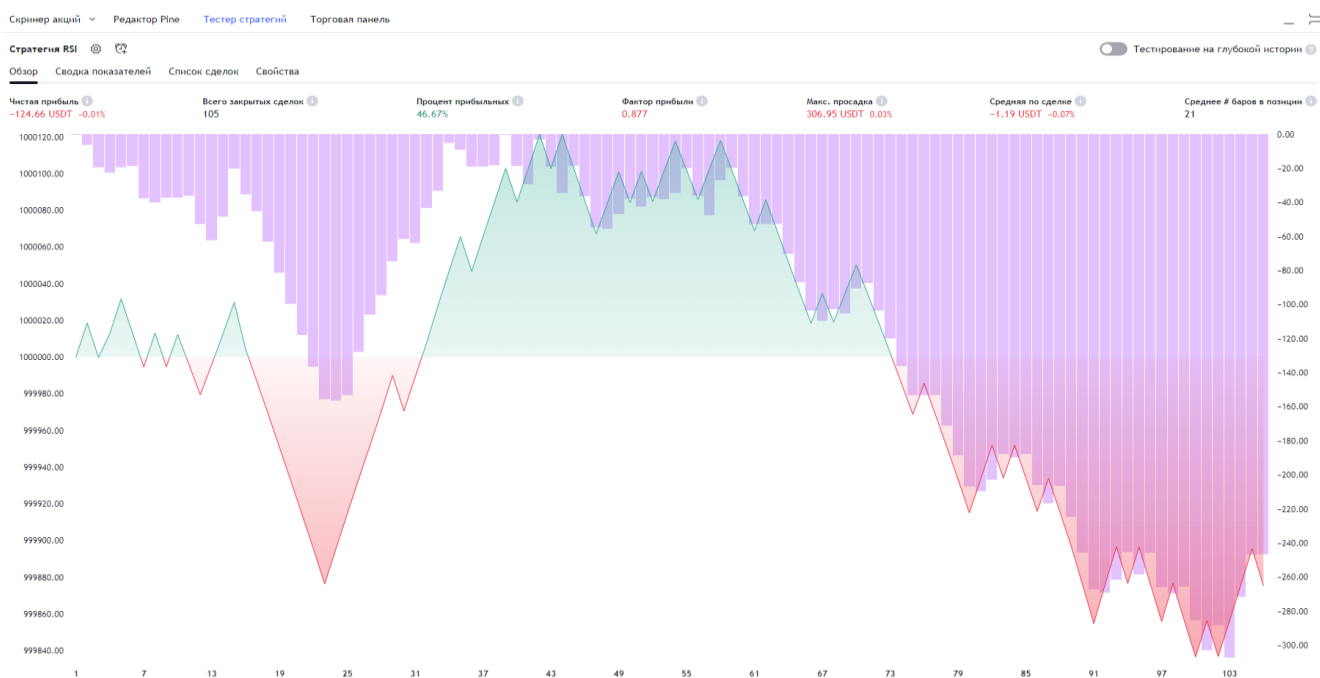


Рис. 2.21. Вигляд тестування стратегії без налаштування

Такий вигляд має графік при зворотному тестуванні без додавання додаткових параметрів для стратегії. Якщо надати доступ до торгівлі у такому

вигляді, то така стратегія принесе збитки. Важливо добре все налаштувати, провести аналіз найкращого тайм-фрейму для графіку, та налаштувань стратегії.

Тайм-фрейм у торгівлі та аналізі ринків, відноситься до обраного періоду часу, який використовується для аналізу цінових графіків та ринкових даних. Він може варіюватися від дуже коротких періодів, як, наприклад, хвилини або години, до більш тривалих, таких як дні, тижні, місяці або навіть роки.

Я буду використовувати короткі тайм-фрейми, щоб досягти швидкого результату, перебуваючи в угоді найменший проміжок часу.

У першу чергу треба провести аналіз кращих результатів на графіку з використанням різного тайм-фрейму. Аналіз я проводжу з використанням таблиці Ексел для візуальної зручності.

ТФ	Прибуток	Кі-сть угод	Він-рейт	Фактор прибутку	Макс просадка	Ср. по угоді	Ср. барів у позі	Тест період (Д)	Ср. Профіт/Місяць	Ср. Ризик/Місяць
1	\$866.10	103	59.22%	1.168	\$1 734.30	\$8.41	138	23	\$1 129.70	\$2 263.1
2	-\$527.70	100	59.00%	0.919	\$1 440.30	-\$5.28	120	37	-\$427.86	\$1 167.81
3	-\$660.10	76	57.89%	0.904	\$1 674.40	-\$8.69	144	51	-\$388.29	\$984.94
4	\$2 919.30	90	64.44%	1.492	\$1 352.90	\$32.44	117	64	\$1 368.42	\$634.17
5	\$970.10	88	64.77%	1.13	\$2 538.70	-\$11.02	118	79	\$368.39	\$964.06
6	-\$1 628.30	79	59.49%	0.836	\$3 481.90	-\$20.61	128	93	-\$525.26	\$1 123.19
7	-\$3 101.30	78	53.96%	0.672	\$4 640.90	-\$44.88	129	107	-\$991.48	\$1 301.19
8	-\$1 521.50	87	62.07%	0.85	\$3 464.70	-\$17.49	121	128	-\$356.60	\$812.04
9	\$1 069.70	89	66.29%	1.099	\$4 722.90	\$12.02	116	142	\$225.99	\$997.80
10	-\$407.70	95	62.11%	0.967	\$5 425.30	-\$4.29	108	155	-\$78.91	\$1 050.06
11	\$4 012.30	100	69.00%	1.326	\$4 462.30	\$40.12	103	170	\$708.05	\$787.46
12	\$5 207.70	89	64.04%	1.472	\$3 644.90	\$58.51	113	184	\$849.08	\$594.28
13	\$2 570.50	97	65.98%	1.182	\$3 793.70	\$26.50	105	198	\$389.47	\$574.80
14	\$1 646.50	92	65.22%	1.114	\$3 145.30	\$17.90	109	212	\$233.00	\$445.09
15	\$2 570.50	97	65.98%	1.182	\$3 793.70	\$26.50	105	175	\$440.66	\$650.35
16	\$5 713.70	114	62.28%	1.35	\$3 307.70	\$50.12	95	265	\$646.83	\$374.46
17	\$4 326.70	104	66.35%	1.256	\$3 887.50	\$41.60	99	265	\$489.82	\$440.09
18	\$7 257.90	111	64.86%	1.421	\$4 657.30	\$65.39	98	295	\$738.09	\$473.62
19	\$3 215.10	103	63.11%	1.172	\$5 333.50	\$31.21	100	295	\$326.96	\$542.39
20	\$9 707.10	108	66.67%	1.573	\$5 208.30	\$89.88	99	324	\$898.81	\$482.25
21	\$12 321.30	110	69.09%	1.756	\$3 801.90	\$112.01	93	324	\$1 140.86	\$352.03
22	\$11 265.10	108	64.81%	1.597	\$4 417.70	\$104.31	99	356	\$949.31	\$372.28
23	\$15 231.70	114	71.93%	1.819	\$3 265.30	\$133.61	90	356	\$1 283.57	\$275.17
24	\$9 783.70	114	67.54%	1.467	\$3 673.10	\$85.82	94	387	\$758.43	\$284.74
25	\$12 875.30	110	66.36%	1.671	\$3 762.10	\$117.05	95	387	\$998.09	\$291.64
26	\$3 957.70	99	63.64%	1.167	\$5 783.90	\$39.98	107	414	\$286.79	\$419.12
27	\$15 821.10	108	67.59%	1.864	\$4 106.30	\$146.49	95	414	\$1 146.46	\$297.56
28	\$3 678.50	92	59.78%	1.17	\$3 944.30	\$39.98	109	414	\$266.56	\$285.82
29	\$7 578.50	102	62.75%	1.327	\$3 724.10	\$74.30	102	447	\$508.62	\$249.94
30	-\$1 743.10	116	58.62%	0.954	\$10 389.70	-\$15.03	113	595	-\$87.89	\$522.31

