

Simulátor akciové burzy



Jan Škoda | WOOD & Company Coding Challenge | 19.7.2016

Osobní představení

Jan Škoda, MFF UK – UI a Robotika

Práce

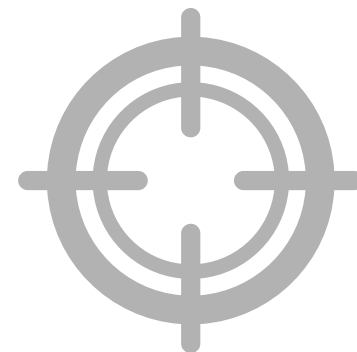
C++ a Python
programátor,
Linuxový správce,
±freelance vývojář,
startup-ista

Záliby

squash, fotografie,
vodáctví,
Japonsko/ština,
technologie, strojové
učení, Python, C++11

Cíle a funkcionalita

- Maximální efektivita
 - velký počet otevřených pokynů
 - v Pythonu
 - udržení původního protokolu
 - bez vláken
- Automatické testy a benchmarky
- Přesné neceločíselné ceny
- Ošetření a hlášení chyb klientům
- Instalační setup.py skript



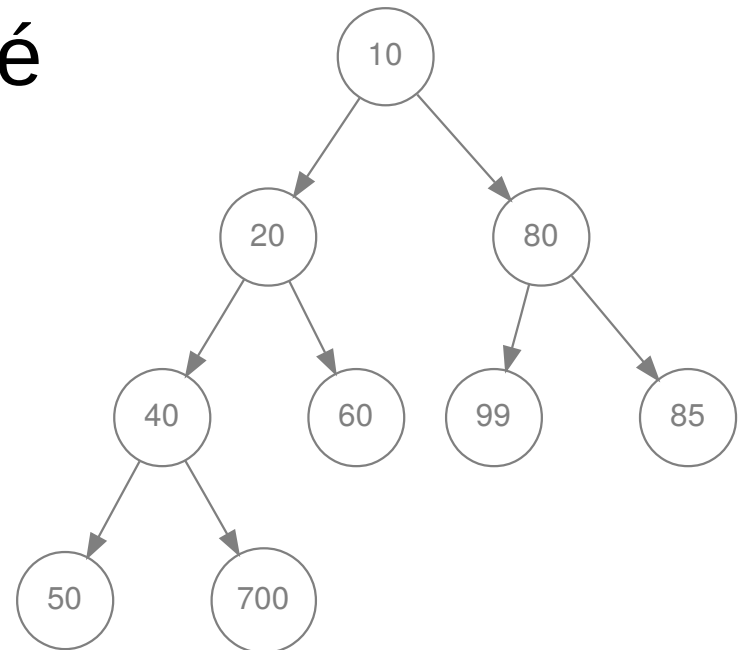
Implementace

- Python 3.5
- Sít': *asyncio*
- Book a pokyny: *heapq* a *decimal*
- Protokol: *ujson* (volitelně)
- Kód: PEP8, dokumentace, typové anotace

[illegible]

Limit Order Book

- Minimové haldy – matching porovnáním kořene
- Mazání pokynů průchodem
- Dotaz na množství podle ceny
 - index pomocí hešovací tabulky
- Ceny mohou být neceločíselné



Časová složitost / Výsledky benchmarku

- Pokyny nad LOB:

Operace	Složitost
vytvoření pokynu	$O(\log(n))$
matching	$O(\log(n))$
dotaz na množství komodity dle ceny	$O(1)$
zrušení pokynu	$O(n)$

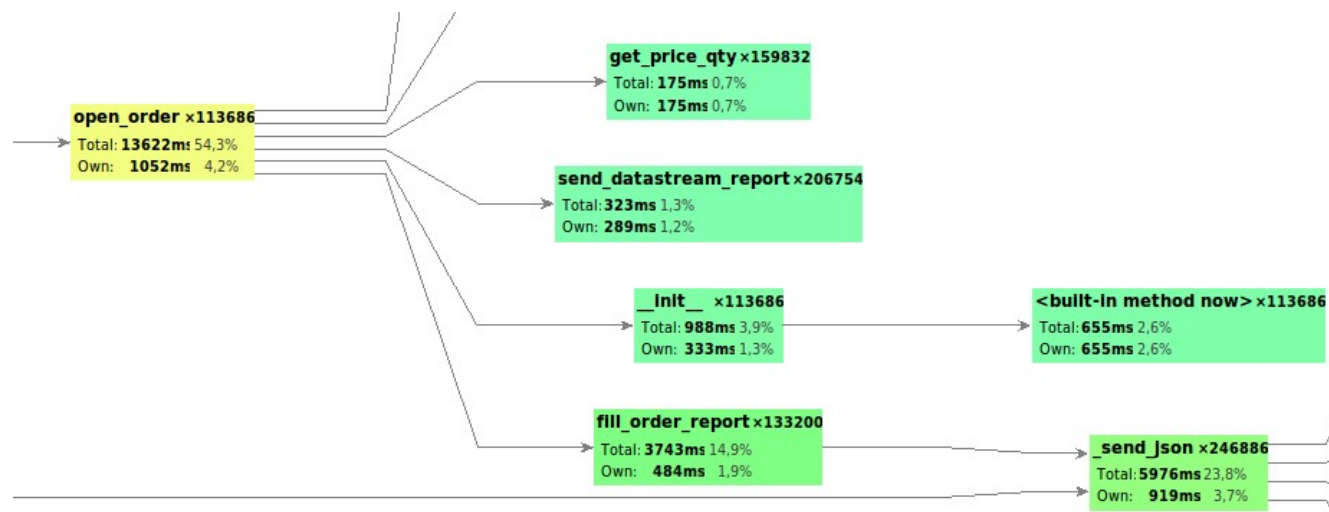
- Průměrný případ
- $n \sim$ počet pokynů

- $\sim 50\,000$ otevřených pokynů
- ceny a množství z normálního rozdělení
- pokyny otvírány přes síť pomocí JSON protokolu
- **10 000 pokynů/s**

Testy a profilování

- modul *unittest*
- testují logiku burzy a datové struktury

▼ OK Test Results	424ms
▼ OK test_server.TestServer	424ms
OK test_order_benchmark	224ms
OK test_order_network_benchmark	200ms
▼ OK test_exchange.Exchange	0ms
OK test_cancel_order	0ms
OK test_datastream_callback	0ms
OK test_decimal	0ms
OK test_fill_callback	0ms
OK test_get_clientid	0ms
OK test_open_order	0ms
OK test_sell_order_price_not_changed	0ms



Hlášení chyb

- Chyby v programu propagovány a ošetřovány pomocí výjimek
- Neošetřené chyby odeslány klientovi a vypsány na standardní chybový výstup

```
def remove_order(self, clientid: int,  
                    orderid: str):  
    (...)  
    raise KeyError("Order with specified" \  
                  "id does not exit")
```



```
Command? 2  
Order ID to cancel? 123  
  
<2016-07-18 12:31:17.504632 sending  
'{"message": "cancelOrder", "orderId": 123}'>  
  
<2016-07-18 12:31:17.510577 received  
'{"message": "error", "report": "Processing your  
request failed", "reason": "KeyError: \'Order  
with specified id does not exit\'"}'>
```


Díky za pozornost!

Jan Škoda | www.janskoda.cz | skoda@jskoda.cz

Python 3.5	Hlášení chyb	cProfile	Efektivita	PEP8	Unittest
Benchmark	Heapq	Asyncio	Decimal	setup.py	