Principles of Cryptocurrency Design

Appunti ed Esercizi

Francesco Pasquale

23 aprile 2024

Nella lezione di oggi abbiamo implementato classi e metodi che abbiamo utilizzato per fare il *parsing* delle transazioni e per identificare le componenti da cui sono formate. Gli esercizi di questa nota si riferiscono al codice scritto in aula, che potete scaricare qui: `https://www.mat.uniroma2.it/~pasquale/dida/aa2324/pcd/pcd240423.zip`

Esercizio 1. Nel file `helpers.py` abbiamo implementato la funzione `varint2int` che prende in input uno *stream* di byte e restituisce il numero intero corrispondente, secondo le specifiche definite qui:
`https://developer.bitcoin.org/reference/transactions.html#compactsize-unsigned-integers`
Scrivere una funzione `int2varint` che esegua l'operazione inversa: prenda in input un intero non negativo minore di 2^{64} e restituisca una sequenza di byte che codifichi l'intero secondo le specifiche.

Esercizio 2. Nel file `transaction.py`, per ognuna delle classi `Tx`, `TxIn` e `TxOut` abbiamo implementato un metodo di classe `parse` che prende in input una sequenza di byte opportuna e restituisce l'oggetto codificato nella sequenza.
Per ognuna delle classi, scrivere un metodo `serialize` che restituisca una sequenza di byte contenente la serializzazione dell'oggetto della classe.

Esercizio 3. Scrivere un metodo `tot_out` per la classe `Tx` che restituisca la somma degli `amount` contenuti negli output della transazione.

Esercizio 4. Scrivere un metodo `fee` per la classe `Tx` che restituisca la *transaction fee* della transazione. La transaction fee è la differenza fra la somma degli `amount` contenuti negli output puntati dagli input della transazione e la somma degli `amount` contenuti negli output della transazione. Per recuperare gli output puntati dagli input di una transazione potete usare, per esempio, le api di un qualche sito che consente di recuperare dati dalla Blockchain, come mostrato per esempio nel file `get_tx.py`.

Esercizio 5. Progettare e implementare un metodo della classe `Tx` che restituisca in output un albero di cui l'istanza dell'oggetto è il nodo radice. I figli di un nodo/transazione u sono le transazioni i cui output sono puntati dagli input di u. Le foglie di un tale albero perciò saranno tutte transazioni *coinbase*.