Рис. 2.22. Аналіз результатів кращого тайм-фрейму

У результаті аналізу, найкращім часовим проміжком виявились 23 хвилини. З цього моменту аналіз та робота стратегії буде побудована на графіку з тайм-фреймом 23 хвилини.

Так як основна ідея стратегії RSI індикатор, то змінювати налаштування та шукати кращі показники я буду стільки у RSI індикаторі. Додаткові індикатори VO та SI я залишу незмінними з стандартними налаштуваннями.

70	14708	115
71	11260	103
72	17803	99
73	16341	96
74	16884	94
75	12358	84
76	7000	74
77	4533	66
78	10411	60
79	10336	52
80	10042	46
81	9715	40
82	14090	36
83	12585	30
84	13032	28
85	10821	20

Рис. 2.23. Аналіз результатів налаштувань RSI перекупленості

На цьому малюнку перший стовпчик це показник перекупленості, нижче 70 розглядати не потрібно, бо це призведе до більш неточних значень у індикаторі, так як значення буде наближатися до нейтрального 50. Наступний стовпчик це прибутковість стратегії з використанням вже 23 хвилин. Останній стовпчик це кількість угод які були виконані за проміжок часу у 10 тисяч барів.

Стандарте значення для перекупленості 70, але за допомогою аналізу найкращім результатом виявилось 72. Так само можна провести аналіз з значенням перепроданості, по стандарту це 30, але кращім результатом виявилось 31. Період я залишила незмінним це 14. Налаштування які я буду використовувати для індикатора RSI це 72, 31, 14.

Нижче наведено вигляд тестування стратегії на різних криптовалютних монетах. За допомогою зворотного тестування, я прагнула до поступово зростаючого графіку, без сильно помітних розривів, так як це було показано на початку.

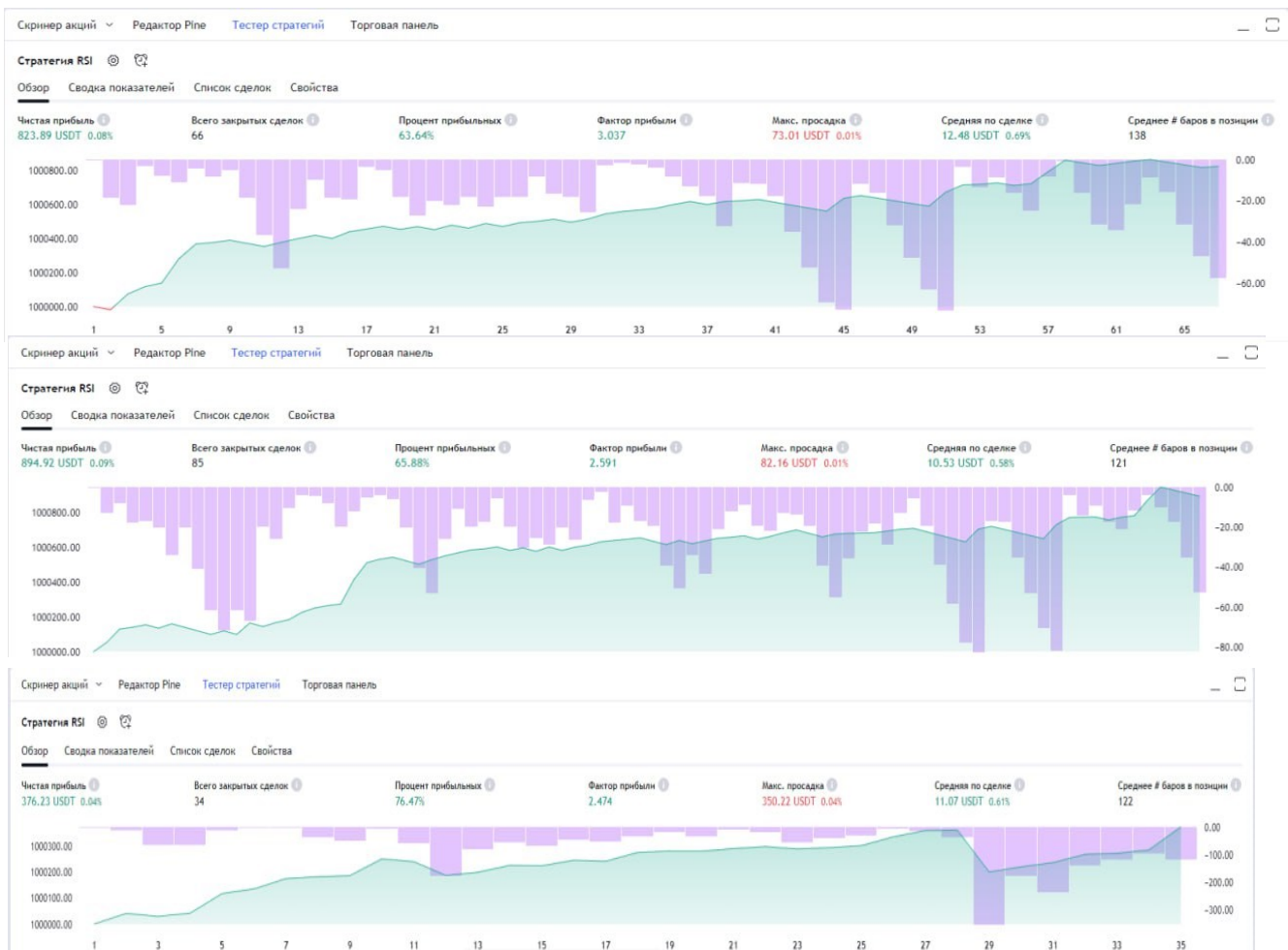


Рис. 2.24. Вигляд тестування стратегії після налаштування

За допомогою зворотного тестування було знайдено найкращі налаштування стратегії, та показано, що стратегія може бути використана для надання сигналів на здійснення угод.

Незважаючи на свою користь, важливо пам'ятати, що результати зворотного тестування не завжди є точним показником, що стратегія завжди буде прибутковою, оскільки ринкові умови постійно змінюються. В такому випадку треба розуміти, що у майбутньому потрібно також робити такі тестування для перевірки справності стратегії, чи запобігання збитків, коли результати угод будуть незадовільними тривалий час.

ВИСНОВКИ 2 РОЗДІЛУ

У другому розділі було розроблено організаційну структуру проекту, включаючи розподіл ролей у команді та розробку плану, який забезпечує ефективно і систематичне створення торгового бота.

Показано, як платформа TradingView може бути використана для розробки, тестування та впровадження стратегій торгового бота. Використання індикатора відносної сили (RSI) у контексті торгового бота.

Описаний процес розробки торгової стратегії з використанням мови програмування Pine Script, яка вбудована в TradingView. Розписаний синтаксис мови та поетапне складання стратегії.

Зроблене зворотне тестування розробленої стратегії. Тут розглядався процес оцінювання ефективності стратегії шляхом її прогону на історичних даних та підбору кращих налаштувань для стратегії.

Кожен з виконаних пунктів був важливим елементом у розробці та налаштуванні торгового бота, дозволяючи створити ефективну та надійну автоматизовану торгову систему на криптовалютному ринку.

РОЗДІЛ 3: ПІДКЛЮЧЕННЯ ТОРГОВОЇ СИСТЕМИ ДО БІРЖІ ТА УКЛАДАННЯ УГОД 3.1 СИСТЕМА СПОВІЩЕНЬ TRADINGVIEW

Система сповіщень TradingView призначена для надання негайних повідомлень при досягненні певних ринкових умов. Ці сповіщення можуть бути засновані на різних критеріях, таких як рівні цін, індикатори, стратегії або малюнки на графіках.

В своїй роботі я використовую сповіщення про стратегію. Вони пов'язані з виконанням ордерів стратегії. Коли ордер стратегії виконується, спрацьовує відповідне оповіщення.



Рис. 3.1. Іконка додавання оповіщення на стратегію

Оповіщення в TradingView створюються на серверах платформи, тобто вони працюють безперервно, незалежно від того, чи відкритий додаток користувача. Коли виконується умова оповіщення, користувач отримує сповіщення через різні канали, такі як візуальні спливаючі вікна, аудіо сигнали, електронні листи, SMS-сповіщення та PUSH-сповіщення на телефоні.

Коли умова оповіщення виконується, це слугує сигналом виконати угоду на основі заздалегідь визначеної логіки стратегії. Це може включати купівлю або продаж активів. Для цього потрібно надати доступ до торгового рахунку через API.

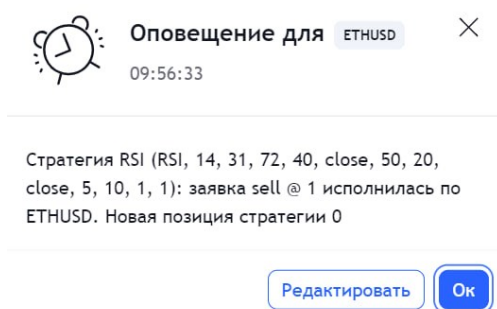


Рис. 3.2. Вигляд миттєвого оповіщення стратегії

3.2 СТВОРЕННЯ BINANCE API

Для початку необхідно мати активний обліковий запис Binance та пройти верифікацію. В налаштуваннях облікового запису є можливість створити ключ API.

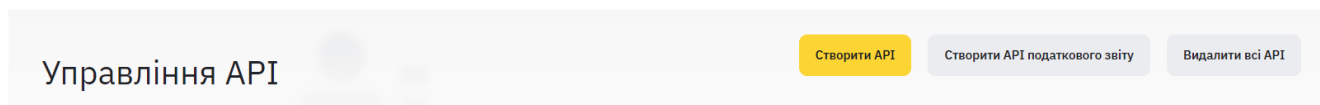


Рис. 3.3. Меню створення API на Binance

У Binance на всі важливі кроки потрібно пройти двофакторну автентифікацію, за допомогою пошти та застосунка Authenticator.

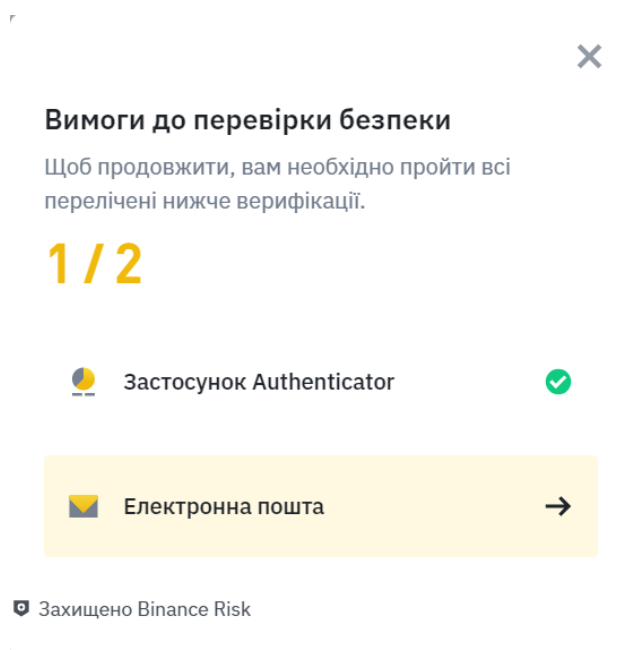


Рис. 3.4. Двофакторна автентифікація на Binance

Цей процес включає в себе присвоєння імені API для легкої ідентифікації та встановлення певних дозволів, наприклад, дозволу на торгівлю або зняття коштів, залежно від цільового використання.

Дуже важливо обережно керувати дозволами API, особливо для автоматизованої торгівлі, щоб забезпечити безпеку і контроль над рахунком.

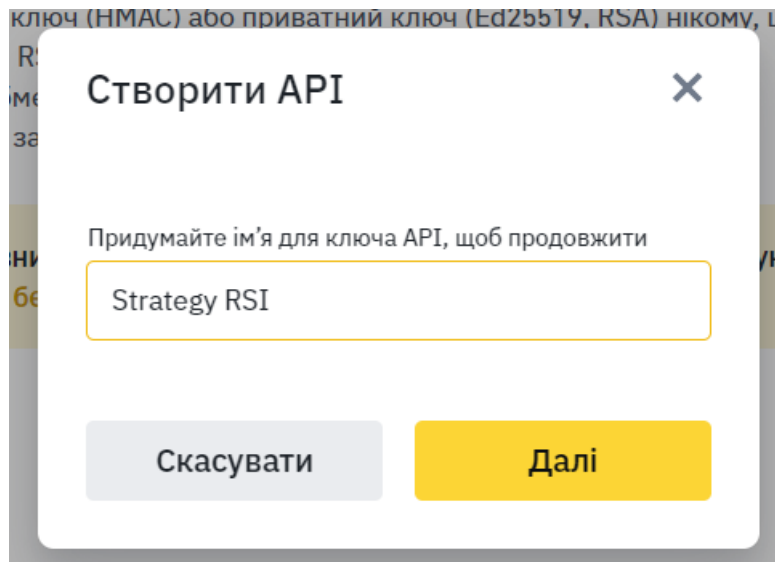


Рис. 3.5. Присвоєння імені API на Binance

При створенні облікового запису надається ключ API і секретний ключ. Ці ключі необхідні для аутентифікації запитів до API і повинні надійно зберігатися, оскільки вони надають прямий доступ до торгового рахунку. Секретний ключ можна побачити лише один раз при створенні API, тому його треба зберегти одразу. Також потрібно дозволити доступ лише до свого ПК, на якому і буде будуватися зв'язок.

Створений API має такий вигляд. Звичайно, для безпеки частина ключа була зашифрована.

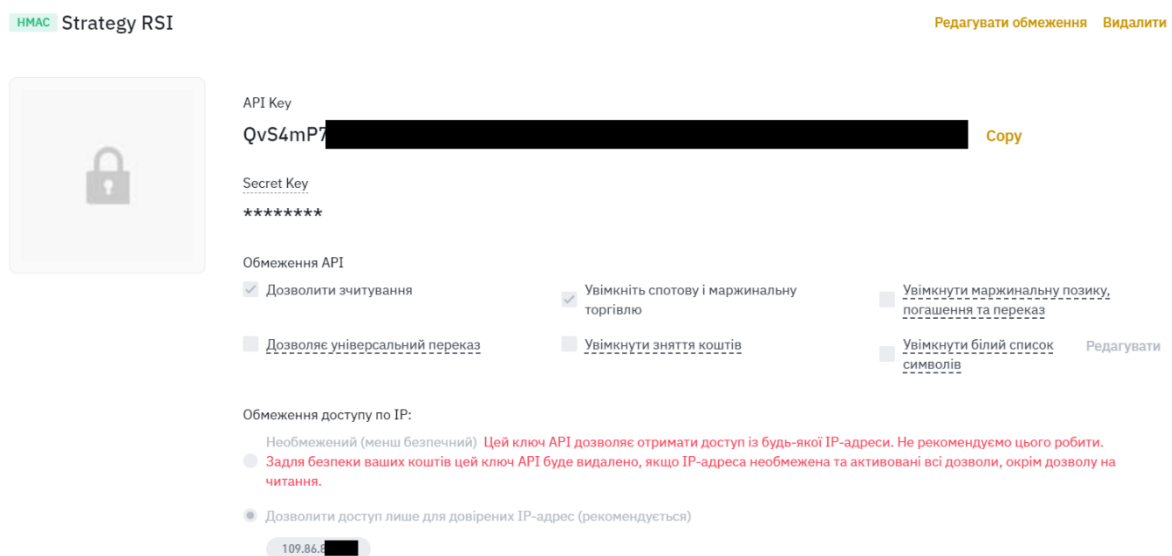


Рис. 3.6. Створений API Key стратегії на Binance

3.3 ПІДКЛЮЧЕННЯ СИГНАЛУ У FINANDY

Finandy разом з TradingView пропонує метод автоматизації торгових стратегій за допомогою використання URL-адрес веб-хуків. Ця система дозволяє отримувати сигнали на купівлю і продаж безпосередньо з TradingView, які потім можуть бути використані для автоматичного виконання угод на платформі Finandy або на біржі Binance в моєму випадку. Процес включає в себе окремі етапи як в середовищі Finandy, так і в середовищі TradingView, що забезпечує безперебійне та ефективне налаштування.

В налаштуваннях Finandy початковим кроком є створення сигнального з'єднання за допомогою веб-хука. Це з'єднання призначене для обробки сигналів на купівлю і продаж з TradingView, що робить його універсальним інструментом для різних торгових стратегій. Після встановлення цього з'єднання потрібно скопіювати надану URL-адресу і сигнальне повідомлення від Finandy. Ця інформація має вирішальне значення для зв'язування сигналів TradingView з платформою Finandy, що дозволяє автоматизувати прийом і обробку цих сигналів.

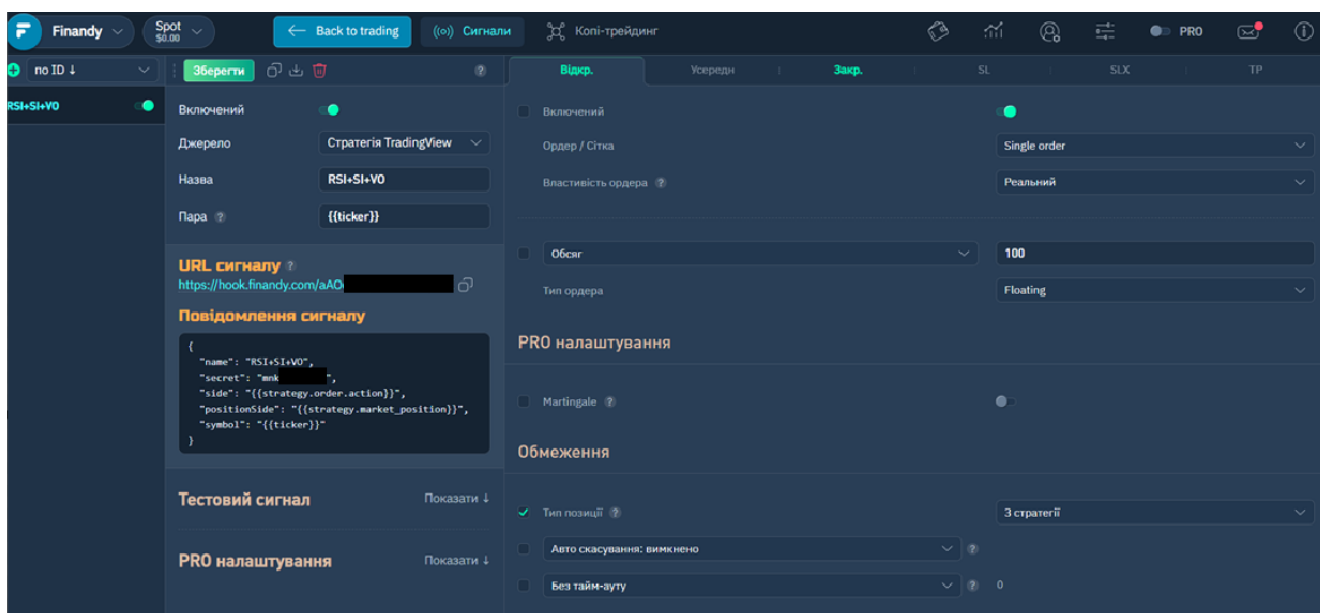


Рис. 3.7. Створення сигнального з'єднання у Finandy

Конфіденційну інформацію, така як URL-адреса та засекречена частина оповіщення сигналу, було скрито. Для забезпечення безпеки даних, але для прикладу, що адреси такі ж самі, я залишила першу частину.

Налаштування TradingView вимагає декількох ключових дій. Важливо звернути увагу на те, що кожен сигнал повинен бути прив'язаний до певного таймфрейму, наприклад, до 23-хвилинного графіка валютної пари. Ця специфіка має вирішальне значення для точності торгових сигналів.

Щоб створити сигнал на купівлю в TradingView, необхідно вибрати відповідний графічний індикатор і умову для сигналу на купівлю. Після присвоєння сигналу імені, наступним кроком є вставка повідомлення, скопійованого з налаштувань Finandy, в TradingView. Цей процес повторюється для сигналів на продаж, з додатковим кроком модифікації повідомлення, щоб змінити параметр "buy" на "sell".

Имя оповещения RSI_v2.0

Сообщение

```
{
  "name": "RSI+SI+VO",
  "secret": "mnk[REDACTED]",
  "side": "{{strategy.order.action}}",
  "positionSide": "
{{strategy.market_position}}",
}
```

Вы можете использовать специальные плейсхолдеры, например, {{close}}, {{time}}, {{plot_0}}.

Отмена Сохранить

Рис. 3.8. Активация оповіщення з Finandy

Це оповіщення використовується тільки для укладання угод згідно з ордером який на даний момент використовується у стратегії (покупка або продаж). У ході роботи я зрозуміла, що затримка, яка буде використовуватися для встановлення, та виконання стопів є дуже великою.

Тож для реалізації стопів я буду використовувати налаштування у Finandy, які будуть автоматично проставляти стопи. В такому випадку, якщо умова виконується, то не потрібно чекати поки оповіщення з TradingView отримає Finandy.

Останній крок у TradingView - перейти на вкладку "Сповіщення", вставити URL-адресу з налаштувань Finandy і створити сигнал.

Настройки Уведомления 4

Уведомления в приложении ?
Присылает push-уведомление в мобильном приложении.

Всплывающее окно
Показывает всплывающее окно с уведомлением на сайте.

Отправить email
Отправляет уведомление на адрес электронной почты, указанный в настройках профиля.

URL вебхука ?
Отправляет POST-запрос на указанный вами URL при срабатывании оповещения.

Звуковой сигнал
Проигрывает звуковой сигнал при срабатывании оповещения.
Весёлый Однократно

Отправить email-по-SMS ?
Отправляет простое текстовое сообщение для SMS на другой адрес электронной почты.

Рис. 3.9. Підключення URL-адреси

Вигляд активного сповіщення буде мати наступним чином.

```
RSI_v2.0  
{ "name": "RSI+SI+VO", "secret": "mnk...", "side": "{{strategy.order.action}}", "positionSide": "{{strategy.market_position}}", "symbol": "{{ticker}}" }  
ETHUSDTP, • Активно • 1 окт., 06:30:01
```

Рис. 3.10. Активне сповіщення стратегії

Також, таке сповіщення можна зробити неактивним. В такому випадку не потрібно видаляти повністю сповіщення, для того щоб Finandy не отримувала данні для укладання угод.

3.4 ПІДКЛЮЧЕННЯ СТОП-ЛОСУ У FINANDY

Для управління ризиками я обрала підключення стоп-лосу у самій платформі Finandy. Додавання стопів у самій стратегії дало мені можливість перевірити їх важливість і вплив на результат. Якщо залишити оповіщення з TradingView на стоп-лос, то тільки коли ціна дійде до необхідної позначки, оповіщення буде надіслане для реалізації стопа. Такий варіант є не дуже швидким та безпечним у плані збереження капіталу, бо пройде деякий час, і ціна буде гіршою для виходу з позиції, що призведе до більших втрат.

Варіант з додаванням стопів у Finandy є кращим. Коли буде укладена угода, то автоматично буде виставлятися стоп, і не треба чекати час, поки ціна дійде до стопа у стратегії.

Щоб розпочати цей процес, необхідно спочатку встановити критерії стоп-лосу в Finandy. Це передбачає визначення умов, за яких спрацьовує стоп-лос, наприклад, певний відсоток падіння або конкретна цінова точка.

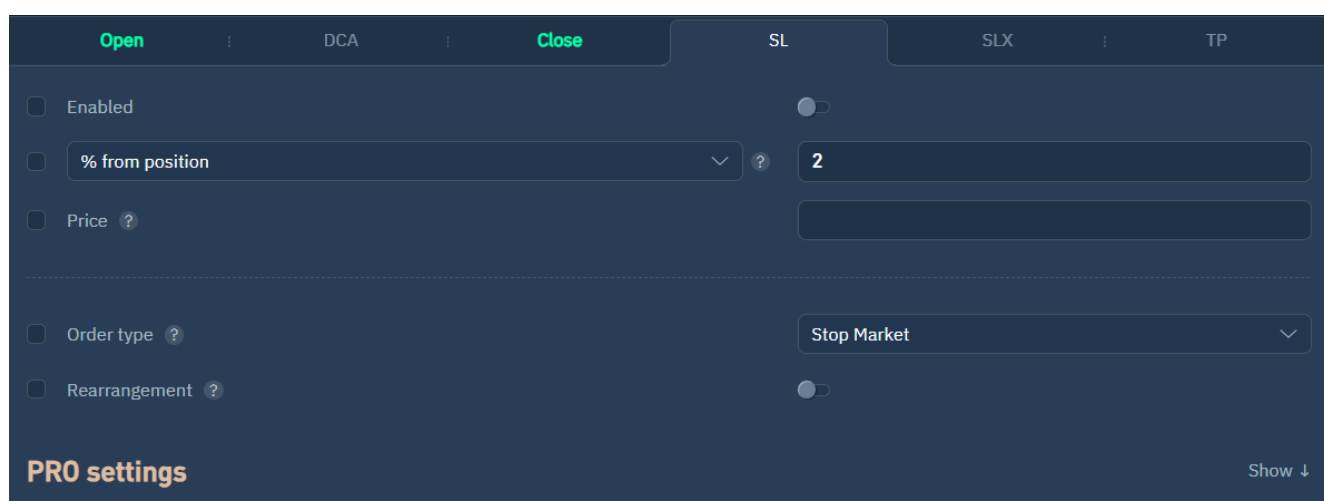


Рис. 3.11. Вікно налаштування стоп-лосу у Finandy

Після налаштування параметрів вони прив'язуються до торгового сигналу. Це гарантує, що укладання угоди на основі сигналу автоматично встановлює відповідний стоп-лосс.

Щоб ці сигнали, надсилалися до Binance, Finandy повинен бути правильно налаштований за допомогою ключів API Binance. Ці ключі повинні бути встановлені з відповідними дозволами, щоб дозволити розміщення торгових ордерів. Вкрай важливо, щоб ці ключі API управлялися безпечно, оскільки вони надають прямий доступ до торгового рахунку.

Коли угода виконується на основі визначеного сигналу стратегії, стоп-лосс одночасно надсилається на Binance за допомогою API. Виклик API містить всю необхідну інформацію для того, щоб Binance встановив ордер стоп-лосс за заздалегідь визначеною ціною спрацьовування. Якщо ця ціна досягнута, ордер стоп-лосс активується, і актив продається, щоб запобігти подальшим втратам.

Це буде працювати таким чином:

TradingView надсилає сигнал на укладання угоди у лонг (довга позиція); в цей час Finandy отримує ці дані і, за допомогою API, надсилає сигнал на покупку, і вже за ціною покупки (наприклад, ціна входу в позицію на момент укладання угоди була 1000), вираховує відсоток стопа від 1-2% в залежності від налаштувань. З цього ми отримуємо, що коли ціна активу буде 990-980, спрацьовує стоп, який вже був відправлений на Binance.

