

Project Report
On
BLACK-BOX TRADING

Submitted by:

180001012(Bhaskar)
180003044(Rishab Kumar Yadav)

Computer Science and Engineering
2nd year

Under the Guidance of

Dr. Kapil Ahuja



Department of Computer Science and Engineering
Indian Institute of Technology Indore
Spring 2020

Contents:

- Introduction
- **Objective**
- Algorithm Analysis
- Algorithm Design
- Complexity Analysis
- Implementation and Results
- References

Introduction:

Algorithmic trading (also called automated trading, black-box trading, or algo-trading) uses a computer program that follows a defined set of instructions (an algorithm) to place a trade. The trade, in theory, can generate profits at a speed and frequency that is impossible for a human trader.

The defined sets of instructions are based on timing, price, quantity, or any mathematical model. Apart from profit opportunities for the trader, algo-trading renders markets more liquid and trading more systematic by ruling out the impact of human emotions on trading activities.

Most algo-trading today is high-frequency trading (HFT), which attempts to capitalize on placing a large number of orders at rapid speeds across multiple markets and multiple decision parameters based on pre-programmed instructions.

Technical Requirements for Algorithmic Trading Implementing the algorithm using a computer program is the final component of algorithmic trading, accompanied by back-testing (trying out the algorithm on historical periods of past stock-market performance to see if using it would have been profitable). The challenge is to transform the identified strategy into an integrated computerized process that has access to a trading account for placing orders. The following are the requirements for algorithmic trading:

- Computer-programming knowledge to program the required trading strategy, hired programmers, or pre-made trading software.
- Network connectivity and access to trading platforms to place orders.
- Access to market data feeds that will be monitored by the algorithm for opportunities to place orders.
- The ability and infrastructure to back-test the system once it is built before it goes live on real markets.
- Available historical data for back-testing depending on the complexity of rules implemented in the algorithm.

Objective:

- Implementing the trading strategies as algorithms and analyzing their efficiencies on the basis of profitability.
- Suggesting some new algorithms or changes in the existing ones for improvement.

Algorithm Analysis:

MEAN-REVERSION Algorithm:

Pseudo Code:

```
For records in records ['Prev Close']):
    record_count += 1
    total_closing_price += record

# Moving average is calculated for every 20 ticks(records)
If record_count >= 20:
    moving_average = total_closing_price / 20
    avg.append ([records ['Date'] [index], moving_average])

# if moving average is greater than last tick, place a buy order
If (1.02 * record < moving_average < 1.05 * record
    or moving_average < 0.8 * record) :
    Tag = 'buy'

Elif (.95 * record > moving_average > .9 * record
    or moving_average > 1.05 * record) :
    Tag = 'sell'

total_closing_price -= records ['Prev Close'] [record_count - 20]
```

VWAP Algorithm:

Pseudo Code:

```
df ['20d'] = np.round (df ['Average Price'].rolling (window=20).mean ())
df ['5d'] = np.round (df ['Average Price'].rolling (window=5).mean ())

While i < len (df):

    If (df ['5d'] [i] - df ['20d'] [i]) * (df ['5d'] [i - 1] - df ['20d'] [i - 1]) < 0:

        If df ['5d'] [i] - df ['20d'] [i] > 0:

            BAL -= df ['5d'] [i]
            bought.append ([df ['Date'] [i], df ['5d'] [i]])

        Elif df ['5d'] [i] - df ['20d'] [i] < 0: # and sbal > 0:

            BAL += df ['5d'] [i];
```

```
sold.append ([df ['Date'][i], df ['5d'][i]])
```

```
i += 1
```

BID-ASK SPREAD Algorithm:

Pseudo Code:

```
df ['20d'] = np.round (df ['Average Price'].rolling (window=20).mean ())  
df ['5d'] = np.round (df ['Average Price'].rolling (window=5).mean ())
```

```
While i < len (df):
```

```
    If (df['5d'][i] - df['20d'][i]) * (df['5d'][i - 1] - df['20d'][i - 1]) < 0:
```

```
        If df ['5d'][i] - df ['20d'][i] > 0:
```

```
            BAL -= df ['5d'][i]  
            bought.append ([df ['Date'][i], df ['5d'][i]])
```

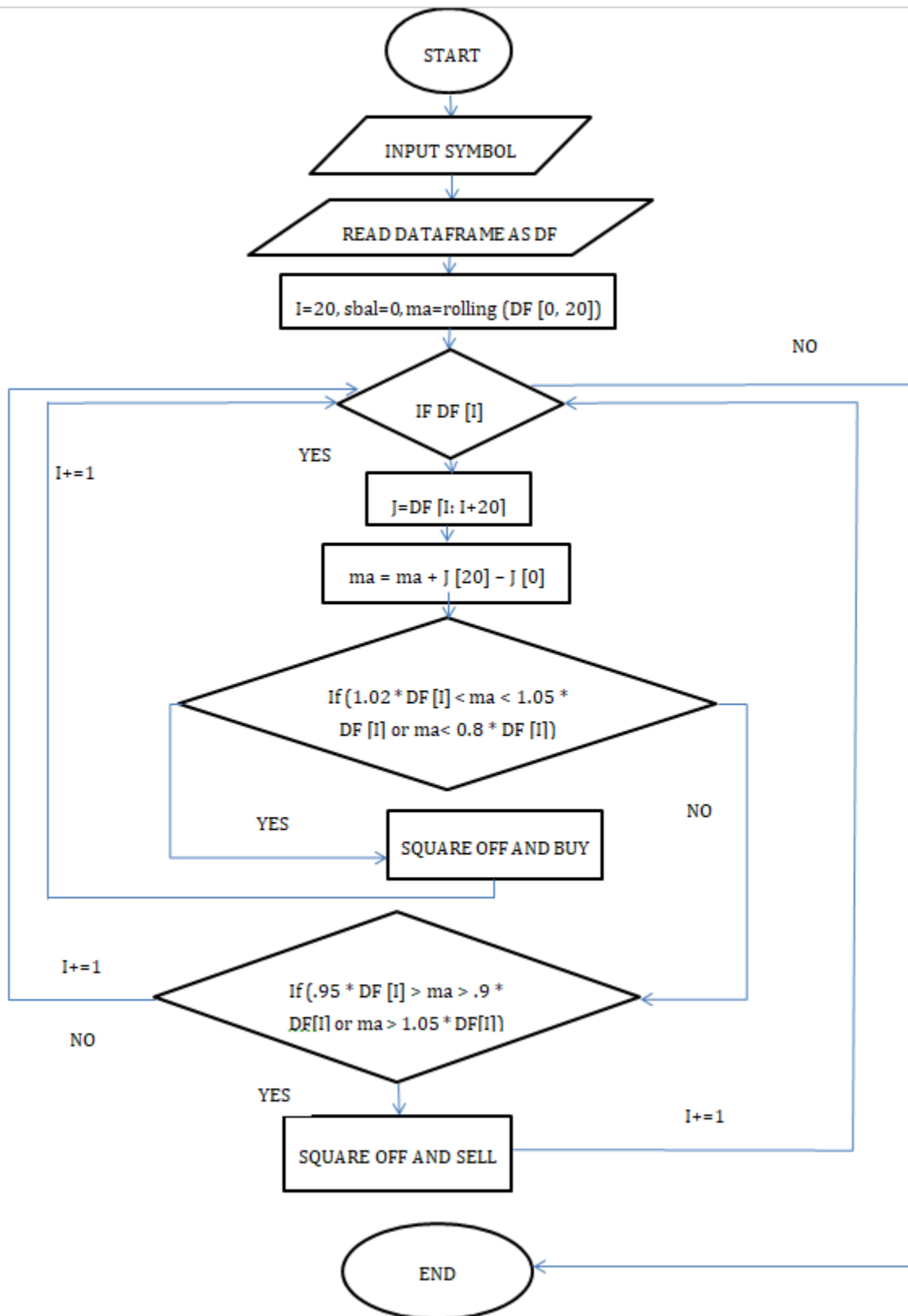
```
        Elif df ['5d'][i] - df ['20d'][i] < 0: # and sbal > 0:
```

```
            BAL += df ['5d'][i];  
            sold.append ([df ['Date'][i], df ['5d'][i]])
```

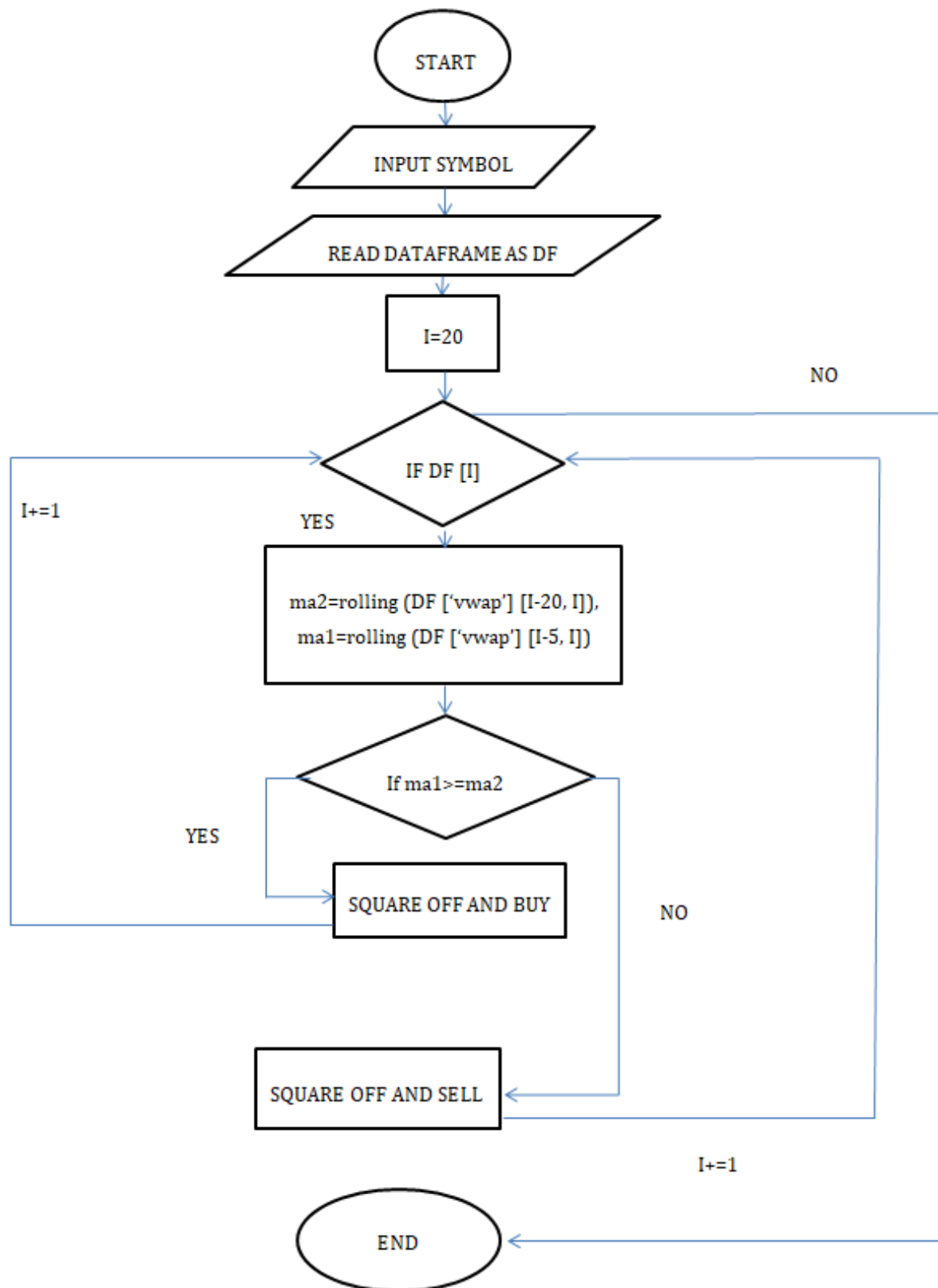
```
        i += 1
```

Algorithm Design:

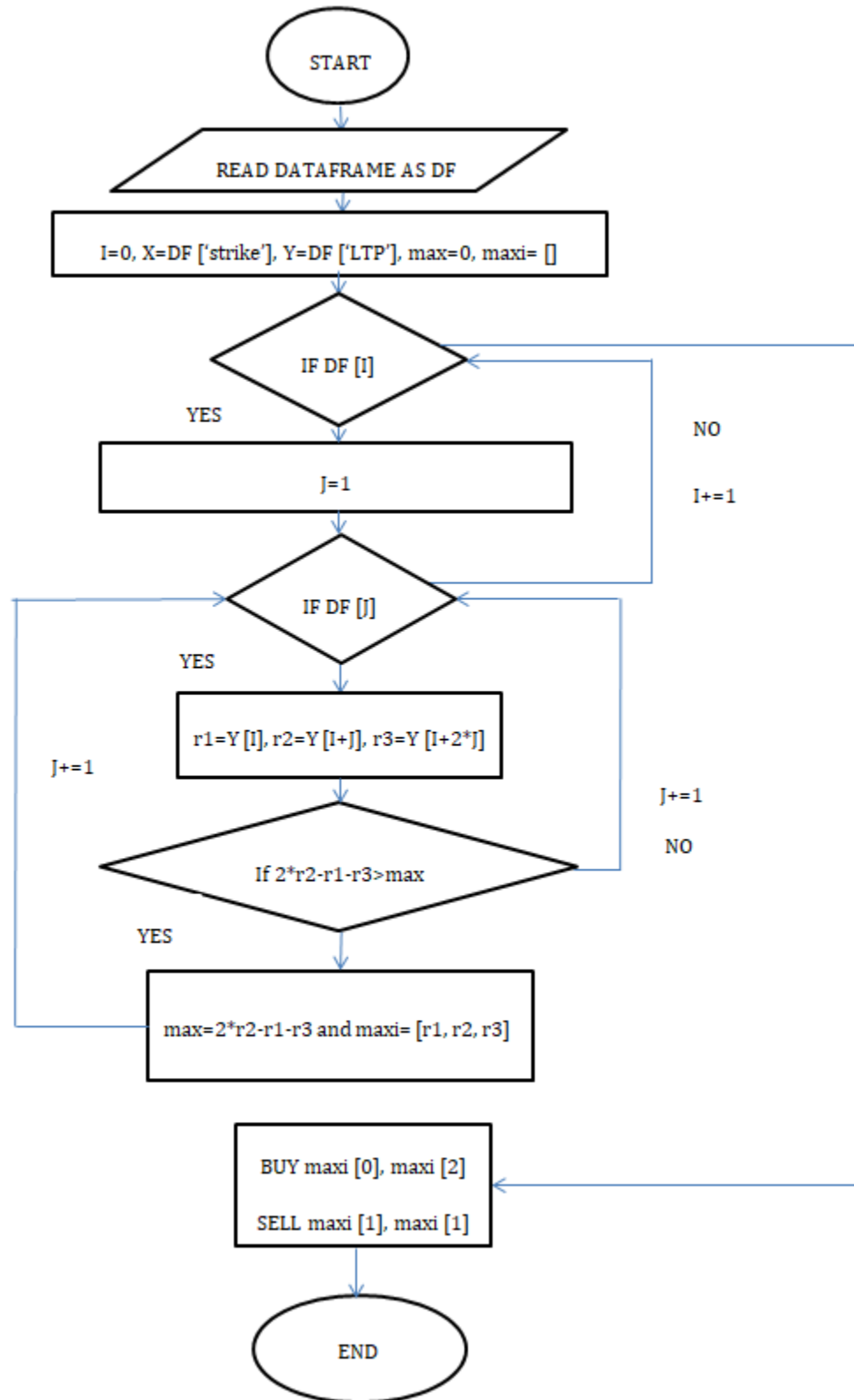
• Mean-Reversion->



• VWAP->



• BID-ASK Spread->



Complexity Analysis:

1. MEAN REVERSION:

It is simply a single loop hence the time complexity = $O(n)$

Also space complexity = $O(n)$

Where, n is the number of days for which last traded price is recorded.

2. VWAP:

Volume weighted average price is calculated as:

$$Vwap = \frac{\sum (\text{volume traded} * \text{last traded price})}{\sum (\text{volume traded})}$$

Vwap is calculated in $O(n)$ and stored separately before

Therefore, it is also simply a single loop and hence the $O(n)$

But space complexity = $2*n$

i.e. still = $O(n)$

Where, n is the number of days for which last traded price is recorded.

3. BID-ASK SPREAD:

It is of $O(n^2)$ calculation is as follows:

$$T = (n-2) + (n-5) + \dots + (n-(n-1))$$

Let T has x terms

$$\Rightarrow 2 + 3*(x-1) = n-1$$

$$\Rightarrow x = n/3$$

$$T = n \cdot (n/3) - (2+5+_{_}_{_}_{_}+n-1)$$

$$\Rightarrow T = n^2 / 3 - (2*n/3 + 3 * (1 + 2 +_{_}_{_}_{_} + (n/3 - 1))$$

$$\Rightarrow T = n^2 / 3 - (2*n/3 + (3*(n/3 - 1)*(n/3)) / 2)$$

$$\Rightarrow T = n^2/3 - (n^2 + n) / 6$$

$$\Rightarrow T = (n^2 - n) / 6$$

$$\Rightarrow T < n^2$$

Therefore time complexity = $O(n^2)$

But space complexity = $O(n)$

Where, n is the number of days for which last traded price is recorded

Implementation and Results:

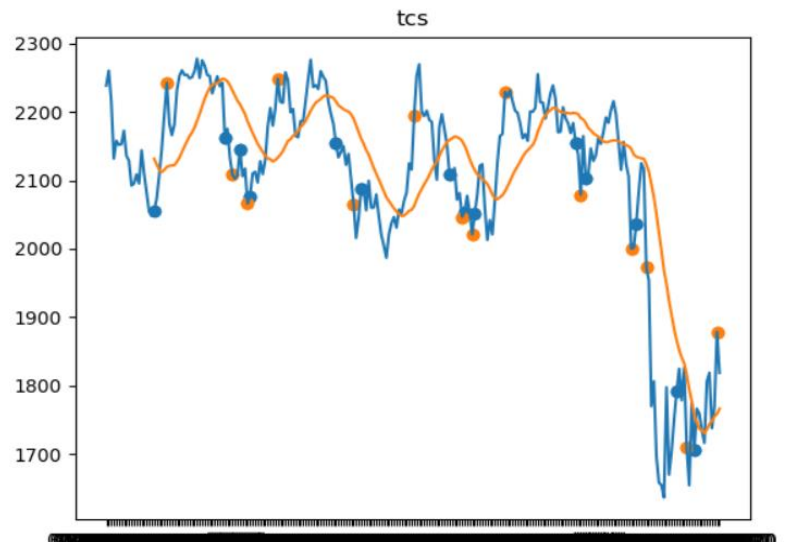
The algorithms are implemented on previous two year data

Of following shares before 14th may and then the conclusion is derived:

1. TCS

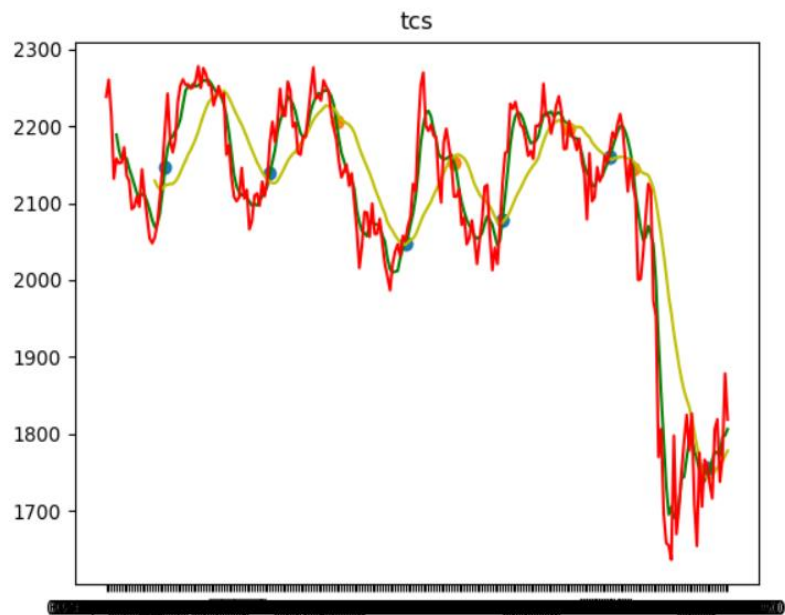
A. Mean Reversion:

```
enter the stock else e to exittcs  
time taken is 0.0 seconds  
% of returns are 28.39695399362577
```



B. VWAP:

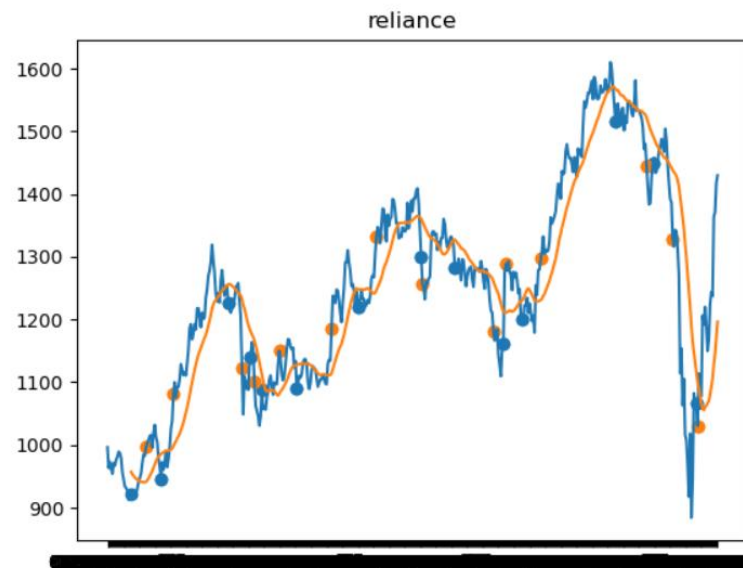
```
enter the stock else e to exittcs  
% returns are 24.4050396640224  
investment required : 4286.0
```



2. RELIANCE

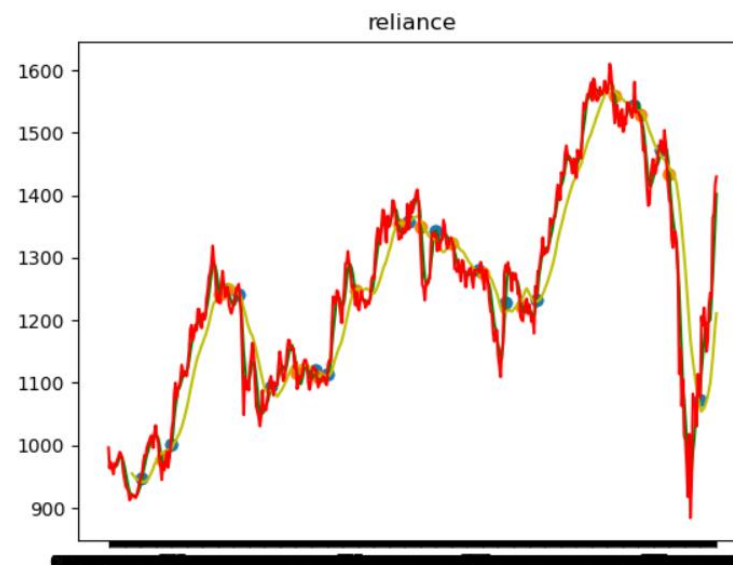
A. Mean Reversion:

```
enter the stock else e to exitreliance  
time taken is 0.0 seconds  
% of returns are -10.032651791961449
```



B. VWAP:

```
enter the stock else e to exitreliance  
% returns are 58.70503597122302  
investment required : 2085.0
```

