The primary objective of the task is for the user to see if the BID or the ASK have the bigger value in a certain time period. The periods are 1 minute, 5 minutes, 20 minutes and 60 minutes.

The app will need to store the sum of all bids and asks for these time periods.

If the interval setting from the config. file = 1 sec.

Variable A will sum bid + bid + bid … for each second for 60 seconds

Variable B will sum bid + bid + bid … for each second for 300 seconds

Variable C will sum bid + bid + bid … for each second for 1200 seconds

Variable D will sum bid + bid + bid … for each second for 3600 seconds

The same formula goes for ASK for variables E, F, G, H, and I.

Now, if the user has set the interval to 0.5 seconds the same logic as above is applied, but the end result numbers will be x2. The same logic goes, depending on what time the user chooses to set in the interval setting in the configuration file.

Now let`s assume that we have the following sum results from above formulas:

1 min bid = 3466 ; 1 min ask = 5700

5 min bid = 24847; 5 min ask = 47838

20 min bid = 158733 ; 20 min ask = 478234

60 min bid = 1432682 ; 60 min ask = 3540782

The algorithm should first compare which number is bigger. The one from the bid or the one from the ask. So, after this is calculated it will look like this:

3466 < 5700

24847 < 47838

478234 > 158733

3540782 > 1432682

Then, the algorithm must apply a simple divide. Where it divides the bigger number to the smaller one, in order to calculate the ratio:

5700 : 3466

47838 : 24847

478234 : 158733

3540782 : 1432682

And finally, it has to display the results

1 min. Bid 1:1.6 Ask

5 min. Bid 1:1.9 Ask

20 min. Bid 1:3 Ask

60 min. Bid 2.4:1 Ask

Please note that above results should replace the values in the red circle as they will not be needed anymore:

