

HUAWEI-NUS Innovation Challenge

Problem Statement Update

23 February 2023



In partnership with:



Outline

1. Tickdata
2. Formula
3. Program
4. Demo

In partnership with:



1. Tickdata : tickdata used for testing your program

New test Tickdata (100stock)

Pre-ranking

Closed-door

Key:

- 1) The format is same as history tickdata file, but only 100stocks.
- 2) The data of new test tickdata file will come from the day before the ranking day.
- 3) We will upload the new tickdata file to website the day after the ranking day.

For example: Mar 8 is ranking day, we use a new tickdata to test your program. The data of this new tickdata will come from Mar 7, and the file will be uploaded to the website on Mar 9.

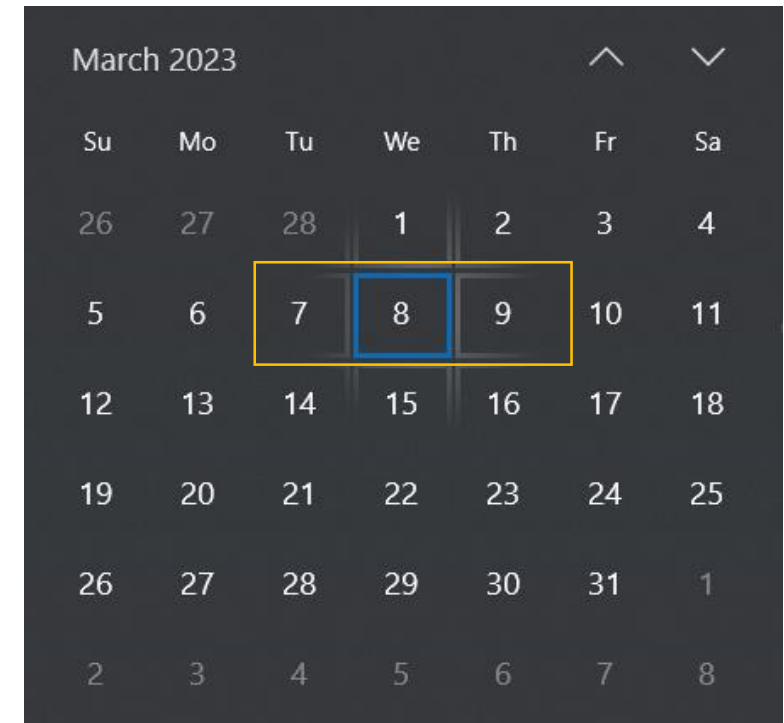
History Tickdata(500stock)

The share expires at:07/16/2023 09:31:42 GMT+08:00

Name

[tickdata](#)

[demos](#)



In partnership with:



2. Formula(updated)

The yield r of a single stock on a single trading day.

$$r = \left(\frac{VWAP - \sum_{i=1}^K p_i / K}{VWAP} \right) * SGN * 10000$$

Old

$$s = \frac{\sum_{i=1}^M r_i}{M}$$



$$r = \left(\frac{VWAP - \sum_{i=1}^K (p_i v_i) / 100}{VWAP} \right) * SGN * 10000$$

$$r.stock = r.buy + r.sell$$

New



evaluation.py

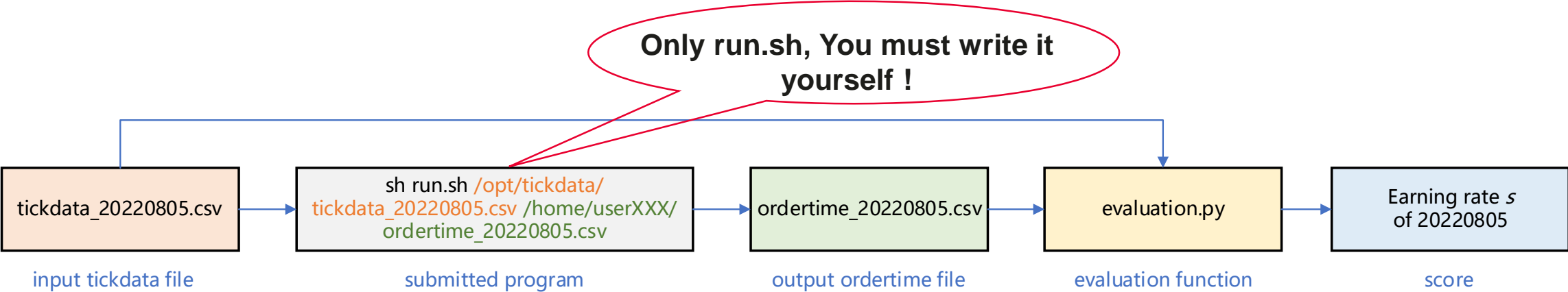
(updated on 22 Feb.)

