Lai-Yang-Li Algorithm

- Implemented by Sushovan Chaudhury
- Id:2020MT13248

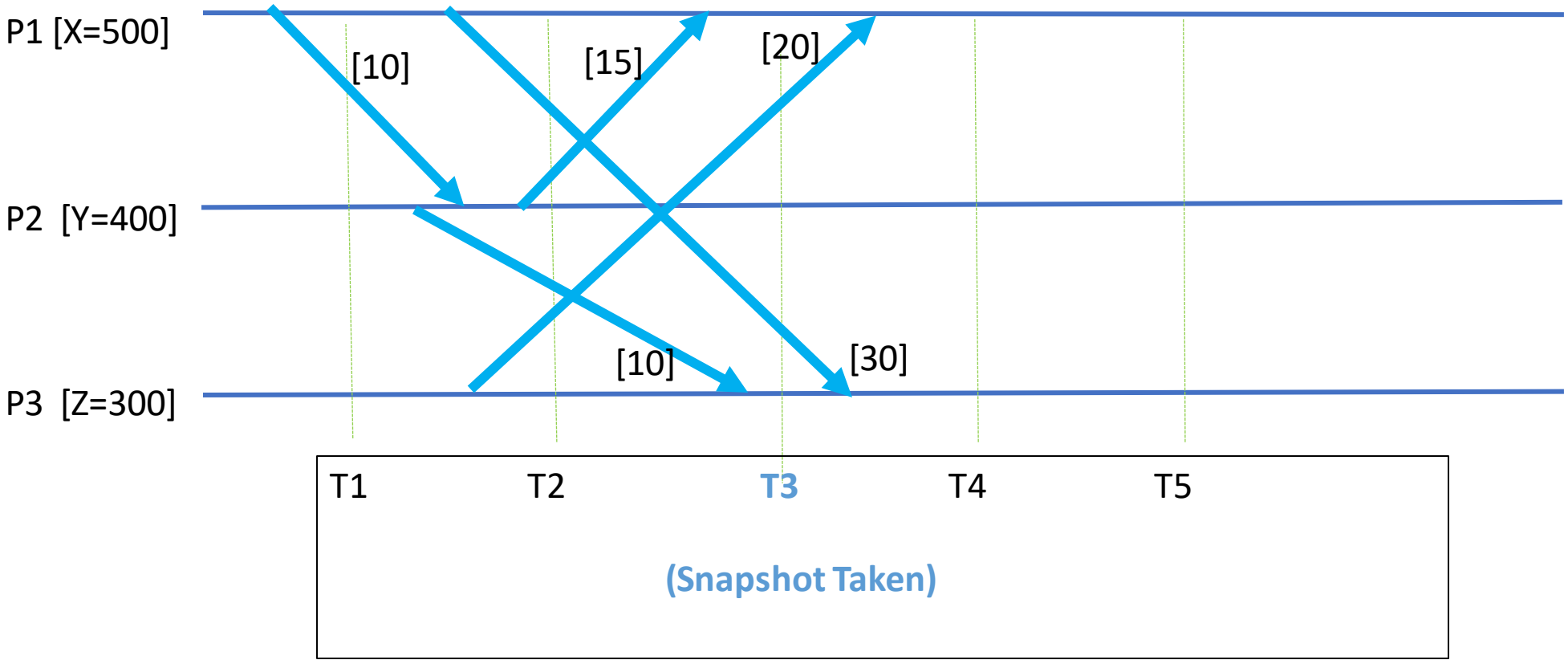
Tools and Platform

- To run the program user can use any 32/64 bit OS.
- Need to have some IDE or shell supporting python environment.
- Can be run in spyder, pycharm.jupyter,pytorch or any supported IDE
- Simply open the file and press the run button in the respective IDE
- This program is tested on windows10, 64 bit system with 8GB RAM and Spyder 4.0 IDE is used for developing the code in python 3.8

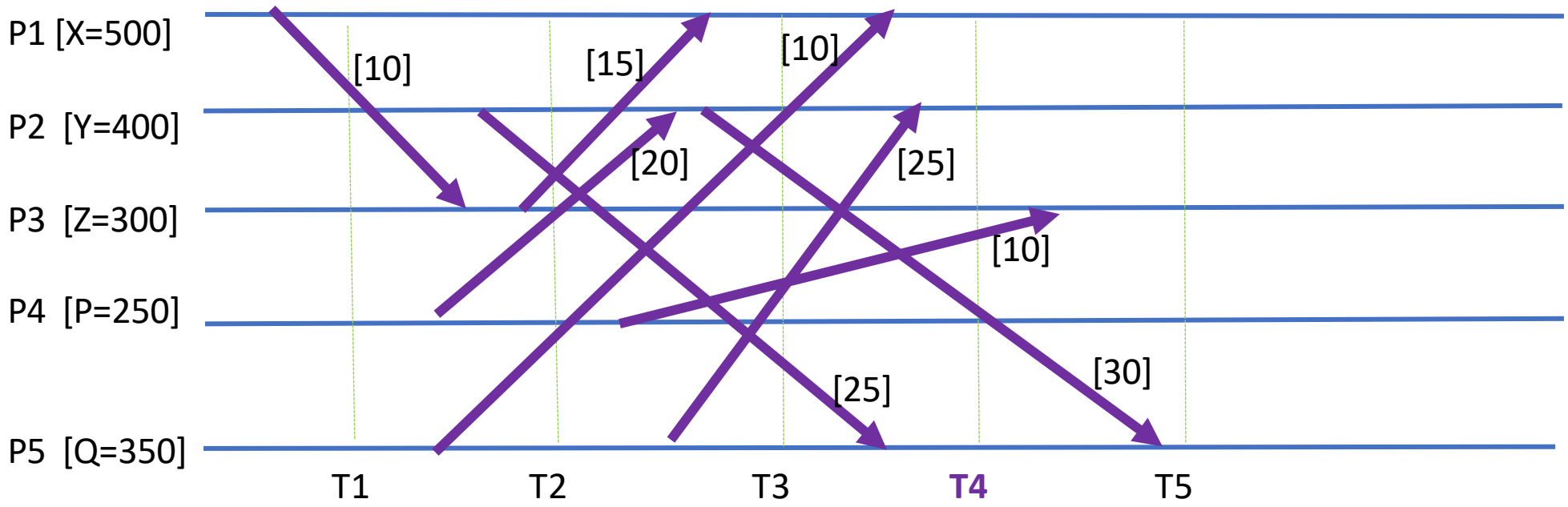
Implementation Assumptions

- 1.The system can handle at most 5 processes/sites, Increasing the number of processes complicate the data structures being used
- 2.Any messages in transit should be handled during program input
- 3.Messages being sent between two consecutive timestamps are assumed to be sent in the next higher timestamp as per convention
- 4.The algorithm is primarily for NON-FIFO and causal ordering of messages are not maintained, however irrespective of FIFO and NON-FIFO Communication channel, it works because such complexities are handled in input
- 5.The program only deals with white messages as they are used to calculate channel states locally
- 6.Messages in transit are calculated in channel states within the program
- 7.White messages received after the first snapshot are ignored
- 8.The program can accommodate a single snapshot at a time and check for consistency. If snapshot is taken again, rerun the program and reset all messages as white
- 9.Program successfully checks if the snapshot being taken is consistent or not
- 10.The programmer assumes that for better clarity the end user has a logical State-Time diagram of the problem handy to compare results

Algorithm Implementation with 3 Processes (Explained in this ReadMe)



Algorithm Implementation with 5 Processes (Can be tested with)



(Snapshot Taken)

Test Run:

- Preparation of Input with 3 Processes
- We input the events happening in processes at a given time stamp. We give the input separately for sent and received event. The program interactively asks for sent and receipt input for each time stamp.
- Each process can send or receive messages from all other processes except itself.
- Since our program can accommodate 5 processes at most, we enter p=5 as user but for lower number of processes, we input 0 for all those events which are not relevant. We will also input the number of time stamps in the system.