Також виникає питання: якщо стоп вже спрацював, а через деякий час надходить сповіщення з TradingView про те, що стоп-лос, встановлений позицією, закрився за стратегією, чи не буде це сповіщення слугувати сигналом, що у моєму випадку буде слугувати відкриттям вже нової позиції в іншу сторону? Ні, бо для призначення сигналів на стоп-лос потрібно додавати нову URL-адресу, і в моєму випадку Finandy опрацьовує лише оповіщення про відкриття довгої або короткої позиції, ігноруючи оповіщення про спрацювання стоп-лоссу самої стратегії.

3.5 РЕЗУЛЬТАТ УКЛАДАННЯ УГОД

Активне тестування стратегії було здійснено у вересні 2023 року. Було обрано основні криптовалютні пари: XRPUSDT.P, SUSHIUSDT.P, ETHUSDT.P, BTCUSDT.P.

Весь аналіз результативності стратегії був проведений на BTCUSDT.P, але для кожного контракту потрібні свої налаштування для кращого результату.

Наприклад, для ETHUSDT.P на якому була більшість кількості угод, були знайдені найкращі параметри такі як:

RSI settings: RSI Length=14. Oversold zone=32. Overbought zone=68. RSI safe zone=38.

SI settings: Convergence factor=125. Squeeze index length=13.

VO settings: VO fast MA=3. VO slow MA=10.

Risk management: Stop loss (%)=1.89. Take profit=3.29.

Time-frame: 9 хвилин.

За допомогою таких параметрів була виконана угода наведена нижче



Рис. 3.12. Графічне зображення угоди за допомогою стратегії RS

ETHUSDT Безстрокові Крос Лонг Закрито	Відкрито 09/07/2023 09:03:05 Закрито 09/07/2023 17:27:02	Ціна входу 1,625.34 USD Серед. ціна закриття 1,646.56 USD	Макс. відкритий інтерес 0.184 ETH Обсяг закритих угод 0.184 ETH	PnL після закриття 3.90 USD
---	---	--	--	-----------------------------

Рис. 3.13. Реалізація угоди на Binance

Угоди на інших криптовалютних парах відбувалися таким чином, як на ETHUSDT.P. Результати є прибуткові і збиткові, але завдяки стоп-лосу, вони були мінімізовані.

XRPUSDT Безстрокові Ізольована Шорт Закрито	Відкрито 09/03/2023 02:40:03 Закрито 09/03/2023 03:10:05	Ціна входу 0.5041 USD Серед. ціна закриття 0.5041 USD	Макс. відкритий інтерес 198.2 XRP Обсяг закритих угод 198.2 XRP	PnL після закриття 0.00 USD
SUSHIUSDT Безстрокові Ізольована Шорт Закрито	Відкрито 09/03/2023 02:50:09 Закрито 09/03/2023 03:00:06	Ціна входу 0.5705 USD Серед. ціна закриття 0.5709 USD	Макс. відкритий інтерес 175 SUSHI Обсяг закритих угод 175 SUSHI	PnL після закриття -0.07 USD
SUSHIUSDT Безстрокові Ізольована Шорт Закрито	Відкрито 09/03/2023 02:20:14 Закрито 09/03/2023 02:40:02	Ціна входу 0.5693 USD Серед. ціна закриття 0.5709 USD	Макс. відкритий інтерес 175 SUSHI Обсяг закритих угод 175 SUSHI	PnL після закриття -0.28 USD
SUSHIUSDT Безстрокові Ізольована Лонг Закрито	Відкрито 09/04/2023 04:10:05 Закрито 09/04/2023 06:50:09	Ціна входу 0.5710 USD Серед. ціна закриття 0.5699 USD	Макс. відкритий інтерес 175 SUSHI Обсяг закритих угод 175 SUSHI	PnL після закриття -0.20 USD
BTCSUSD Безстрокові Ізольована Лонг Закрито	Відкрито 09/03/2023 18:10:04 Закрито 09/03/2023 20:10:02	Ціна входу 25,974.40 USD Серед. ціна закриття 25,930.30 USD	Макс. відкритий інтерес 0.007 BTC Обсяг закритих угод 0.007 BTC	PnL після закриття -0.31 USD

Рис. 3.14. Збиткові угоди на Binance

ETHUSDT Безстрокові Крос Лонг Закрито	Відкрито 09/07/2023 09:03:05 Закрито 09/07/2023 17:27:02	Ціна входу 1,625.34 USD Серед. ціна закриття 1,646.56 USD	Макс. відкритий інтерес 0.184 ETH Обсяг закритих угод 0.184 ETH	PnL після закриття 3.90 USD
ETHUSDT Безстрокові Крос Шорт Закрито	Відкрито 09/04/2023 14:18:02 Закрито 09/07/2023 09:03:05	Ціна входу 1,636.92 USD Серед. ціна закриття 1,625.34 USD	Макс. відкритий інтерес 0.183 ETH Обсяг закритих угод 0.183 ETH	PnL після закриття 2.11 USD
ETHUSDT Безстрокові Крос Лонг Закрито	Відкрито 09/04/2023 13:15:02 Закрито 09/06/2023 14:18:02	Ціна входу 1,615.01 USD Серед. ціна закриття 1,636.92 USD	Макс. відкритий інтерес 0.185 ETH Обсяг закритих угод 0.185 ETH	PnL після закриття 4.05 USD
ETHUSDT Безстрокові Крос Лонг Закрито	Відкрито 09/03/2023 13:06:05 Закрито 09/03/2023 17:00:03	Ціна входу 1,628.77 USD Серед. ціна закриття 1,642.67 USD	Макс. відкритий інтерес 0.184 ETH Обсяг закритих угод 0.184 ETH	PnL після закриття 2.55 USD
ETHUSDT Безстрокові Крос Шорт Закрито	Відкрито 09/02/2023 11:36:07 Закрито 09/03/2023 13:06:05	Ціна входу 1,640.88 USD Серед. ціна закриття 1,628.77 USD	Макс. відкритий інтерес 0.182 ETH Обсяг закритих угод 0.182 ETH	PnL після закриття 2.20 USD
SUSHIUSDT Безстрокові Ізольована Шорт Закрито	Відкрито 09/03/2023 03:10:04 Закрито 09/03/2023 03:20:02	Ціна входу 0.5709 USD Серед. ціна закриття 0.5702 USD	Макс. відкритий інтерес 175 SUSHI Обсяг закритих угод 175 SUSHI	PnL після закриття 0.12 USD

Рис. 3.15. Прибуткові угоди на Binance

З отриманих результатів, можна побачити, що автоматизована торгова система без втручання виконала певну кількість угод. Головне завдання, яке було поставлене у роботі було виконане. Це демонструє високу ефективність системи в аналізі ринкових умов та самостійному ухваленні рішень про купівлю чи продаж. Автоматизований підхід до торгівлі виявився корисним у мінімізації емоційного фактора та помилок, що зазвичай впливають на процес торгівлі.

Застосування стратегії дало змогу оптимізувати процес торгівлі та підвищити його прозорість. Результати показали, що система здатна ефективно реагувати на швидкі зміни ринкової ситуації, адаптуючись до різних ринкових умов. Особливо цінним виявилось впровадження механізмів управління ризиками, зокрема автоматичне встановлення стоп-лосів, що значно знижує потенційні втрати.

Водночас, слід зазначити, що успіх автоматизованої системи залежить від постійного аналізу та оновлення алгоритмів. Необхідно регулярно проводити зворотне тестування та адаптацію стратегій з урахуванням нових даних та ринкових тенденцій. Це дозволить системі залишатися актуальною та ефективною в довгостроковій перспективі.

Слід зауважити, що така система краще працює на одній криптовалютній парі. TradingView дає можливість користуватися стратегією тільки з одними налаштуваннями, не зважаючи на різні монети. У такій ситуації можливо заплутатися та підключити криптовалютну пару до налаштувань іншої, що призведе до збитків. Більшість інформації про налаштування стратегії до певної монети губиться, якщо її не записувати. Тому я обрала тільки одну криптовалютну пару ETHUSDT.P для якої і були збережені всі налаштування для подальшої роботи, саме вона і виявилась найрезультативнішою.

Саме тому вибір конкретної криптовалютної пари дозволив зосередитися на ретельному вивченні та оптимізації стратегії саме під цю пару. Це дало змогу глибше аналізувати ринкові зміни, особливості ціноутворення та поведінку цієї

криптовалюти. Завдяки такому підходу, стратегія була адаптована під специфічні характеристики та волатильність ETHUSDT.P.

Крім того, зосередження уваги на одній парі дозволило уникнути плутанини та помилок, пов'язаних з одночасним управлінням кількома криптовалюотними парами. Це знизило ризики невдалого входу у позицію, забезпечуючи більш стабільні та передбачувані результати

У підсумку, використання автоматизованої торгової системи на базі однієї криптовалюотної пари виявилось ефективним рішенням, яке дозволило не тільки оптимізувати торгову стратегію, але й забезпечити більшу надійність та безпеку торгових операцій. Цей досвід відкриває перспективи для подальших досліджень та вдосконалення автоматизованих торгових систем в окремих криптовалюотних парах.

ВИСНОВКИ З РОЗДІЛУ

У третьому розділі описується, як використовувати систему сповіщень TradingView для отримання вчасних повідомлень про умови, які задовольняють визначені параметри торгової стратегії.

Було показано процес створення API ключа на платформі Binance, для дозволу торговій системі взаємодіяти з біржею, виконуючи торгові операції. Підключення сигналу у Finandy, за допомогою якого оповіщення про стратегію у TradingView, стає повноцінним сигналом для укладання угоди і відправки даних на Binance.

Також за допомогою Finandy було вирішено проблему з швидким виставлянням стоп-лосу. Завдяки підключенню стопів у Finandy, вони автоматично проставляються при укладанні угоди, що зменшує час на вихід з позиції при невдалій угоді.

Був проведений аналіз результатів укладання угод автоматизованою торговою системою, з якого можна виділити наступні речі:

- Автоматизована торгова система була налаштована та підключена правильно.
- Використання стоп-лосу у Finandy зменшило збитки при невдалих угодах.
- Було виявлено найкращу криптовалютну пару для стратегії та збереження результатів для неї.

ВИСНОВОК

Актуальність теми проекту підкреслена стрімким розвитком криптовалютного ринку та зростаючим попитом на автоматизовані торгові системи. Розробка власного торгового бота відкрила нові перспективи для автономної торгівлі.

Протягом виконання роботи було досягнуто ключових цілей, зокрема, була розроблена та протестована торгова стратегія, що базується на поєднанні технічних індикаторів RSI, SI та VO. Бек-тестування стратегії виявило її ефективність та потенціал для реалізації у реальних торгових умовах. Підключення стратегії до API біржі Binance через фінансовий сервіс Finandy забезпечила можливість проводити автоматичні торгові операції, що згодом підтвердилося успішними угодами без потреби втручання з боку людини.

У ході роботи була оптимізована та вирішена проблема швидкого використання стоп-лосу. Використання оповіщень про стопи у TradingView було недоліком для швидкого використання та зменшення збитків, тому було вирішено проблему за допомогою виставляння автоматичних стопів у Finandy під час укладання угоди. Такий спосіб є більш безпечним та ефективнішим.

Також було оптимізовано та налаштовано стратегію під конкретну криптовалютну пару. Такий підхід забезпечив більшу кількість позитивних угод та кращий аналіз характеру поведінки монети.

Слід зауважити, що параметри стратегії та криптовалютну пару можна змінювати, що робить цю стратегію універсальною для використання.

Робота не тільки показує процес розробки та оптимізації торгового бота, але й спонукає до подальшого дослідження в сфері автоматизації торгових систем на криптовалютному ринку. Зокрема, залишається відкритим питання пошуку нових методів та підходів у розробці автоматизованих систем, що могли б ще більше підвищити ефективність торгівлі та зменшити ризики.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Binance. [Електронний ресурс] – Режим доступу: <https://www.binance.com/uk-UA>
2. David L.C. Handbook of Digital Currency: Bitcoin, Innovation, Financial Instruments, and Bid Data/ David L.C. – San Diego: Elsevier Inc., 2015. – 588 с.
3. Delfabbro P. The psychology of cryptocurrency trading: risk and protective factors. / Delfabbro P., King D., Williams J. Journal of behavioral addictions, 2021. - 201-207 с.
4. Demiralay S. On the dynamic equicorrelations in cryptocurrency market/ Demiralay S., Golitsis P. The quarterly review of economics and finance, 2021. - 524- 533 с.
5. Donald MacKenzie. Trading at the Speed of Light/ Donald MacKenzie, 2023. - 49 с.
6. Finandy. [Електронний ресурс] – Режим доступу: <https://finandy.com/uk/panel>
7. Kim D. Are suspicious activity reporting requirements for cryptocurrency exchanges effective?/ Kim D. Financial innovation, 2021. - 1-17 с.
8. Ratford S. Algorithmic Trading Strategies/ Ratford S., 2022. - 24 с.
9. Thomas H. All-in On AI/ Thomas H., Mittal N., 2023. - 76 с.
10. TradingView. [Електронний ресурс] – Режим доступу: <https://ru.tradingview.com/>
11. TradingView. Pine Script Language Reference Manual [Електронний ресурс] // TradingView Pine Script Documentation, версія 5. - Режим доступу: <https://ru.tradingview.com/pine-script-reference/v5/>
12. Батюк Л., Кравченко О. Криптовалюта: механізм функціонування та перспективи поширення в умовах глобалізації. Інноваційна економіка/ Л. Батюк, О. Кравченко, 2021. - 131-139 с.