- $VWAP$ is the **weighted** average market price of the stock on the current day
- SGN is the buy and sell tag, buying $SGN=1$, selling $SGN=-1$
- K is the number of transactions(buy or sell)
- p_i is the latest price given by the contestant at each entrustment moment
- v_i is the volume given by the contestant at each entrustment moment, total volume is 100
- M is the number of stocks in the basket

In partnership with:



3. Program 1



3. Program 2 (updated): 3-steps circle

We cannot **use future data** to make decision **on history trading**

tickdata_xxx.csv



Read all lines
first



ordertime

Table 1 Tickdata File Data Field Description (Part)

No	Field Name	Field Description
1	COLUMN01	Index
2	COLUMN02	Stock code
3	COLUMN03	Tick time (accurate to milliseconds)
4	COLUMN04	Opening price of the day

Step1: read one line from the *tickdata* file

Tickdata line, **COLUMN01** = 3

tickdata_xxx.csv



3-steps circle

run.sh

Step2: Make decision on current line

- 1) Buy in
- 2) Sell out
- 3) Do nothing

Ordertime line: { dataidx = 3 ✓ Good
dataidx < or > 3 ✗ Invalid }



ordertime.csv

Table 2 Ordertime File Data Field Description

No	Field Name	Field Description
1	symbol	Stock code
2	BSflag	Buying or Selling Instruction ('B': Buy In, 'S': Sell Out, 'N': Do nothing)
3	dataidx	Index field that the current instruction corresponding to the line of tickdata file (The value of COLUMN01)
4	volume	The stock order volume of current transaction. If the BSflag = 'N', volume set "0"

Step3: write the decision order of current line to the
ordertime file

dataidx = COLUMN01, BSflag = ('S' or 'B' or 'N')

Key:

- 1) Follow the 3-steps circle
- 2) **Read one tickdata line, must write one and only one ordertime line**
- 3) **dataidx = COLUMN01**

3. Program 3 (updated) : ordertime

Table 2 Ordertime File Data Field Description

No	Field Name	Field Description
1	symbol	Stock code
2	BSflag	Buying or Selling Instruction ('B': Buy In, 'S': Sell Out)
3	dataidx	Index field that the current instruction corresponding to the line of tickdata file (The value of COLUMN01)
4	volume	The stock order volume of current transaction

Old



No	Field Name	Field Description
1	symbol	Stock code
2	BSflag	Buying or Selling Instruction ('B': Buy In, 'S': Sell Out, 'N': Do nothing)
3	dataidx	Index field that the current instruction corresponding to the line of tickdata file (The value of COLUMN01)
4	volume	The stock order volume of current transaction If the Bsflag = 'N', volume set "0"

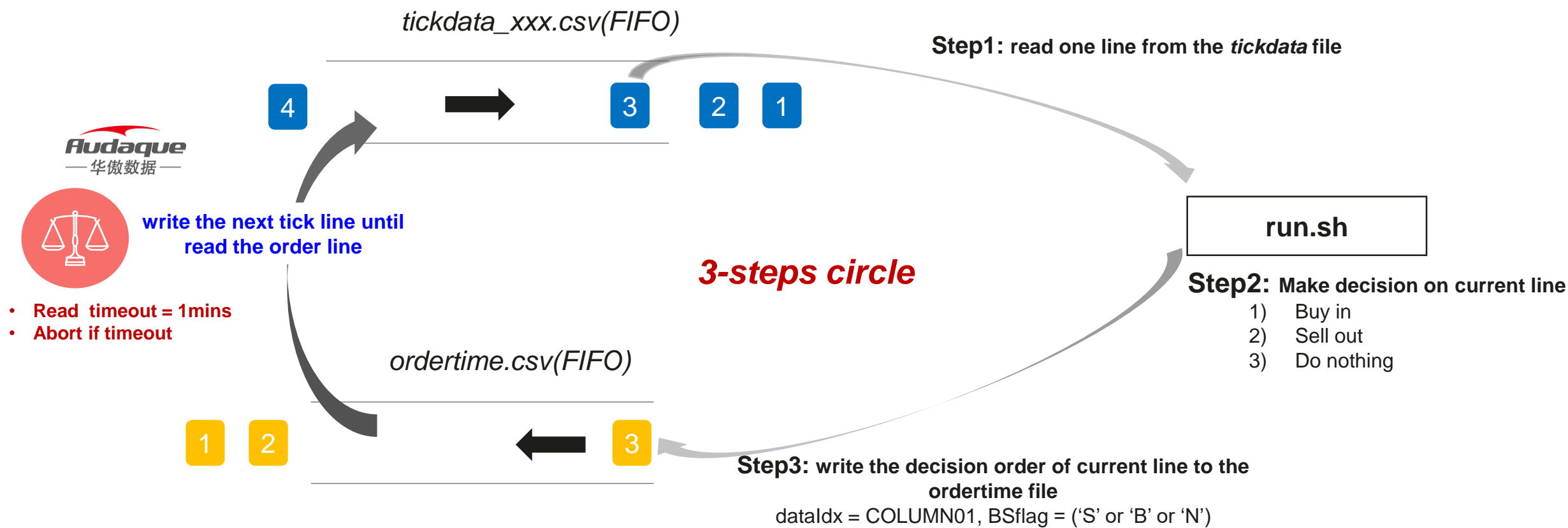
New

In partnership with:



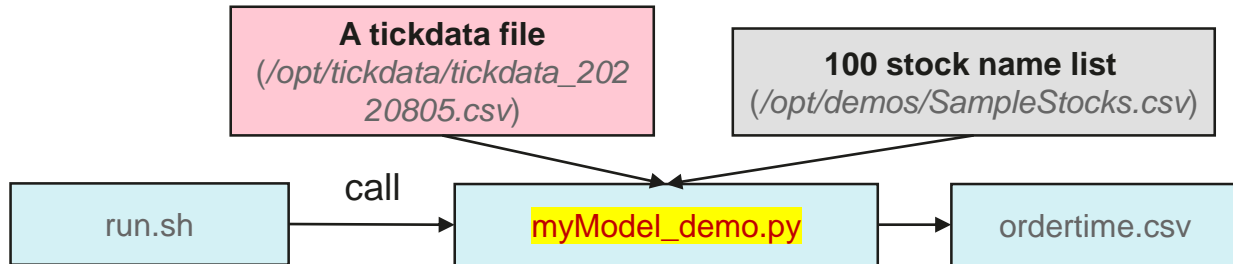
3. Program 4(updated) : use FIFO to replace FILE as interface to test your program

(you don't need to change your code if it follow 3-steps circle)



Notice: In the [demo run.sh](#) program, we have already written the input and output interfaces properly, it could support both FIFO and FILE interface, you just need copy it.

4. Demo(updated) : show demo1



**Replaced with
your strategy
code**

Notice:

The `myModel_demo.py` program would support both FIFO and FILE interface, you should replace the strategy part with your code. The demo2 is similar in this part.

```
39 # ----- Loop -----
40 #
41 # loop read all tick lines from tickdata file, do decision with your strategy
42 # write the order to the ordertime file.
43 #
44 while True:
45     # -----read one tick line-----
46     tick_line = tick_data.readline()
47     if row_str.strip() == 'stop':
48         break
49
50     if row_str:
51
52     # ----- Your Strategy code Begin-----
53     # please replace your code here
54     # tick_line is the tick data from tickdata file
55     #
56     row = tick_line.split(',')
57     sym = row[1]
58     idx = idx_dict[sym]
59     tm = int(row[2])
60
61     order_buy = 0
62     if tm < 145000000:
63         tm_rate = get_time_rate(tm)
64         if tm_rate > od_idx_buy[idx] / od_nCount_buy:
65             order_buy = od_vol_buy
66             od_idx_buy[idx] += 1
67             cum_vol_buy[idx] += od_vol_buy
68         elif target_vol - cum_vol_buy[idx] > 0: # force complete before market closes
69             order_buy = target_vol - cum_vol_buy[idx]
70             cum_vol_buy[idx] = target_vol
71
72     order_sell = 0
73     if tm < 145000000:
74         tm_rate = get_time_rate(tm)
75         if tm_rate > od_idx_sell[idx] / od_nCount_sell:
76             order_sell = od_vol_sell
77             od_idx_sell[idx] += 1
78             cum_vol_sell[idx] += od_vol_sell
79         elif target_vol - cum_vol_sell[idx] > 0: # force complete before market closes
80             order_sell = target_vol - cum_vol_sell[idx]
81             cum_vol_sell[idx] = target_vol
82
83     #
84     if order_buy > order_sell:
85         order = f'{sym},B,{row[0]},{order_buy - order_sell}'
86     elif order_buy == order_sell:
87         order = f'{sym},N,{row[0]},0'
88     else:
89         order = f'{sym},S,{row[0]},{order_sell - order_buy}'
90     # ----- Your Strategy code End -----
91
92     # -----write one order line to ordertime file-----
93     order_time.writelines(order + '\n')
94     order_time.flush()
```

Q&A (maybe affect your strategy)

Q1: Are we allowed to do **short selling**? (for example, sell 10 units of stock A before we buy it later on the same day)

A1: **Yes**, We support do short selling. (It is necessary to clarify that although the Chinese market does not support short selling, but it can actually be achieved by pre-purchasing a certain amount of shares.)

Q2: There is a description “The interval between two consecutive transactions should be no less than 1 minute” on the website, **but the demo(old) program sells and buys a stock at the same time**

A2: The consecutive transactions here refer to a continuous buying or continuous selling of a particular stock. The buy or sell decision could be separately. (In fact, the 1 minute-rule is used to avoid high commissions in the real stock market)

But, if your program **produce buy and sell order of one stock at the same tick time**, please **combine the buy and sell order to one ordertime line**. We have updated our demo program to support this new requirement.

Questions and Answers



In partnership with:



Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and
organization for a fully connected,
intelligent world.

**Copyright©2018 Huawei Technologies Co., Ltd.
All Rights Reserved.**

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

