Project

COMP5568
Winter 2023

Project TAs:

Junzhe Jiang junzhe.jiang@connect.polyu.hk

Zhi Yuan Sun zhi-yuan.sun@connect.polyu.hk

General Requirements

- Form a team of 4-5 members
- Presentations on April 13 - **40% of project grade**
  - Reminder: the project is worth 30% of the total course grade
- Final report submission deadline April 15 - **60% of project grade**
- Documentation on development environment file will be provided to you
- 3 topics to choose from:
  1. Survey of NFTs across chains (3 groups max, FCFS)
  2. Implementation of a coin swap on an Ethereum testnet
  3. Implementation of marketplaces and arbitrage opportunity detection tool

Survey of NFTs across chains

- NFT standards, implementations, and features vary across blockchains
- Investigate:
  - how NFTs are supported on Ethereum, Solana, Polygon, and Chia
  - how NFT trading is done on those blockchains (custodial?)
- Compare options and requirements for content storage (on chain or off chain?)
- Compare the user experience in terms of wallet support
  - does Metamask support NFTs? How?
  - what other major wallets support them?
- 20 pages (excluding references)

Implementation of a coin swap on an Ethereum testnet

- Build and deploy two ERC20 tokens
- Build a coin swap protocol, with two liquidity pools following two different market making strategies:
  - Constant Product Automated Market Making (AMM)
  - Curve-like AMM
- Liquidity pools (LPs) should provide:
  - swap() function to swap two tokens
  - ability to add liquidity into the pool, in exchange for newly minted LP tokens
- Build a web3 interface that connects to the user's wallet (in Javascript)
  - Let user choose LP
  - Select coins to swap
  - See current (estimated) exchange rate based on blockchain status
  - Enter amount to exchange
  - Perform the exchange
- 8-10 pages (exclude references)
- Code submission

Implementation of marketplaces and arbitrage opportunity detection tool

- /!\ For more advanced students
- Task 1: Your own marketplaces
  - **Build** two decentralized marketplaces for ETH vs. a token
  - **Oracles** sending rate updates to the chain every 30 seconds
  - **Random price fluctuation** (deterministic starting with fixed seed for reproducibility) between a certain range (e.g., ETH->USDC, between 1000-1010)
  - Predefined liquidity
  - Implement **buy(), sell()** functions
  - Build a client that **monitors price updates** from the two different marketplaces, e.g., in Python (with web3py)
  - Calculate transaction fees plus platform fees with buy/sell rates, decide when to make a trade, report all fees and rates (and block #), make a single transaction that buys and sells tokens on the two marketplaces to **make an profit in ETH**

Implementation of marketplaces and arbitrage opportunity detection tool

- Task 2: Real marketplaces
  - **Track exchange rates** of various tokens on Uniswap, Sushiswap, Binance, by using their official Software Development Kit or APIs
  - Show what a transaction would look like to **buy low/sell** high across platforms
  - Optional: explore cross-chain arbitrage
- 8-10 pages report (excluded references)
- Code Submission

Grading Rubric

Report and code - 60%

- Demonstrates a well-developed focus, thorough points of development, and a logical pattern of organization of ideas and concepts.
- Supports result with images and data
- Illustrate key parts/mechanisms of the code
- Proper background and references
- Completeness, correct format, appropriate language and accuracy
- Code is well documented, runs, is not plagiarized, proper references are made
- Survey (Option 1): background, breadth and depth of coverage, systematization of knowledge (tables, figures), originality, proper references

Presentation - 40%

- Content (topic is well defined, there is meat to discuss)
- Delivery (organization and coherence, logical flow, good overview, good timing, questions are properly answered)

Groups and topics

- Send an email to the first TA about your **group composition** (one email per group) and which **topic you selected**
  - Junzhe Jiang junzhe.jiang@connect.polyu.hk
- Deadline: by next week's lecture
- Otherwise: groups will be formed with unassigned students