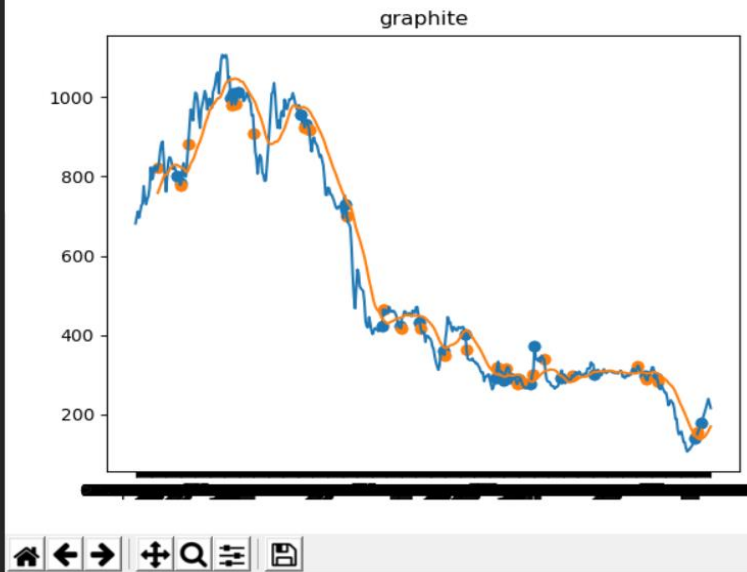


3. GRAPHITE

A. Mean Reversion:

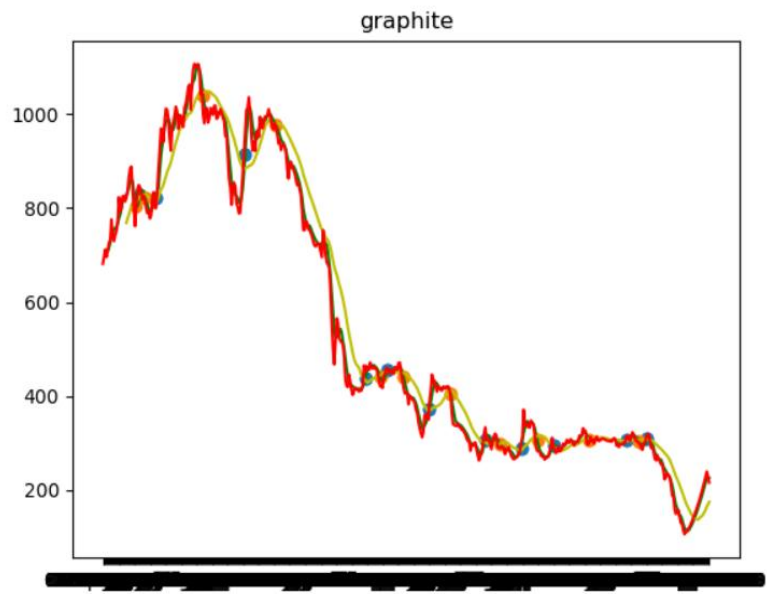
```
enter the stock else e to exitgraphite  
time taken is 0.0 seconds  
% of returns are 33.11026131293815
```

Figure 1



B. VWAP:

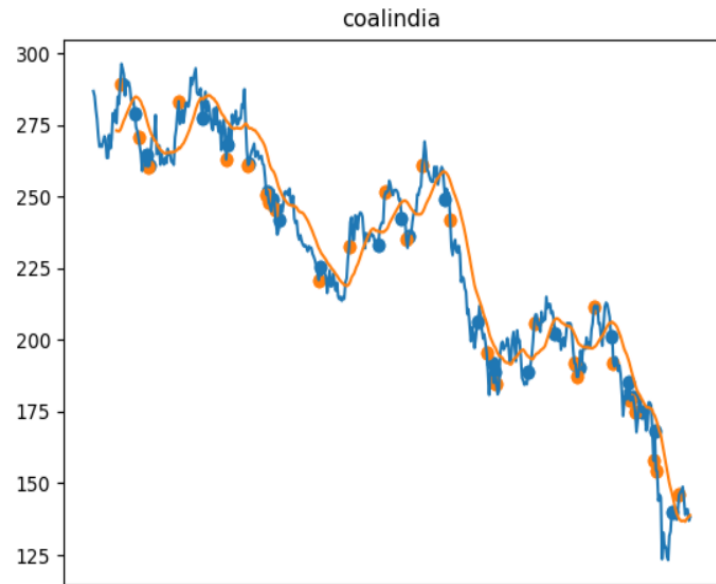
```
enter the stock else e to exitgraphite  
% returns are 121.80539273153576  
investment required : 853.0
```



4. COALINDIA

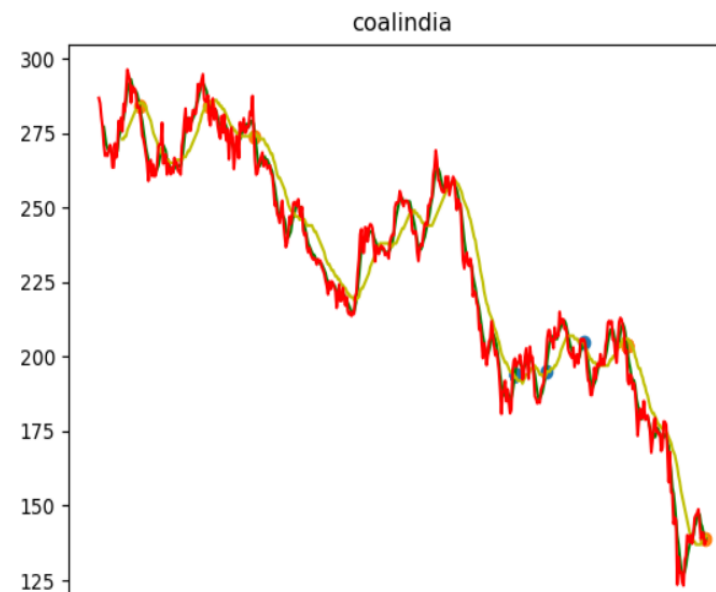
A. Mean Reversion:

```
enter the stock else e to exit coalindia  
time taken is 0.0 seconds  
% of returns are 12.395465526853679
```