Preparing Input

- As per diagram if an event occur between t1 and t2, we assume it to be happened in t2.In general, in such cases we assume that the event occurred in next time stamp in sequence.
- While inputting we have to give cumulative value of sent and receipt till that time stamp. For example if P1 send 10 INR to P3 at t2 and 20 INR to P3 at t3,in the sent array of P1 against P3 at t2, we input 10 but in the array we input 30(20+10) at t3 to consider the cumulative sum of the previous sum.
- One such input array of sent and receipt for each process and 5 timestamps are shown in next page.
- Please put all 0s for events of P4 and P5 as we are dealing with 3 processes. For the other problem with 5 processes, we should give input accordingly.

Sent and Receipt of P1 forexample problem

Time Stamp(SENT)	P2	P3	P4	P5
T1	10	0	0	0
T2	10	30	0	0
T3	10	30	0	0
T4	10	30	0	0
T5	10	30	0	0

Time Stamp (RECEIPT)	P2	P3	P4	P5
T1	0	0	0	0
T2	0	0	0	0
T3	15	0	0	0
T4	15	20	0	0
T5	15	20	0	0

Sent and Receipt of P2 forexample problem

Time Stamp(SENT)	P1	P3	P4	P5	Time Stamp (RECEIPT)	P1	P3	P4	P5
T1	0	0	0	0	T1	0	0	0	0
T2	15	10	0	0	T2	10	0	0	0
T3	15	10	0	0	T3	10	0	0	0
T4	15	10	0	0	T4	10	0	0	0
T5	15	10	0	0	T5	10	0	0	0

Sent and Receipt of P3 for the example problem

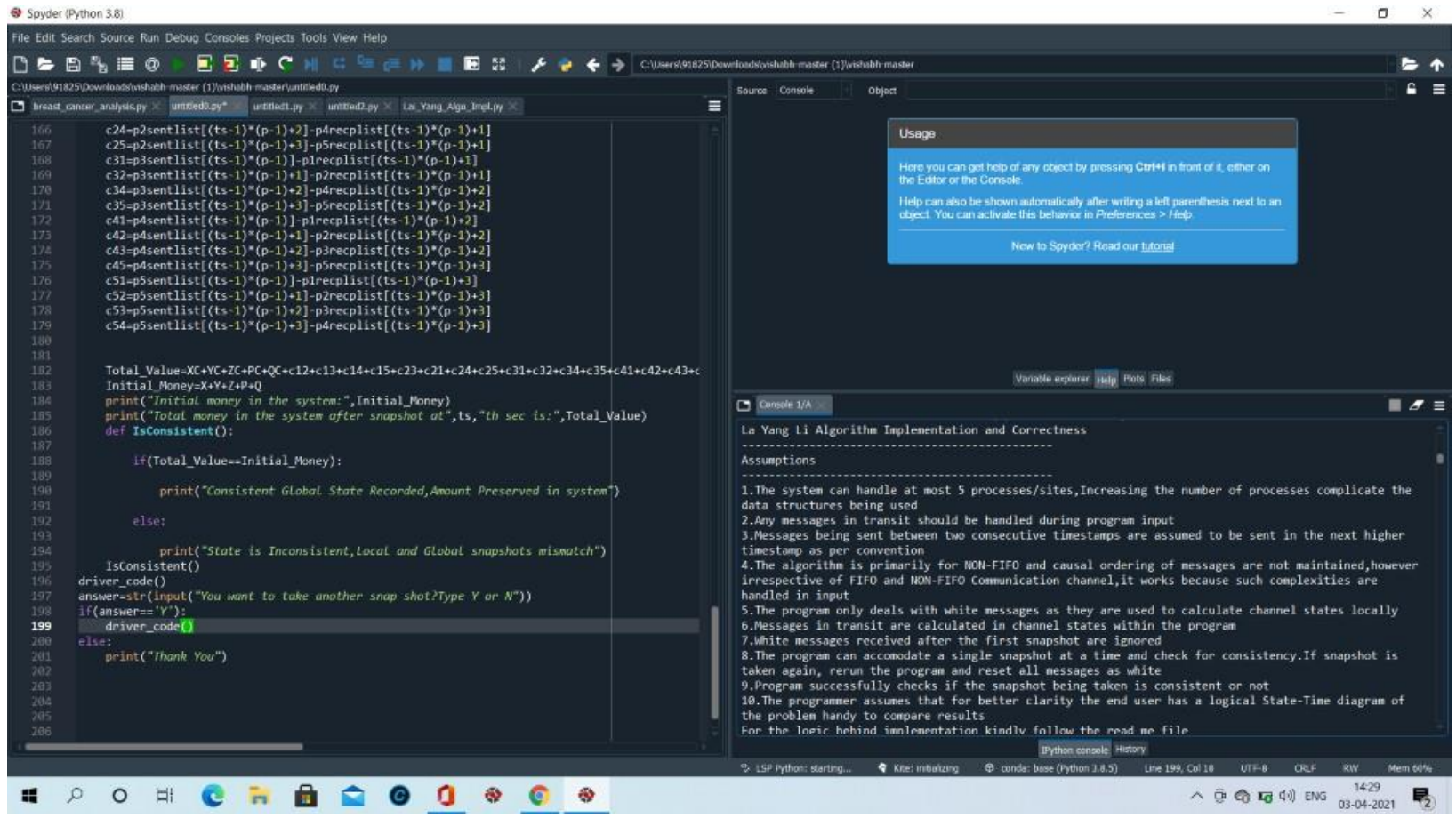
Time Stamp(SENT)	P1	P2	P4	P5
T1	0	0	0	0
T2	20	0	0	0
T3	20	0	0	0
T4	20	0	0	0
T5	20	0	0	0

Time Stamp (RECEIPT)	P1	P2	P4	P5
T1	0	0	0	0
T2	0	0	0	0
T3	0	10	0	0
T4	30	10	0	0
T5	30	10	0	0

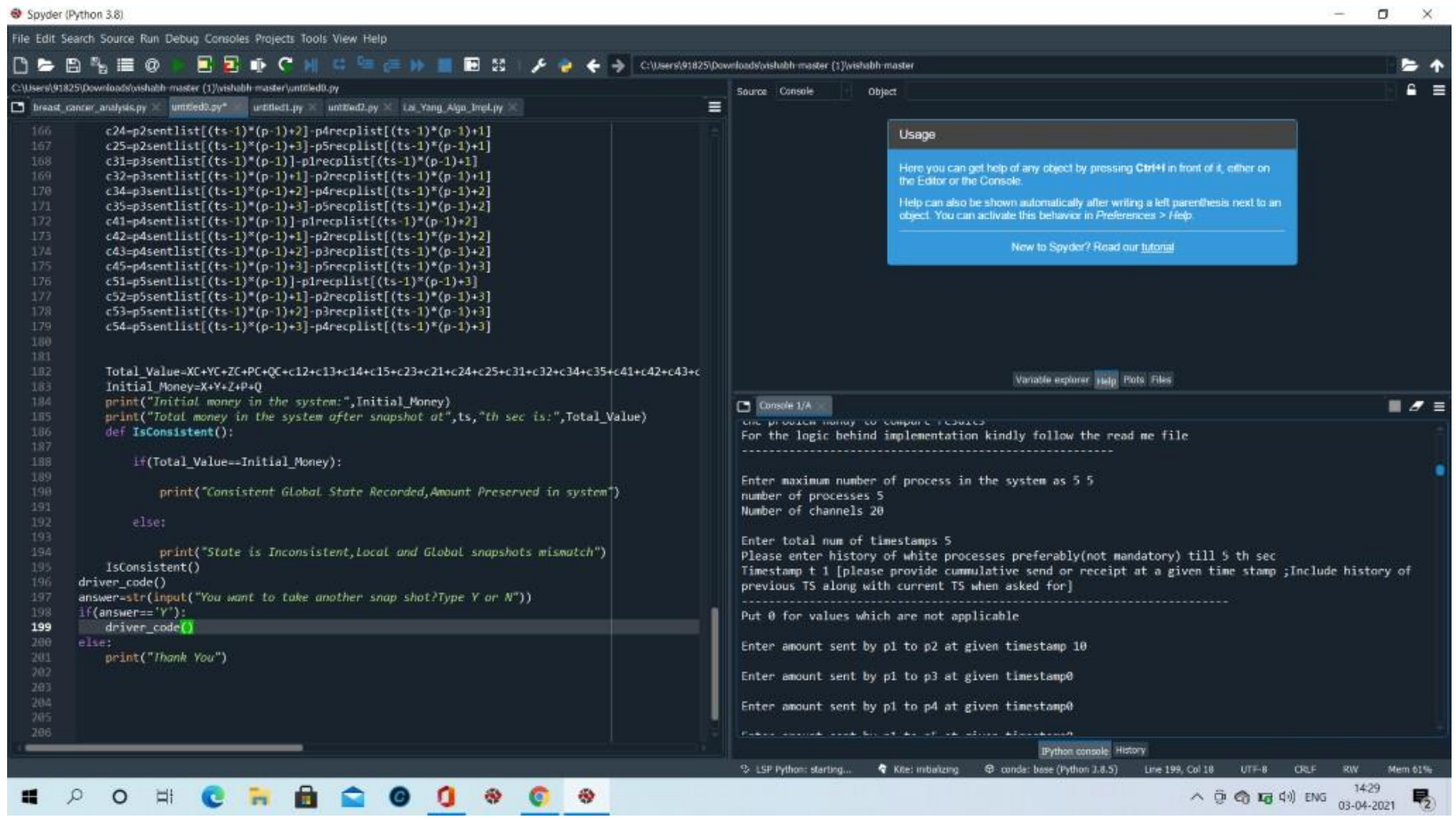
Preparing input

- After all the events(sent or receipt) for each process and time stamp is stored in lists , the system asks for initial money with each of the processes and the time stamp where user wants to take a snapshot.
- In our example sent and receipt of P4 and P5 are all zeros because we considered 3 processes and respective arrays of p4 and p5 are also zero. Hence not shown in diagram, unlike sent and receipt of P1,P2,P3

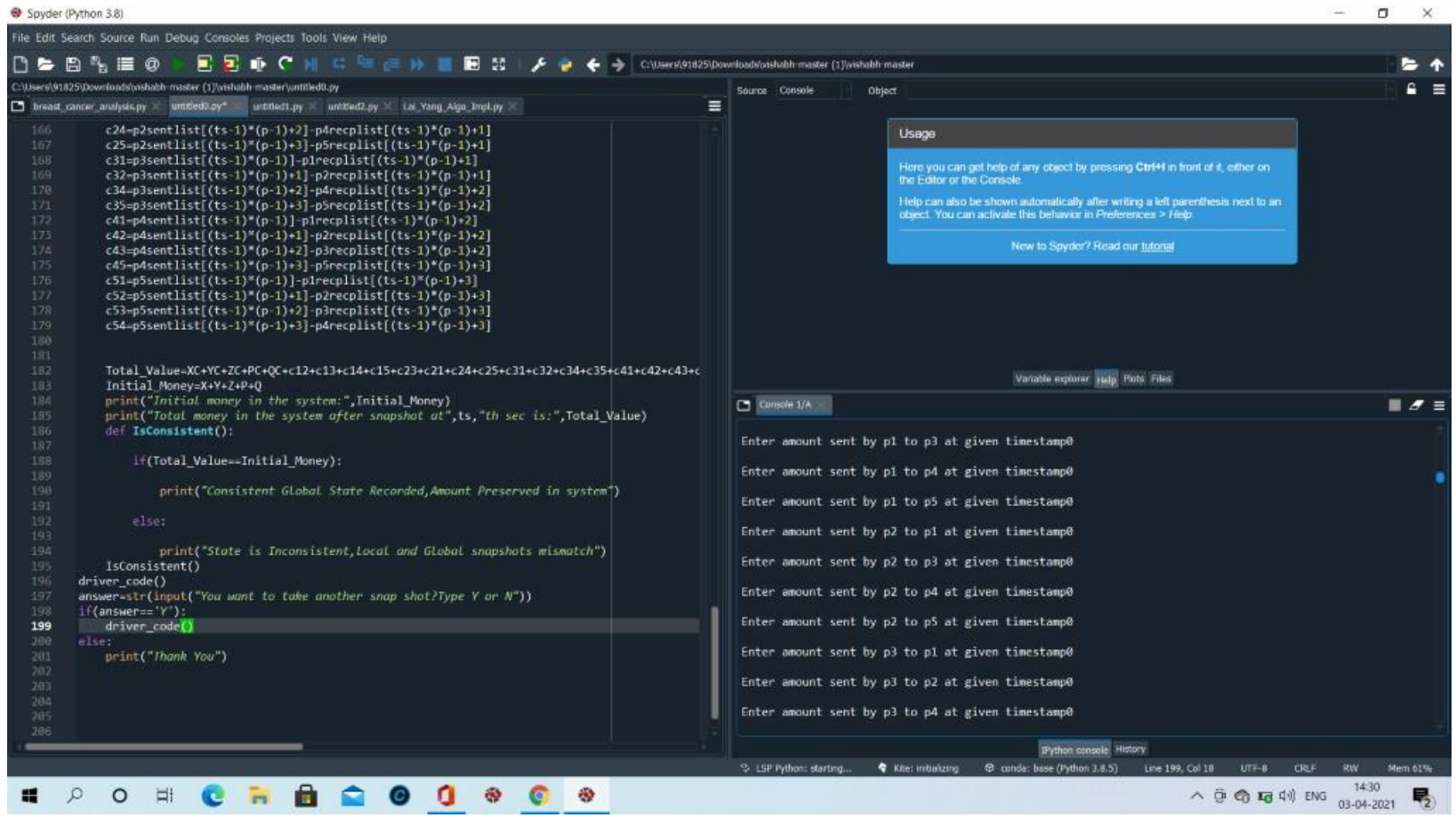
Assumption


Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
La Yang Li Algorithm Implementation and Correctness
Assumptions
1.The system can handle at most 5 processes/sites,Increasing the number of processes complicate the data structures being used
2.Any messages in transit should be handled during program input
3.Messages being sent between two consecutive timestamps are assumed to be sent in the next higher timestamp as per convention
4.The algorithm is primarily for NON-FIFO and causal ordering of messages are not maintained,however irrespective of FIFO and NON-FIFO Communication channel,it works because such complexities are handled in input
5.The program only deals with white messages as they are used to calculate channel states locally
6.Messages in transit are calculated in channel states within the program
7.White messages received after the first snapshot are ignored
8.The program can accomodate a single snapshot at a time and check for consistency.If snapshot is taken again, rerun the program and reset all messages as white
9.Program successfully checks if the snapshot being taken is consistent or not
10.The programmer assumes that for better clarity the end user has a logical State-Time diagram of the problem handy to compare results
For the logic behind implementation kindly follow the read me file

