

```
In [15]: from IPython.display import display, Math, Latex
import pandas as pd
import numpy as np
import yfinance as yf
import matplotlib.pyplot as plt
```

```
In [16]: limit_book = pd.read_csv("BTCUSD_2019_09_01.csv")
limit_book
```

```
Out[16]:
```

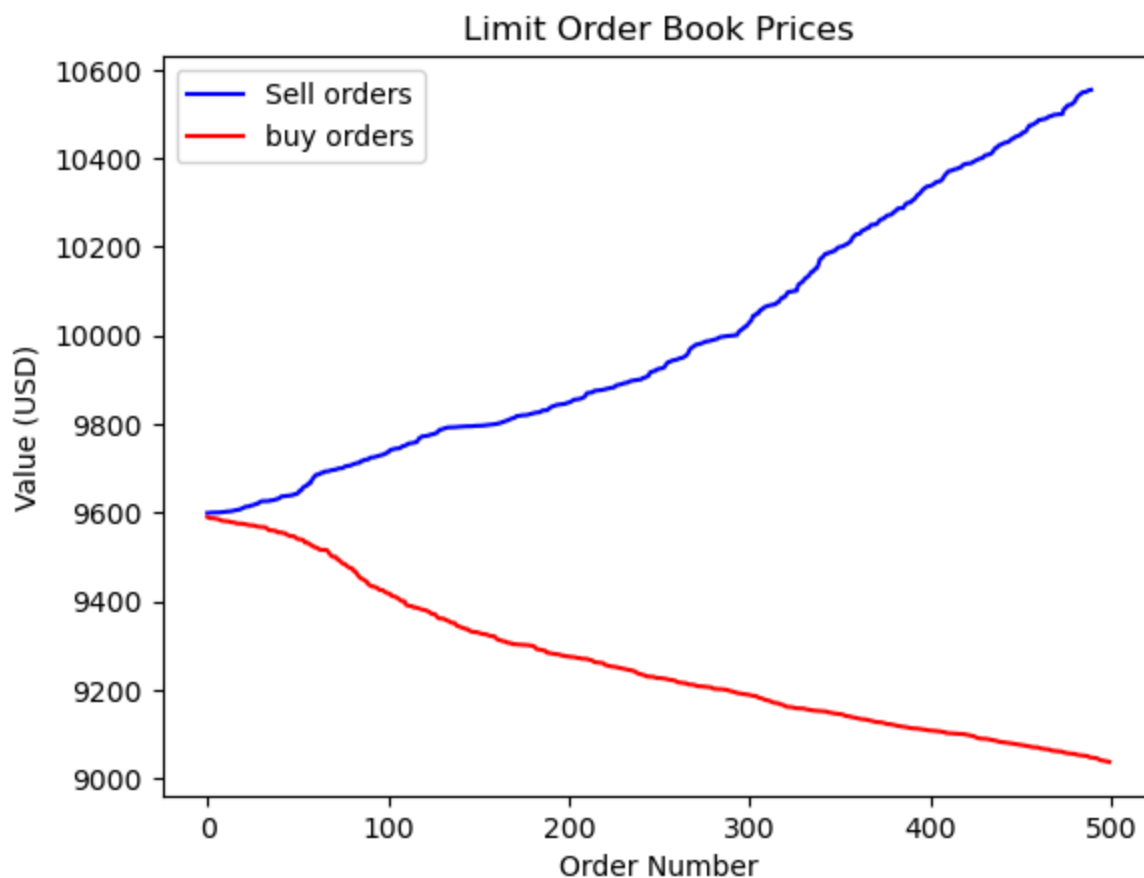
	type	b_price	b_amount	type.1	s_price	s_amount
0	b	9590.0	0.749	s	9599.0	2.000
1	b	9588.0	0.172	s	9599.1	0.040
2	b	9587.2	0.307	s	9599.8	1.815
3	b	9587.1	0.108	s	9599.9	0.681
4	b	9587.0	0.040	s	9600.0	0.053
...	...	...	...	...	...	...
495	b	9040.0	5.001	NaN	NaN	NaN
496	b	9039.6	0.151	NaN	NaN	NaN
497	b	9038.3	0.100	NaN	NaN	NaN
498	b	9038.0	0.028	NaN	NaN	NaN
499	b	9037.0	0.750	NaN	NaN	NaN