B. VWAP:

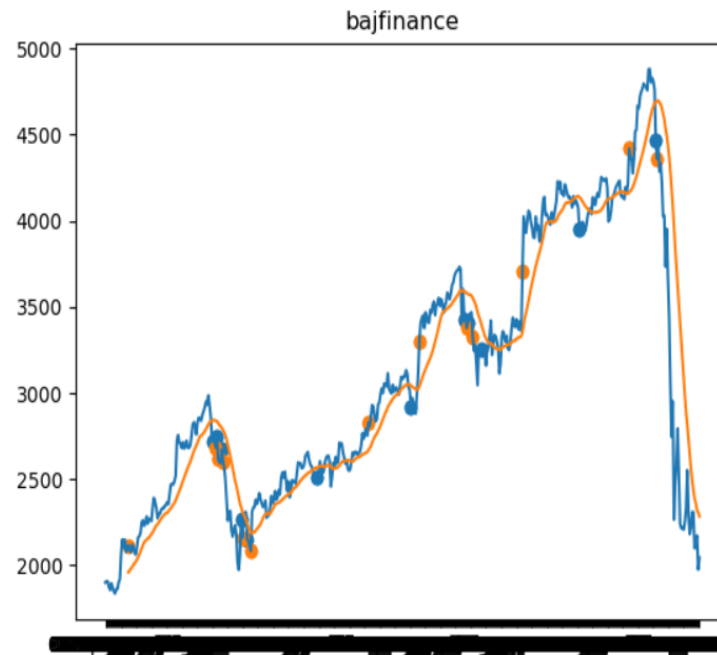
```
enter the stock else e to exit coalindia  
% returns are 102.6946107784431  
investment required : 334.0
```



5. BAJFINANCE

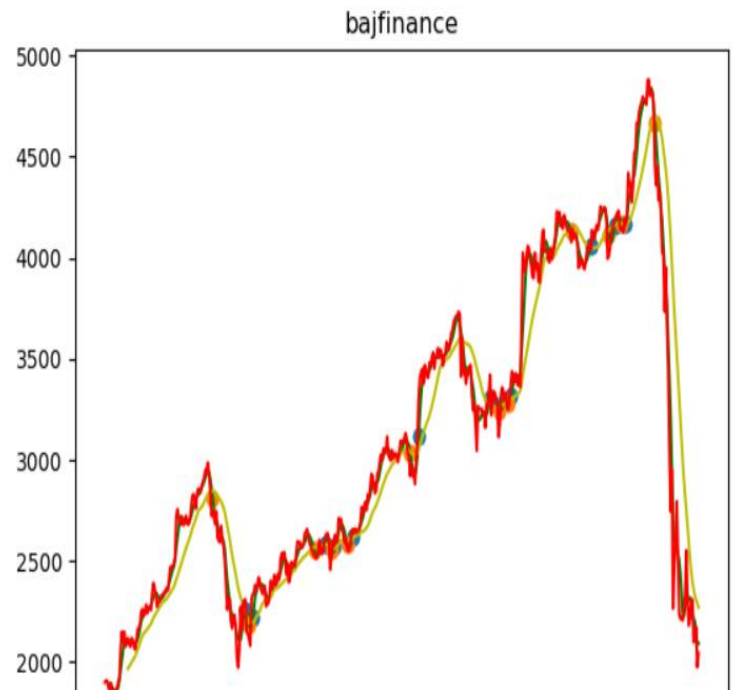
A. Mean Reversion:

```
enter the stock else e to exitbajfinance  
time taken is 0.0 seconds  
% of returns are 42.29138254347082
```



B. VWAP:

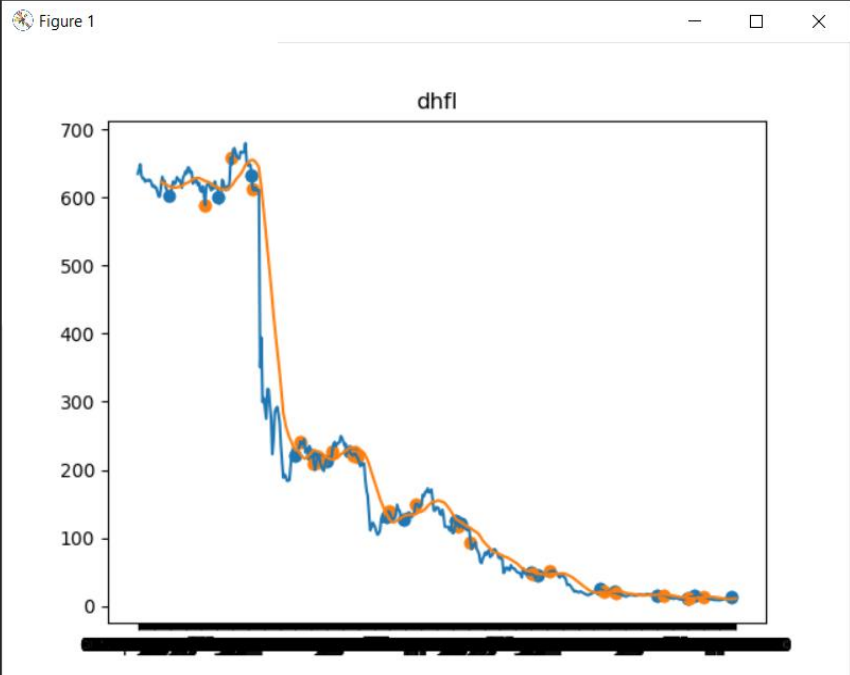
```
enter the stock else e to exitbajfinance  
% returns are 91.88790560471976  
investment required : 5424.0
```



6. DHFL

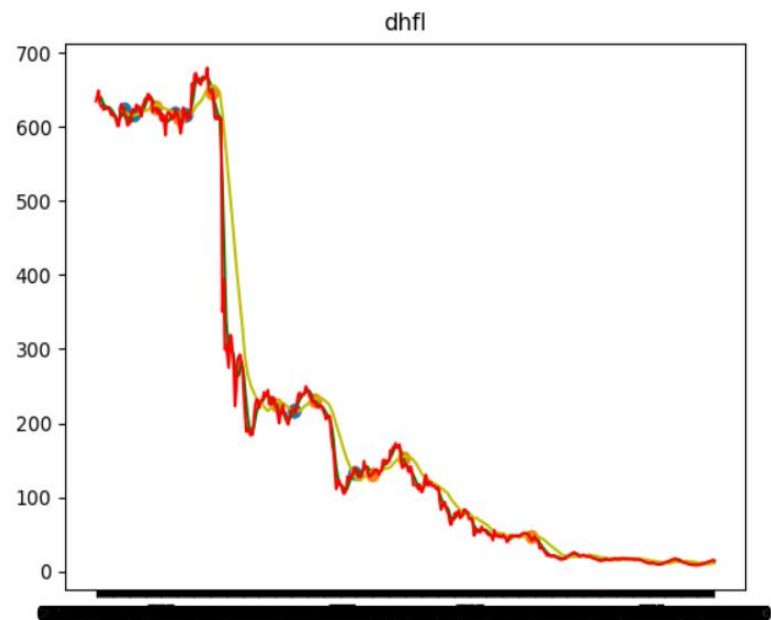
A. Mean Reversion:

```
enter the stock else e to exit dhfl  
time taken is 0.0 seconds  
% of returns are 101.71117863425563
```



B. VWAP:

```
enter the stock else e to exit dhfl  
% returns are 71.07171635777598  
investment required : 1241.0
```

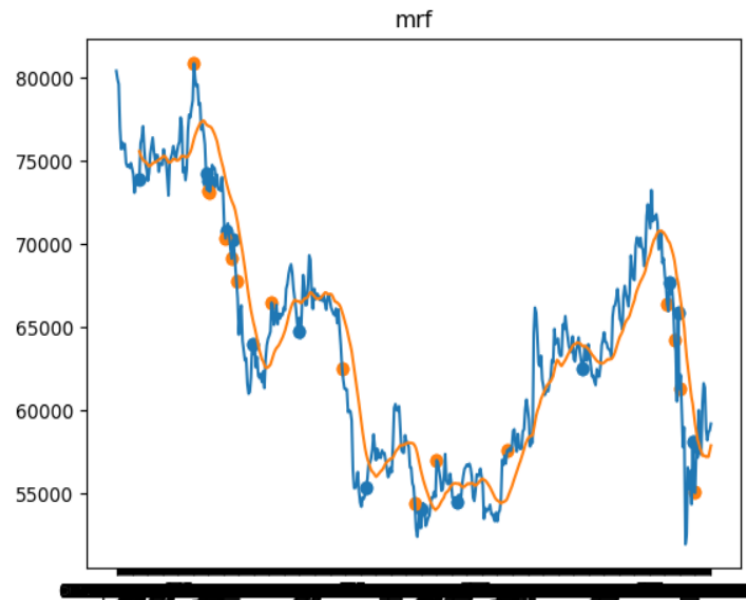


7. MRF

A. Mean Reversion:

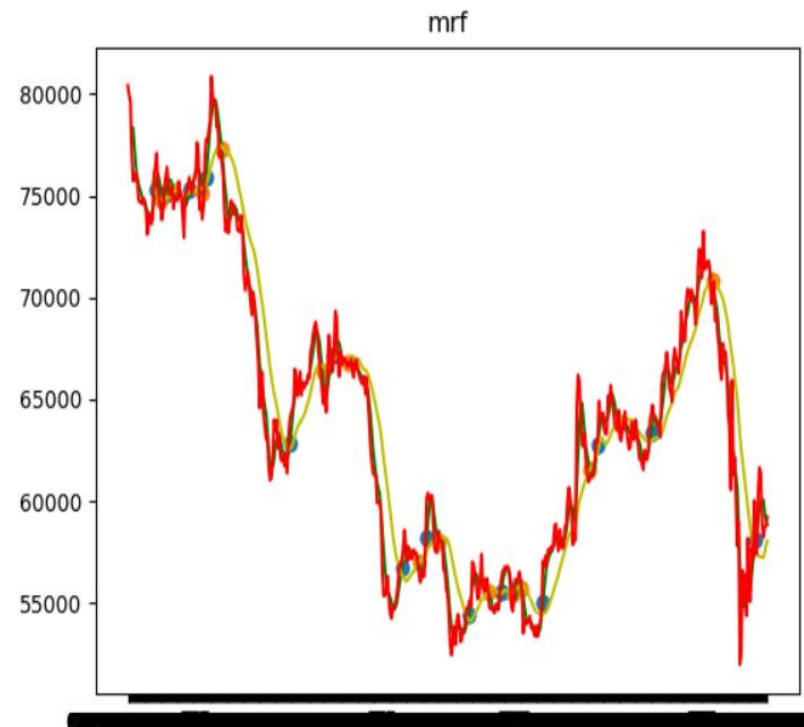
```
enter the stock else e to exitmrf  
time taken is 0.0 seconds  
% of returns are 8.596804501099255
```

Figure 1



B. VWAP:

```
enter the stock else e to exitmrf  
% returns are 74.50722733245729  
investment required : 77622.0
```



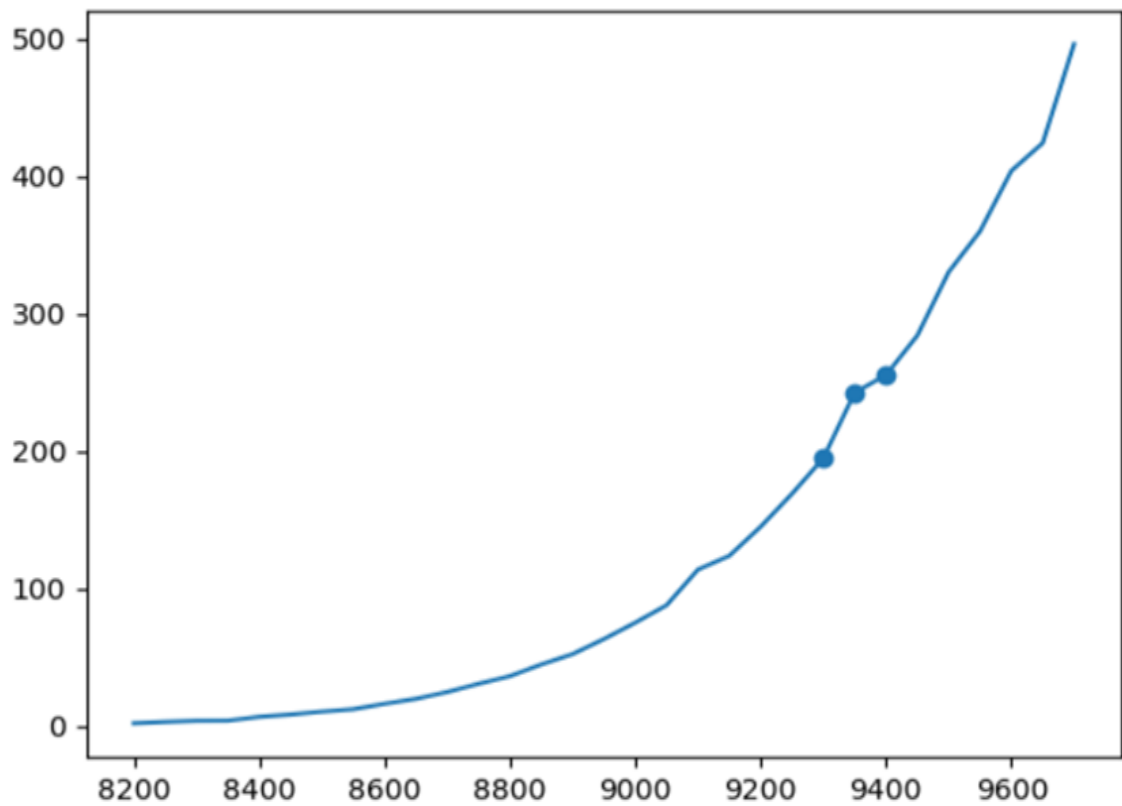
IMPLEMENTAION OF BID-ASK SPREAD WITH OPTION CHAIN

OF 11th may FOR 14th may EXPIRY FOR RISK-FREE TRADE:

STRIKE PRICE	BID QTY	BID PRICE	ASK PRICE	ASK QTY	CHNG	LTP	IV	VOLUME	CHNG IN OI	OI
8200.00	150	2.25	2.55	75	-4.55	2.25	46.06	8885	572	2201
8250.00	750	3.3	4.75	-68.1	3.25	-	42	25	25	
8300.00	1125	3.45	4.4	600	-5.9	4.05	44.21	11228	1244	2991
8350.00	150	4.15	5.9	1200	-8.5	4.15	42.61	446	171	175
8400.00	75	6.55	6.9	7350	-7.25	6.9	42.33	18973	1167	2827
8450.00	525	8.7	9.45	1200	-7.75	8.6	41.6	994	58	157
8500.00	75	10.6	11.525	-9.6	10.7	40.97	51355	3051	12462	
8550.00	1200	11.2	13.8	75	-11.25	12.45	40.22	1773	155	238
8600.00	225	16.3	17.1275	-12.6	16.4	39.87	34970	2773	4925	
8650.00	300	19.4	20.75	-14.45	20	39.38	3601	329	388	
8700.00	375	24.4	25.750	-15.8	24.95	39.04	44791	844	4737	
8750.00	75	29.2	31.25	300	-17	30.95	38.72	6326	481	596
8800.00	3375	36.5	37.45	675	-20.55	36.5	38.29	104215	3204	10410
8850.00	225	42.85	45.450	-20.9	45	38.1	12782	348	510	
8900.00	150	52.5	53.825	-25.2	52.55	37.86	102222	-1021	7988	
8950.00	75	62.45	64.225	-28	63.5	37.57	12831	570	830	
9000.00	75	75.4	76.675	-31.8	75.4	37.41	207412	3608	15884	
9050.00	150	88	89.5	75	-34.85	88.1	37.11	16295	366	617
9100.00	225	106.1	108.35	75	-27.65	114	36.93	136259	3610	6733
9150.00	75	123.1	126.75	-38.7	124	36.89	15117	414	717	
9200.00	150	145.15	147.450	-39.25	145.15	36.86	177070	-356	9141	
9250.00	75	167.05	168.8	75	-39.75	169.1	36.73	21403	719	938
9300.00	450	193	195.45	150	-42.1	195	36.86	179444	3000	7050
9350.00	75	221.6	226.35	150	-23	242.65	37.04	17436	539	568
9400.00	225	254	256.25	150	-43.15	255.5	36.94	51993	991	2147
9450.00	75	281.65	291.85	75	-38.55	284.2	36.58	850	73	76
9500.00	225	325.2	328.45	75	-40.45	330.4	37.39	20023	3091	5016
9550.00	75	359.25	382.8	75	-55	360	32.71	96	18	22
9600.00	75	400.7	412.45	1200	-37.65	404	37.25	2804	80	278
9650.00	150	432.35	458.45	225	-79.05	424.2	-	1	1	1
9700.00	75	490.500	500.25	1050	-33.3	496.15	33.45	631	63	321

Strike prices with their premium and maximum profit trade

```
most profitable range of trade is [9300.0, 9350.0, 9400.0]  
premiums of most profitable range are [195.0, 242.65, 255.5]  
profits made by a single lot of most profitable trade is 2610.0000000000001  
  
Process finished with exit code 0  
|
```



Conclusion and Future Work:

1. VWAP is a better strategy for blue-chip, stable and non-volatile stocks.
2. Mean-reversion is advantageous when some share shows unexpected results or is very much volatile.
3. By means of bid-ask spread we can find opportunities for risk-free trades with infinite returns in the option chain.
4. Black-box trading or Algo-trading is the future of trade. Today we have a lot of indicators and technical tools for trading and the field is still developing at a nice pace so in near future development of even more better algorithms and strategies is very much a possibility.

References:

<https://www1.nseindia.com/>

<https://financialmodelingprep.com/developer/docs/>

<https://www.investopedia.com/>