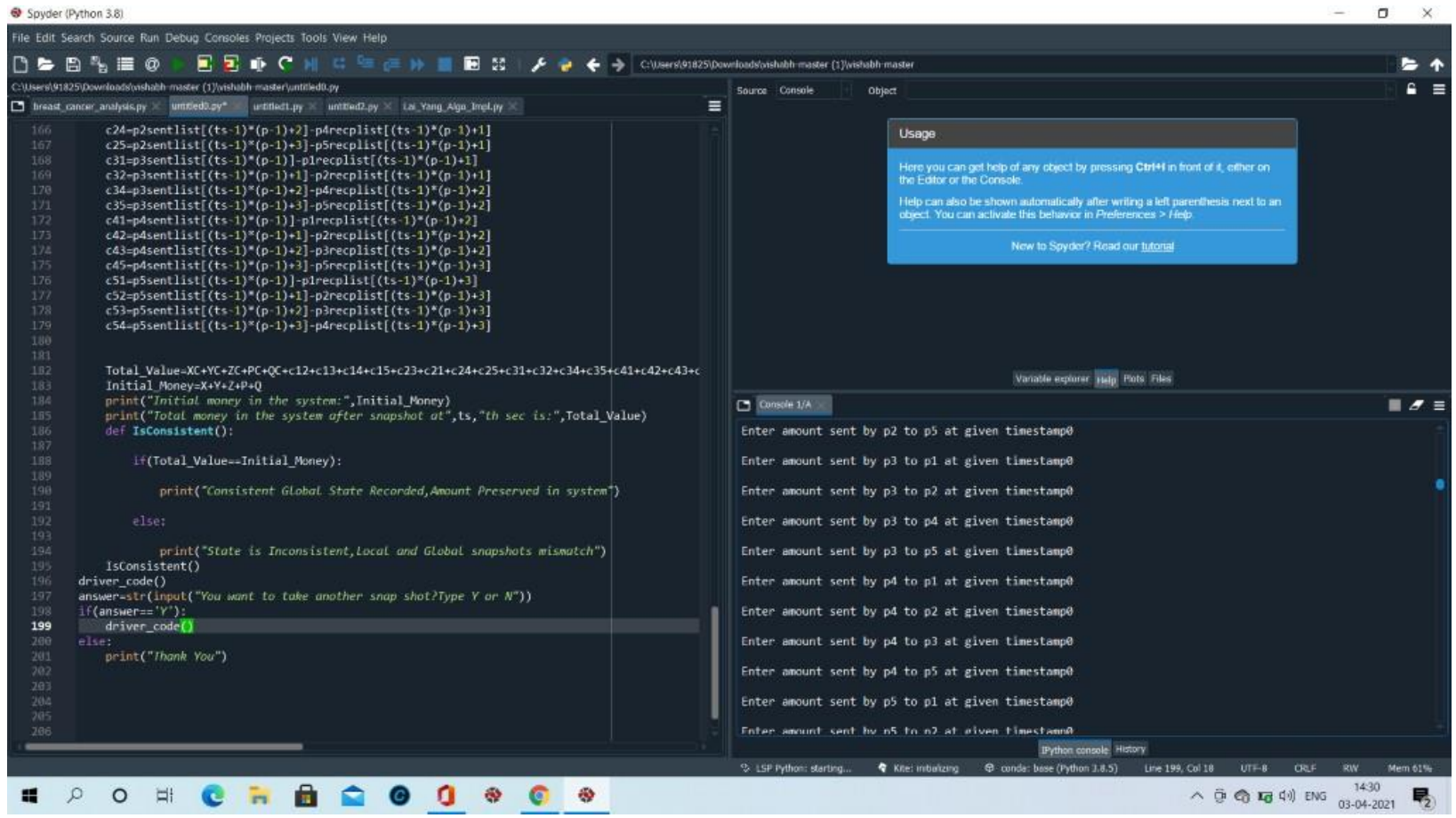
Inputting values



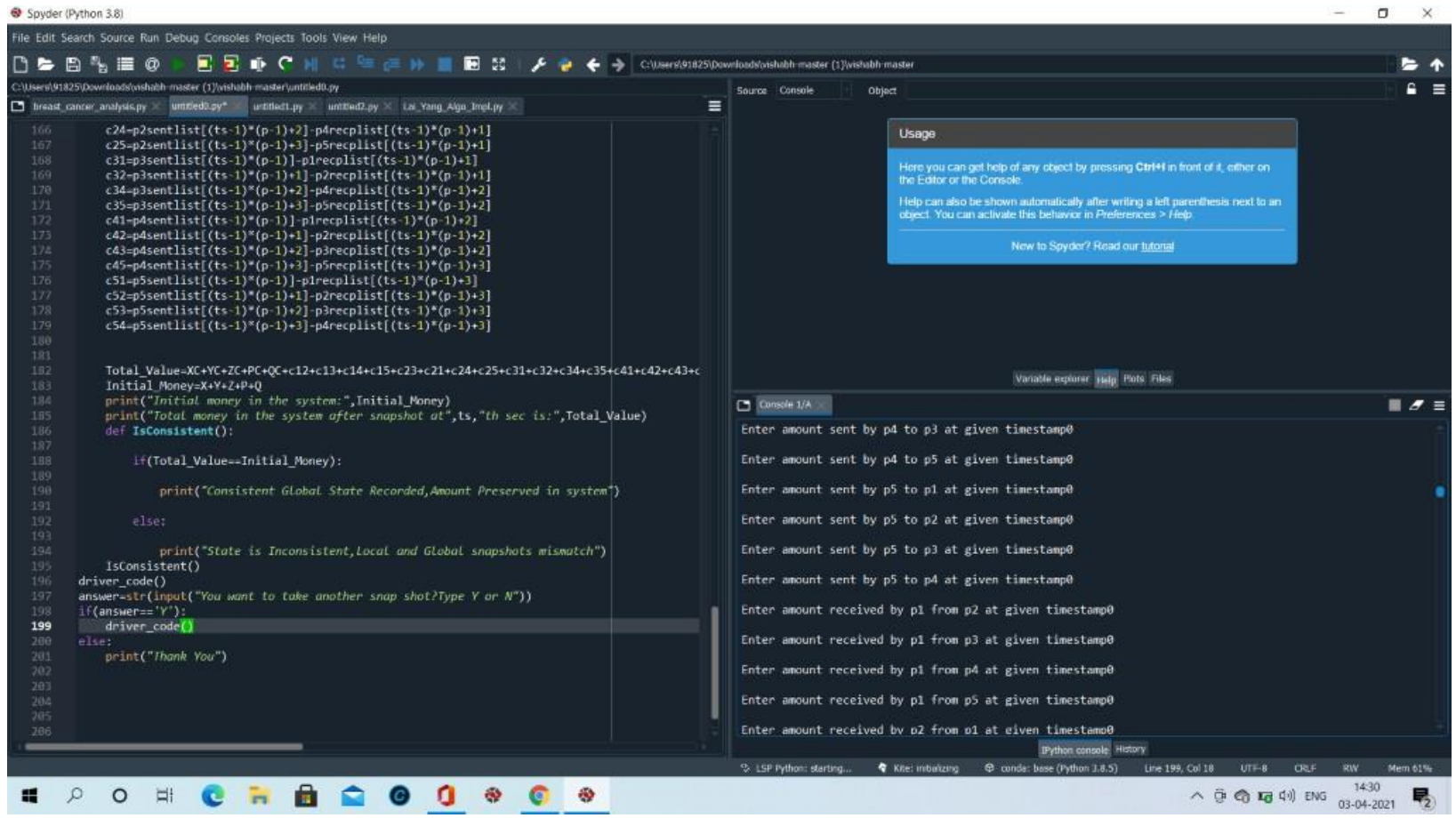
Inputting Values



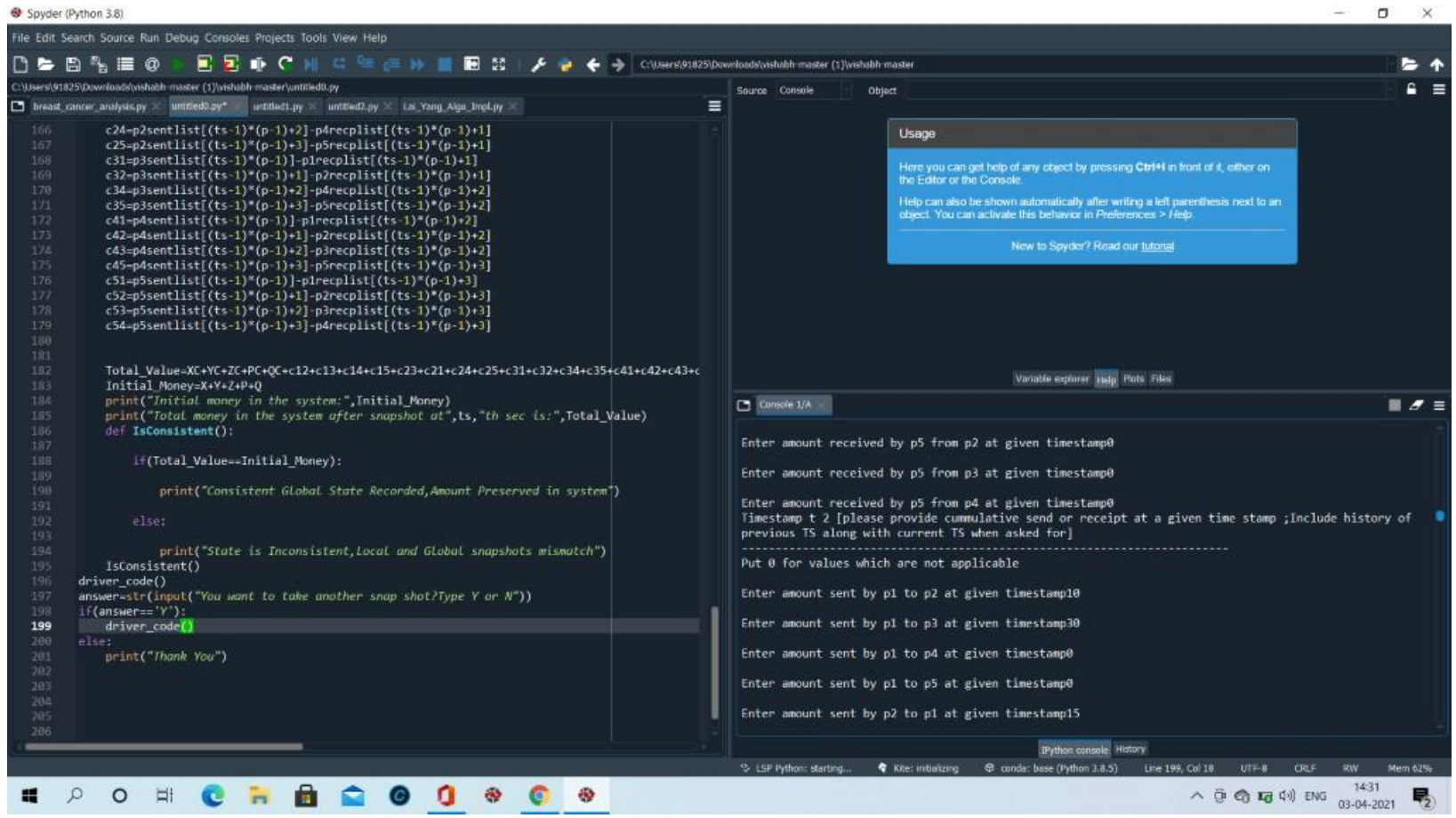
Inputting values



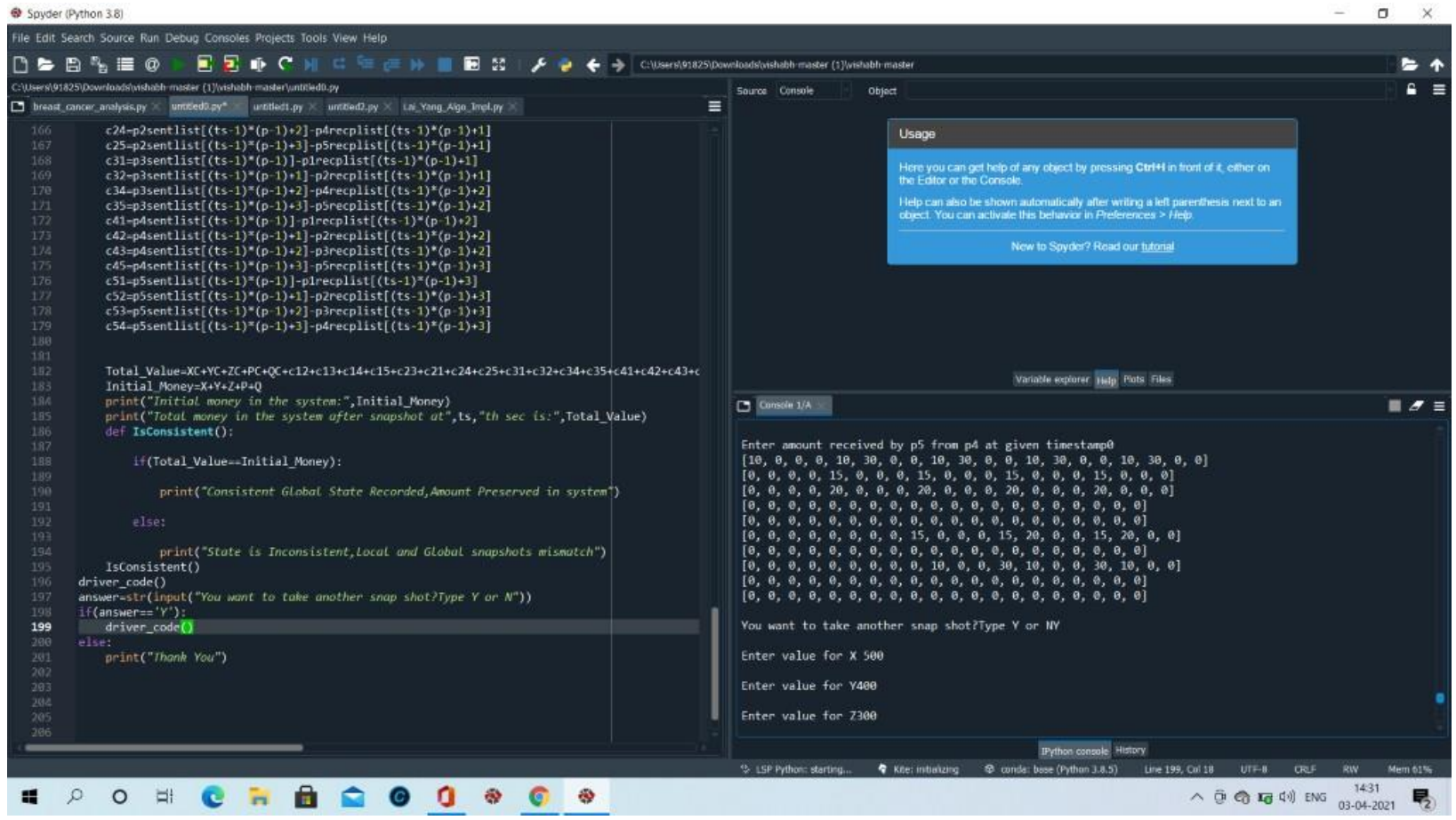
Inputting Values



Inputting values in next time stamp



Final array lists of all processes(both sent and receipt in respective lists, stored in different time stamps)



Working of the drivercode(processing)

- Since we are giving cumulative amounts for sent and receipt amounts for each process at a given timestamp and each process is sending and receiving from all other processes other than itself, the list indices are manipulated to calculate the channel states and amount present in each process during snapshot.
- We define 10 lists(2 for each process) namely p1sentlist, p1recplist, p2sentlist,p2recplist.....

For example to calculate total amount sent by P1 to all other processes across all time stamps,
totalp1sent=p1sentlist[(ts-1)*(p-1)]+p1sentlist[(ts-1)*(p-1)+1]+p1sentlist[(ts-1)*(p-1)+2]+p1sentlist[(ts-1)*(p-1)+3], where p=no. of processes and ts=timestamp of snap shot recording

Likewise other sent and receipt are calculated for each process.Total we have 10(2X5) variables, pertaining to 5 processes.

Calculating current values of X,Y,Z,P,Q

The current amount available with X,Y,Z,P,Q(during ts) are calculated as below.

XC=X+totalp1recp-totalp1sent

YC=Y+totalp2recp-totalp2sent

ZC=Z+totalp3recp-totalp3sent

PC=P+totalp4recp-totalp4sent(P is zero for our demo)

QC=Q+totalp5recp-totalp5sent(Q is zero for our demo)

Calculating Channel States

The channel states are calculated as per the algorithm step as white messages sent by P1 to P2 on C12 – white messages received by the process P2 from P1 across the channel C12.

c12=p1sentlist[(ts-1)*(p-1)]-p2recplist[(ts-1)*(p-1)]

Similarly all other channel states are calculated.

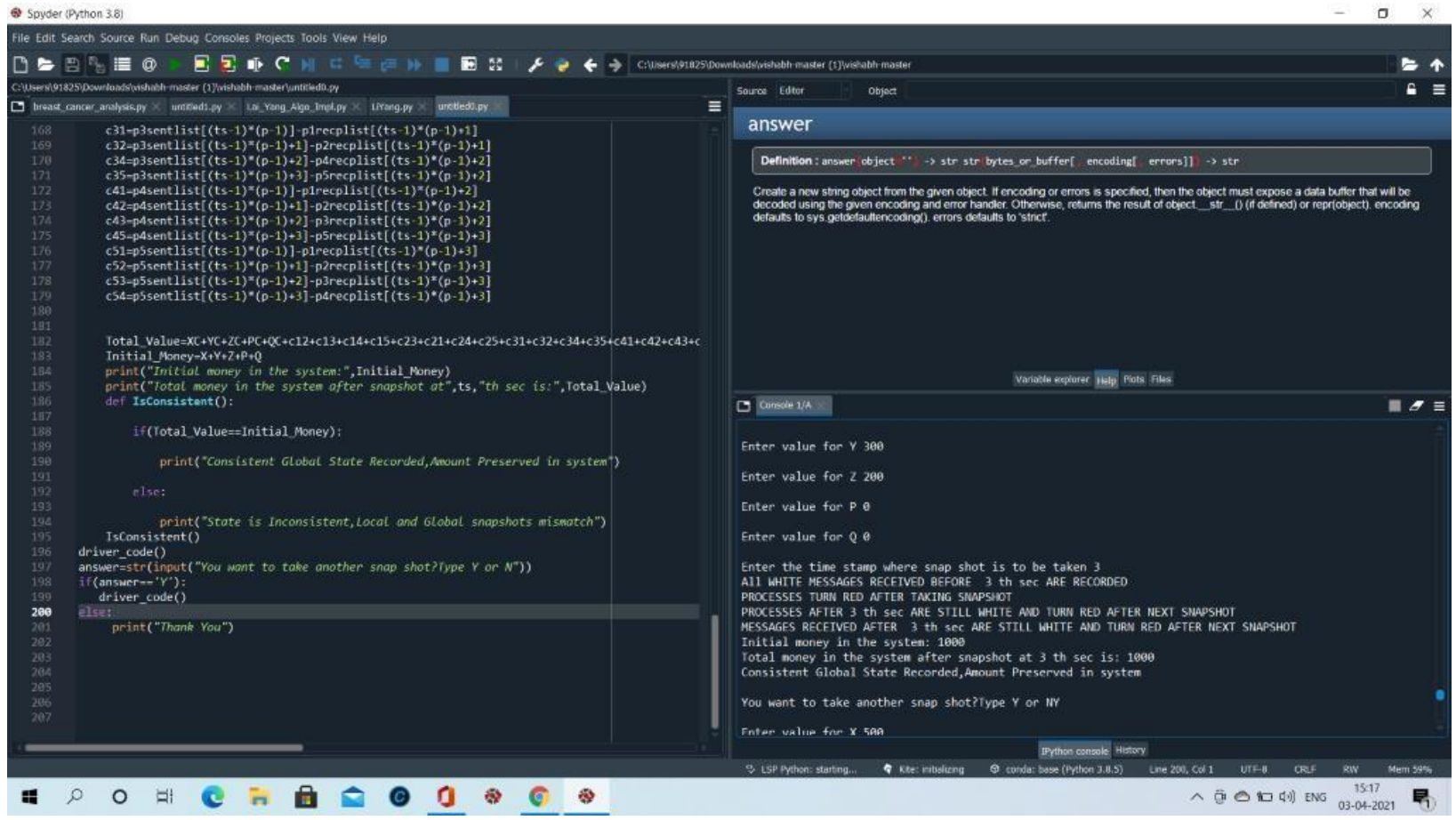
The local snap shot is calculated as

XC+YC+ZC+PC+QC+c12+c13+c14+c15+c23+c21+c24+c25+c31+c32+c34 +c35+c41+c42+c43+c45+c51+c52+c53+c54 and compared with X+Y+Z+P+Q , to check for consistency.

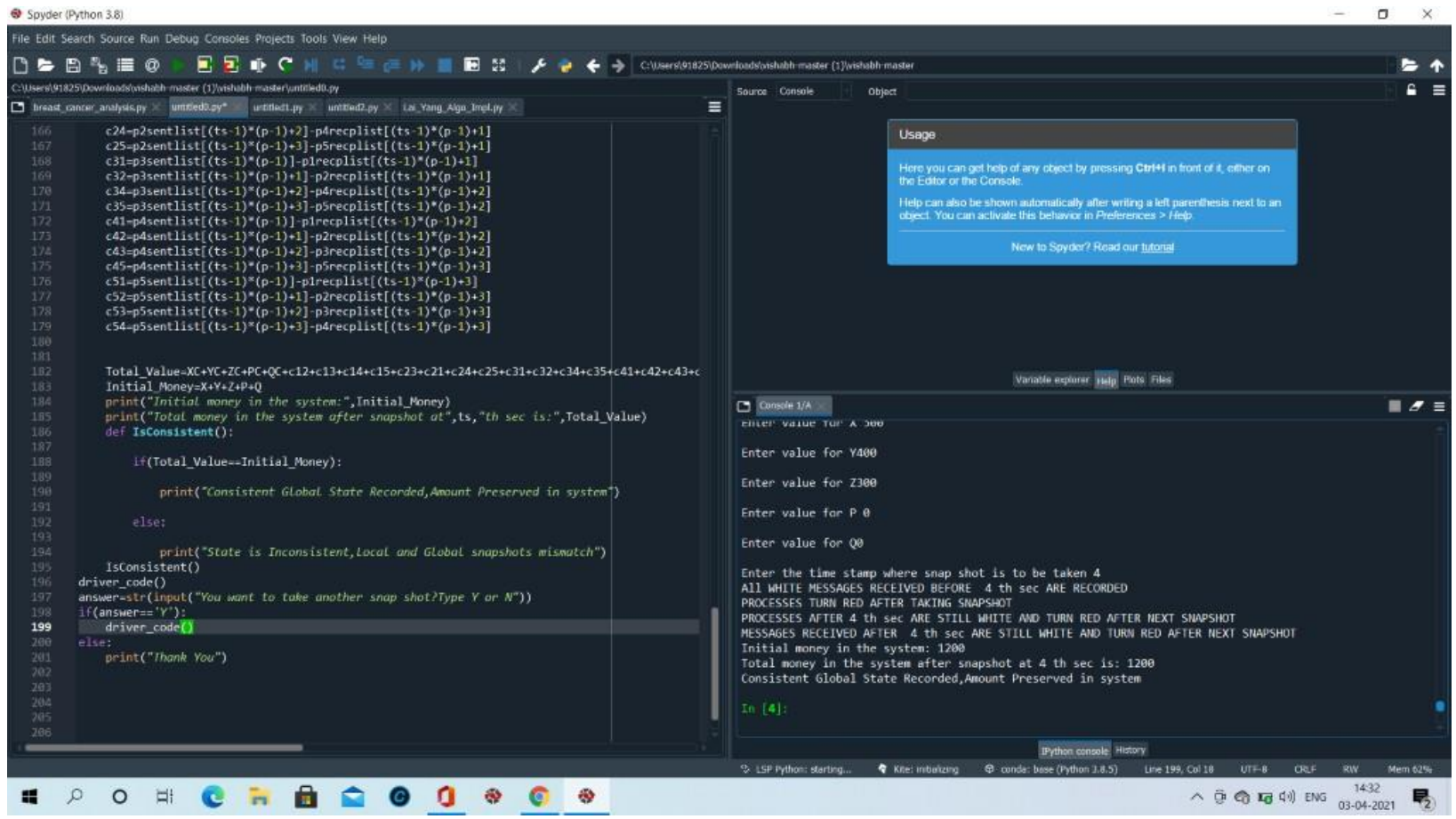
Output

- The program calculates channel states for each of the P(P-1) channel where p=no. of processes.
- It will Print, if the system has taken a consistent global snapshot at the given time stamp along with the message that the amount of money is preserved in the system.
- It will interactively ask, if user wants to take any other snap shots, if yes, it resets all messages to white and retakes the snap shot and check the consistency.

Calculating local snap shots and finding values for channel states and check for consistency



Taking another snapshot at some other interval



In [3]: runfile('C:/Users/91825/OneDrive/Desktop/DC_Assignment/LiYang.py', wdir='C:/Users/91825/OneDrive/Desktop/DC_Assignment')

La Yang Li Algorithm Implementation and Correctness

Assumptions

1.The system can handle at most 5 processes/sites,Increasing the number of processes complicate the data structures being used

2.Any messages in transit should be handled during program input

3.Messages being sent between two consecutive timestamps are assumed to be sent in the next higher timestamp as per convention

4.The algorithm is primarily for NON-FIFO and causal ordering of messages are not maintained,however irrespective of FIFO and NON-FIFO Communication channel,it works because such complexities are handled in input

5.The program only deals with white messages as they are used to calculate channel states locally

6.Messages in transit are calculated in channel states within the program

7.White messages received after the first snapshot are ignored

8.The program can accomodate a single snapshot at a time and check for consistency.If snapshot is taken again, rerun the program and reset all messages as white

9.Program successfully checks if the snapshot being taken is consistent or not

10.The programmer assumes that for better clarity the end user has a logical State-Time diagram of the problem handy to compare results

For the logic behind implementation kindly follow the read me file

---

Enter maximum number of process in the system as 5 5

Enter relevant number of processes 3

number of max processes 5

number of max channels 20

number of relevant processes 3

Number of relevant channels 6

Enter total num of timestamps 5

Please enter history of white processes preferably(not mandatory) till 5 th sec

Timestamp t 1 [please provide cummulative send or receipt at a given time stamp ;Include history of previous TS alo
ng with current TS when asked for]

Put 0 for values which are not applicable

For input values see read me file or prepare your own input as per instruction in read me file

Enter amount sent by p1 to p2 at given timestamp 10

Enter amount sent by p1 to p3 at given timestamp0

Enter amount sent by p1 to p4 at given timestamp0

Enter amount sent by p1 to p5 at given timestamp0

Enter amount sent by p2 to p1 at given timestamp0

Enter amount sent by p2 to p3 at given timestamp0

Enter amount sent by p2 to p4 at given timestamp0

Enter amount sent by p2 to p5 at given timestamp0

Enter amount sent by p3 to p1 at given timestamp0

Enter amount sent by p3 to p2 at given timestamp0

Enter amount sent by p3 to p4 at given timestamp0

Enter amount sent by p3 to p5 at given timestamp0

Enter amount sent by p4 to p1 at given timestamp0

Enter amount sent by p4 to p2 at given timestamp0

Enter amount sent by p4 to p3 at given timestamp0

Enter amount sent by p4 to p5 at given timestamp0

Enter amount sent by p5 to p1 at given timestamp0

Enter amount sent by p5 to p2 at given timestamp0

Enter amount sent by p5 to p3 at given timestamp0

Enter amount sent by p5 to p4 at given timestamp0

Enter amount received by p1 from p2 at given timestamp0

Enter amount received by p1 from p3 at given timestamp0

Enter amount received by p1 from p4 at given timestamp0

Enter amount received by p1 from p5 at given timestamp0

Enter amount received by p2 from p1 at given timestamp0

Enter amount received by p2 from p3 at given timestamp0

Enter amount received by p2 from p4 at given timestamp0

Enter amount received by p2 from p5 at given timestamp0

Enter amount received by p3 from p1 at given timestamp0

Enter amount received by p3 from p2 at given timestamp0

Enter amount received by p3 from p4 at given timestamp0

Enter amount received by p3 from p5 at given timestamp0

Enter amount received by p4 from p1 at given timestamp0

Enter amount received by p4 from p2 at given timestamp0

Enter amount received by p4 from p3 at given timestamp0

Enter amount received by p4 from p5 at given timestamp0

Enter amount received by p5 from p1 at given timestamp0

Enter amount received by p5 from p2 at given timestamp0

Enter amount received by p5 from p3 at given timestamp0

Enter amount received by p5 from p4 at given timestamp0

Timestamp t 2 [please provide cummulative send or receipt at a given time stamp ;Include history of previous TS alo ng with current TS when asked for]

Put 0 for values which are not applicable

For input values see read me file or prepare your own input as per instruction in read me file

Enter amount sent by p1 to p2 at given timestamp10

Enter amount sent by p1 to p3 at given timestamp30

Enter amount sent by p1 to p4 at given timestamp0

Enter amount sent by p1 to p5 at given timestamp0

Enter amount sent by p2 to p1 at given timestamp15

Enter amount sent by p2 to p3 at given timestamp10

Enter amount sent by p2 to p4 at given timestamp0

Enter amount sent by p2 to p5 at given timestamp0

Enter amount sent by p3 to p1 at given timestamp20

Enter amount sent by p3 to p2 at given timestamp0

Enter amount sent by p3 to p4 at given timestamp0

Enter amount sent by p3 to p5 at given timestamp0

Enter amount sent by p4 to p1 at given timestamp0

Enter amount sent by p4 to p2 at given timestamp0

Enter amount sent by p4 to p3 at given timestamp0

Enter amount sent by p4 to p5 at given timestamp0

Enter amount sent by p5 to p1 at given timestamp0

Enter amount sent by p5 to p2 at given timestamp0

Enter amount sent by p5 to p3 at given timestamp0

Enter amount sent by p5 to p4 at given timestamp0

Enter amount received by p1 from p2 at given timestamp0

Enter amount received by p1 from p3 at given timestamp0

Enter amount received by p1 from p4 at given timestamp0

Enter amount received by p1 from p5 at given timestamp0

Enter amount received by p2 from p1 at given timestamp10

Enter amount received by p2 from p3 at given timestamp0

Enter amount received by p2 from p4 at given timestamp0

Enter amount received by p2 from p5 at given timestamp0

Enter amount received by p3 from p1 at given timestamp0

Enter amount received by p3 from p2 at given timestamp0

Enter amount received by p3 from p4 at given timestamp0

Enter amount received by p3 from p5 at given timestamp0

Enter amount received by p4 from p1 at given timestamp0

Enter amount received by p4 from p2 at given timestamp0

Enter amount received by p4 from p3 at given timestamp0

Enter amount received by p4 from p5 at given timestamp0

Enter amount received by p5 from p1 at given timestamp0

Enter amount received by p5 from p2 at given timestamp0

Enter amount received by p5 from p3 at given timestamp0

Enter amount received by p5 from p4 at given timestamp0

Timestamp t 3 [please provide cummulative send or receipt at a given time stamp ;Include history of previous TS alo ng with current TS when asked for]

Put 0 for values which are not applicable

For input values see read me file or prepare your own input as per instruction in read me file

Enter amount sent by p1 to p2 at given timestamp10

Enter amount sent by p1 to p3 at given timestamp30

Enter amount sent by p1 to p4 at given timestamp0

Enter amount sent by p1 to p5 at given timestamp0

Enter amount sent by p2 to p1 at given timestamp15

Enter amount sent by p2 to p3 at given timestamp10

Enter amount sent by p2 to p4 at given timestamp0

Enter amount sent by p2 to p5 at given timestamp0

Enter amount sent by p3 to p1 at given timestamp20

Enter amount sent by p3 to p2 at given timestamp0

Enter amount sent by p3 to p4 at given timestamp0

Enter amount sent by p3 to p5 at given timestamp0

Enter amount sent by p4 to p1 at given timestamp0

Enter amount sent by p4 to p2 at given timestamp0

Enter amount sent by p4 to p3 at given timestamp0

Enter amount sent by p4 to p5 at given timestamp0

Enter amount sent by p5 to p1 at given timestamp0

Enter amount sent by p5 to p2 at given timestamp0

Enter amount sent by p5 to p3 at given timestamp0

Enter amount sent by p5 to p4 at given timestamp0

Enter amount received by p1 from p2 at given timestamp15

Enter amount received by p1 from p3 at given timestamp0

Enter amount received by p1 from p4 at given timestamp0

Enter amount received by p1 from p5 at given timestamp0

Enter amount received by p2 from p1 at given timestamp10

Enter amount received by p2 from p3 at given timestamp0

Enter amount received by p2 from p4 at given timestamp0

Enter amount received by p2 from p5 at given timestamp0

Enter amount received by p3 from p1 at given timestamp0

Enter amount received by p3 from p2 at given timestamp10

Enter amount received by p3 from p4 at given timestamp0

Enter amount received by p3 from p5 at given timestamp0

Enter amount received by p4 from p1 at given timestamp0

Enter amount received by p4 from p2 at given timestamp0

Enter amount received by p4 from p3 at given timestamp0

Enter amount received by p4 from p5 at given timestamp0

Enter amount received by p5 from p1 at given timestamp0

Enter amount received by p5 from p2 at given timestamp0

Enter amount received by p5 from p3 at given timestamp0

Enter amount received by p5 from p4 at given timestamp0

Timestamp t 4 [please provide cummulative send or receipt at a given time stamp ;Include history of previous TS alo ng with current TS when asked for]