- 13.Бизкровний О. Дослідження ціноутворення криптовалюти та ефективність використання машинного навчання для пошуку тренду вартості / О. Бизкровний, 2022. – 115 с.
- 14.Валєєв Р. Мистецтво трейдингу. Практичні рекомендації для трейдерів з досвідом/ Р. Валєєв. - Видання: Манн, Иванов и Фербер, 2018. - 198 с
- 15.Варенко В.М. Основи аналітики: навчальний посібник/ В.М. Варенко Ліра-К, 2022.- 98 с
- 16.Галушка Є. Сутність криптовалют та перспективи їх розвитку. Молодий вчений/ Є. Галушка, О. Пакон, 2017. - 634-638 с.
- 17.Гонак І. Криптовалюта як об'єкт інвестування. Scientific notes of ostroh academy national university," economics" series/ І. Гонак, 2022. - 71-84 с.
- 18.Гонак І. Теоретичне обґрунтування визначення криптовалюти як грошей. Інноваційна економіка / І. Гонак, 2021. - 123-130 с.
- 19.Дж. О'Брайен. Финансовый анализ и торговля ценными бумагами/ Дж. О'Брайен, С. Шривастава, 2014. 127 с.
- 20.Кравець Д. Теоретичні та практичні аспекти ролі криптовалюти як елементу фінансових активів : науковий вісник/ Д. Кравець, 2022. – 60 с.
- 21.Проект Закону «Про внесення змін до Податкового Кодексу України щодо оподаткування операцій з віртуальними операціями в Україні» від 27.10.2018 №9083-1.
- 22.Проект Закону про обіг криптовалют в Україні [Електронний ресурс] // Офіційний веб-портал Верховної ради України [Інтернет-портал], 2017. – Режим доступу: http://w1.c1.rada.gov.ua/pls/zweb2/webproc4_1?pf3511=62684
- 23.Трегуб В.Г. Проектування систем автоматизації навчальний: посібник/ В.Г. Трегуб. Ліра-К, 2019. - 154 с
24. Хосп Дж. Про криптовалюту просто. Біткойни, ефіріум, блокчейн, децентралізація, майнинг, ICO & Co/ Дж. Хосп. - Видання: FinanzBuch Verlag, 2017. - 310 с.
- 25.Штонда Р. М. Соціальні мережі в інтернеті як інструмент загрози національній системі кібербезпеки України. Актуальні проблеми управління

інформаційною безпекою держави/ Р. М.Штонда, Н. А. Паламарчук, С.М. Островський, 2018. - 190–192 с.

ДОДАТКИ

Повний код стратегії

```
strategy("RSI Strategy", overlay=true)

//Inputs
string conftype = input.string("All","Entry Method", options = ["RSI", "RSI+SI", "RSI+VO",
"All"], group = "Position Initiation Settings")
length = input(14, title = "RSI Lenght", group = "RSI Settings")
overSold = input.float(30, minval = 0, maxval = 100, title = "Oversold Zone", group = "RSI
Settings")
overBought = input.float(70, minval = 0, maxval = 100, title = "Overbought Zone", group =
"RSI Settings")
RSI_Safe_Zone = input(40, title = "RSI Safe Zone", group = "RSI Settings")
price = input(close, title = "RSI Source", group = "RSI Settings")
sq_conv = input(50, title = "Convergence Factor", group = "Squeeze Index Settings")
sq_length = input.int(20, minval = 1, title = "Squeeze Index Lenght", group = "Squeeze
Index Settings")
sq_src = input(close, title = "Squeeze Index Source", group = "Squeeze Index Settings")
shortlen = input.int(5, minval=1, title = "Volume Oscillator Fast MA", group = "Volume
Oscillator Settings")
longlen = input.int(10, minval=1, title = "Volume Oscillator Slow MA", group = "Volume
Oscillator Settings")
StopLoss = input.float(1, minval = 0.01, title = "Stop Loss (%)", group = "Risk
Managment") / 100
profit = input.float(1, minval = 0.01, title = "Take Profit (%)", group = "Risk
Managment") / 100

//Style
col_0 = input(#ffeb3b, 'Gradient'
, inline = 'inline0'
, group = 'Style')

col_1 = input(#ff5d00, ''
, inline = 'inline0'
, group = 'Style')

col_2 = input(#ff1100, ''
, inline = 'inline0'
, group = 'Style')

//Squeeze index
var max = 0.
var min = 0.
max := nz(math.max(sq_src, max - (max - sq_src) / sq_conv), sq_src)
min := nz(math.min(sq_src, min + (sq_src - min) / sq_conv), sq_src)
diff = math.log(max - min)
```

```

psi = -50 * ta.correlation(diff, bar_index, sq_length) + 50

//Volume Oscillator
var cumVol = 0.
cumVol += nz(volume)
if barstate.islast and cumVol == 0
    runtime.error("No volume is provided by the data vendor.")
short = ta.ema(volume, shortlen)
long = ta.ema(volume, longlen)
osc = 100 * (short - long) / long

//Trend Protection
curhi = request.security(syminfo.tickerid, timeframe = "1D", expression = high)
prehi = request.security(syminfo.tickerid, timeframe = "1D", expression = high[1])
curlo = request.security(syminfo.tickerid, timeframe = "1D", expression = low)
prelo = request.security(syminfo.tickerid, timeframe = "1D", expression = low[1])
datr = request.security(syminfo.tickerid, timeframe = "1D", expression = ta.atr(14))

lol = curhi > prehi and ((prehi - prelo) > datr) or ((curhi - curlo) > datr)
sol = curlo < prelo and ((prehi - prelo) > datr) or ((curhi - curlo) > datr)
lolo = not lol
solo = not sol

//Functions
vrsi = ta.rsi(price, length)
co = ta.crossover(vrsi, overSold) and vrsi <= 0 + RSI_Safe_Zone
cu = ta.crossunder(vrsi, overBought) and vrsi >= 100 - RSI_Safe_Zone
sc = psi < 80
voc = osc > 0

//Risks
lstop = strategy.position_avg_price * (1 - StopLoss)
sstop = strategy.position_avg_price * (1 + StopLoss)

//Take Profits
lprofit = strategy.position_avg_price * (1 + profit)
sprprofit = strategy.position_avg_price * (1 - profit)

//Position Initiation Method
bool is_long = switch conftype
    "RSI" => co
    "RSI+SI" => co and sc
    "RSI+VO" => co and voc
    "All" => co and sc and voc
bool is_short = switch conftype
    "RSI" => cu
    "RSI+SI" => cu and sc
    "RSI+VO" => cu and voc
    "All" => cu and sc and voc

//Executions
if (not na(vrsi))
    if is_long and (lol or sol)

```

```

    strategy.entry(id = "L", direction = strategy.long, comment = "L")
  if is_short and (lol or sol)
    strategy.entry(id = "S", direction = strategy.short, comment = "S")

//Stops and Takes
if strategy.position_size > 0
  strategy.exit(id = "LX", stop = lstop, limit = lprofit)
if strategy.position_size < 0
  strategy.exit(id = "SX", stop = sstop, limit = sprofit)

//plot(strategy.equity, title="equity", color=color.red, linewidth=2,
style=plot.style_areabr)

//Plots
css1 = color.from_gradient(psi, 0, 80, col_0, col_1)
css2 = color.from_gradient(psi, 80, 100, css1, col_2)
plot_0 = plot(psi, 'PSI', psi > 80 ? na : css2)
plot(psi, 'Dots', psi > 80 ? css2 : na, style = plot.style_cross)

plot_1 = plot(80, display = display.none, editable = false)

fill(plot_0, plot_1, psi < 80 ? na : color.new(#ff1100, 80))

plot(vrsi, color = color.purple, display = display.pane)
//
hline(50, color = #787B86, title="Zero")
hline(80)
hb = hline(overBought, color = color.red, linewidth = 1)
hs = hline(overSold, color = color.green, linewidth = 1)
fill(hb, hs, color = color.rgb(65, 89, 156, 77))

```