500 rows × 6 columns

```
In [17]: sell = limit_book[["s_price", "s_amount"]]
buy = limit_book[["b_price", "b_amount"]]
```

```
In [18]: # Plotting the buy price and sell price
plt.plot(sell.s_price, color='b', label='Sell orders')
plt.plot(buy.b_price, color='r', label='buy orders')
plt.legend(loc='best')
plt.title('Limit Order Book Prices')
plt.xlabel('Order Number')
plt.ylabel('Value (USD)')
```

```
Out[18]: Text(0, 0.5, 'Value (USD)')
```



```
In [19]: bid=buy.b_price[0]
ask=sell.s_price[0]
print("bid: $",bid, sep='')
print("ask: $",ask, sep='')
print("bid-ask spread: $",ask-bid, sep='')

```

```
bid: $9590.0
ask: $9599.0
bid-ask spread: $9.0

```

```
In [20]: # A Market Buy Order.
order_size= 500
shares=0
count=0
cost=0
diff=0

while shares < order_size:
    shares= shares + sell.s_amount[count]
    if shares > order_size:
        diff = shares - order_size
        shares = order_size
    cost+=sell.s_price[count]*(sell.s_amount[count] - diff)
    count+=1
print("total shares", shares)
print("Total Cost: $",np.round_(cost, decimals=2), sep='')
print("Average Price Paid: $",np.round_(cost/order_size, decimals=2), sep='')

```

```
print("Last Transaction Price BOUGHT: $",np.round_(sell.s_price[count-1], de
print("Price Movement: ", sell.s_price[count - 1] - sell.s_price[0])
```

total shares 500  
 Total Cost: \$4890662.45  
 Average Price Paid: \$9781.32  
 Last Transaction Price BOUGHT: \$10000.0  
 Price Movement: 401.0

```
In [21]: # A Market Sell Order.
order_size= 300
shares=0
count=0
proceeds =0
diff=0
while shares < order_size:
    shares= shares + buy.b_amount[count]
    if shares > order_size:
        diff = shares - order_size
        shares = order_size
    proceeds+=buy.b_price[count]*(buy.b_amount[count] - diff)
    count+=1
print("total shares", shares)
print("Total Proceeds: $",np.round_(proceeds, decimals=2), sep='')
print("Average Price SOLD: $",np.round_(proceeds/order_size, decimals=2), se
print("Last Price SOLD: $",np.round_(buy.b_price[count-1], decimals=2), sep=
print("Price Movement: ", buy.b_price[count - 1] - buy.b_price[0])
```

total shares 300  
 Total Proceeds: \$2841942.42  
 Average Price SOLD: \$9473.14  
 Last Price SOLD: \$9338.0  
 Price Movement: -252.0

```
In [22]: # A Limit Order Buy
order_size = 500
shares_bought = 0
price_willing = 9800
count = 0
cost = 0
diff = 0
while shares_bought < order_size and sell.s_price[count] <= price_willing:
    shares_bought = shares_bought + sell.s_amount[count]
    if shares_bought > order_size:
        diff = shares_bought - order_size
        shares_bought = order_size
    cost += sell.s_price[count] * (sell.s_amount[count] - diff)
    count +=1

print("Total shares bought is ", shares_bought)
print("Total Cost is ", np.round(cost, decimals = 2), sep='')

try:
    average_price = np.round(cost / shares_bought, decimals = 2)
    price_movement = np.around(sell.s_price[count - 1] - sell.s_price[0])
    print("Average Price sold is",average_price, sep='')
    print("Price Movement is", price_movement)
```

```
except:
    print("No average Price sold is available")
```

Total shares bought is 285.4439999999996  
 Total Cost is 2766670.66  
 Average Price sold is 9692.52  
 Price Movement is 201.0

```
In [23]: # A Limit Order Sell
shares_sold = 0
price_willing = 9400
order_size = 50
proceeds = 0
differ = 0
count = 0
while shares_sold < order_size and buy.b_price[count] >= price_willing:
    shares_sold = shares_sold + buy.b_amount[count]
    if shares_sold > order_size:
        differ = shares_sold - order_size
        shares_sold = order_size
    proceeds += buy.b_price[count] * (buy.b_amount[count] - differ)
    count += 1

print("Total shares sold is ", shares_sold)
print("Total Proceed is ", np.round(proceeds, decimals = 2), sep='')

try:
    average_price = np.round(proceeds/shares_sold, decimals = 2)
    price_movement = np.around(buy.b_price[count - 1] - buy.b_price[0])
    print("Average Price sold is ", average_price, sep='')
    print("Price Movement is", price_movement)

except:
    print("No average Price sold is available")
```

Total shares sold is 50  
 Total Proceed is 478798.22  
 Average Price sold is 9575.96  
 Price Movement is -22.0