Put 0 for values which are not applicable

For input values see read me file or prepare your own input as per instruction in read me file

Enter amount sent by p1 to p2 at given timestamp10

Enter amount sent by p1 to p3 at given timestamp30

Enter amount sent by p1 to p4 at given timestamp0

Enter amount sent by p1 to p5 at given timestamp0

Enter amount sent by p2 to p1 at given timestamp15

Enter amount sent by p2 to p3 at given timestamp10

Enter amount sent by p2 to p4 at given timestamp0

Enter amount sent by p2 to p5 at given timestamp0

Enter amount sent by p3 to p1 at given timestamp20

Enter amount sent by p3 to p2 at given timestamp0

Enter amount sent by p3 to p4 at given timestamp0

Enter amount sent by p3 to p5 at given timestamp0

Enter amount sent by p4 to p1 at given timestamp0

Enter amount sent by p4 to p2 at given timestamp0

Enter amount sent by p4 to p3 at given timestamp0

Enter amount sent by p4 to p5 at given timestamp0

Enter amount sent by p5 to p1 at given timestamp0

Enter amount sent by p5 to p2 at given timestamp0

Enter amount sent by p5 to p3 at given timestamp0

Enter amount sent by p5 to p4 at given timestamp0

Enter amount received by p1 from p2 at given timestamp15

Enter amount received by p1 from p3 at given timestamp20

Enter amount received by p1 from p4 at given timestamp0

Enter amount received by p1 from p5 at given timestamp0

Enter amount received by p2 from p1 at given timestamp10

Enter amount received by p2 from p3 at given timestamp0

Enter amount received by p2 from p4 at given timestamp0

Enter amount received by p2 from p5 at given timestamp0

Enter amount received by p3 from p1 at given timestamp30

Enter amount received by p3 from p2 at given timestamp10

Enter amount received by p3 from p4 at given timestamp0

Enter amount received by p3 from p5 at given timestamp0

Enter amount received by p4 from p1 at given timestamp0

Enter amount received by p4 from p2 at given timestamp0

Enter amount received by p4 from p3 at given timestamp0

Enter amount received by p4 from p5 at given timestamp0

Enter amount received by p5 from p1 at given timestamp0

Enter amount received by p5 from p2 at given timestamp0

Enter amount received by p5 from p3 at given timestamp0

Enter amount received by p5 from p4 at given timestamp0

Timestamp t 5 [please provide cummulative send or receipt at a given time stamp ;Include history of previous TS alo
ng with current TS when asked for]

Put 0 for values which are not applicable

For input values see read me file or prepare your own input as per instruction in read me file

Enter amount sent by p1 to p2 at given timestamp10

Enter amount sent by p1 to p3 at given timestamp30

Enter amount sent by p1 to p4 at given timestamp0

Enter amount sent by p1 to p5 at given timestamp0

Enter amount sent by p2 to p1 at given timestamp15

Enter amount sent by p2 to p3 at given timestamp10

Enter amount sent by p2 to p4 at given timestamp0

Enter amount sent by p2 to p5 at given timestamp0

Enter amount sent by p3 to p1 at given timestamp20

Enter amount sent by p3 to p2 at given timestamp0

Enter amount sent by p3 to p4 at given timestamp0

Enter amount sent by p3 to p5 at given timestamp0

Enter amount sent by p4 to p1 at given timestamp0

Enter amount sent by p4 to p2 at given timestamp0

Enter amount sent by p4 to p3 at given timestamp0

Enter amount sent by p4 to p5 at given timestamp0

Enter amount sent by p5 to p1 at given timestamp0

Enter amount sent by p5 to p2 at given timestamp0

Enter amount sent by p5 to p3 at given timestamp0

Enter amount sent by p5 to p4 at given timestamp0

Enter amount received by p1 from p2 at given timestamp15

Enter amount received by p1 from p3 at given timestamp20

Enter amount received by p1 from p4 at given timestamp0

Enter amount received by p1 from p5 at given timestamp0

Enter amount received by p2 from p1 at given timestamp10

Enter amount received by p2 from p3 at given timestamp0

Enter amount received by p2 from p4 at given timestamp0

Enter amount received by p2 from p5 at given timestamp0

Enter amount received by p3 from p1 at given timestamp30

Enter amount received by p3 from p2 at given timestamp10

Enter amount received by p3 from p4 at given timestamp0

Enter amount received by p3 from p5 at given timestamp0

Enter amount received by p4 from p1 at given timestamp0

Enter amount received by p4 from p2 at given timestamp0

Enter amount received by p4 from p3 at given timestamp0

Enter amount received by p4 from p5 at given timestamp0

Enter amount received by p5 from p1 at given timestamp0

Enter amount received by p5 from p2 at given timestamp0

Enter amount received by p5 from p3 at given timestamp0

Enter amount received by p5 from p4 at given timestamp0

[10, 0, 0, 0, 10, 30, 0, 0, 10, 30, 0, 0, 10, 30, 0, 0, 10, 30, 0, 0]

[0, 0, 0, 0, 15, 10, 0, 0, 15, 10, 0, 0, 15, 10, 0, 0, 15, 10, 0, 0]

[0, 0, 0, 0, 20, 0, 0, 0, 20, 0, 0, 0, 20, 0, 0, 0, 20, 0, 0, 0]

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

[0, 0, 0, 0, 0, 0, 0, 0, 15, 0, 0, 0, 15, 20, 0, 0, 15, 20, 0, 0]

[0, 0, 0, 0, 10, 0, 0, 0, 10, 0, 0, 0, 10, 0, 0, 0, 10, 0, 0, 0]

[0, 0, 0, 0, 0, 0, 0, 0, 0, 10, 0, 0, 30, 10, 0, 0, 30, 10, 0, 0]

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Enter value for X 500

Enter value for Y 400

Enter value for Z 300

Enter value for P 0

Enter value for Q 0

Enter the time stamp where snap shot is to be taken 3

All WHITE MESSAGES RECEIVED BEFORE 3 th sec ARE RECORDED

PROCESSES TURN RED AFTER TAKING SNAPSHOT

PROCESSES AFTER 3 th sec ARE STILL WHITE AND TURN RED AFTER NEXT SNAPSHOT

MESSAGES RECEIVED AFTER 3 th sec ARE STILL WHITE AND TURN RED AFTER NEXT SNAPSHOT

Value of C12: 0

Value of C13: 30

Value of C14: 0

Value of C15: 0

Value of C21: 0

Value of C23: 0

Value of C24: 0

Value of C25: 0

Value of C31: 20

Value of C32: 0

Value of C34: 0

Value of C35: 0

Value of C41: 0

Value of C42: 0

Value of C43: 0

Value of C45: 0

Value of C51: 0

Value of C52: 0

Value of C53: 0

Value of C54: 0

----Value of X,Y,Z,P and Q at the time of snapshot----------

Current value of X is: 475

Current value of Y is: 385

Current value of Z is: 290

Current value of P is: 0

Current value of Q is: 0

Initial money in the system: 1200

Total money in the system after snapshot at 3 th sec is: 1200

Consistent Global State Recorded,Amount Preserved in system

You want to take another snap shot?Type Y or NY

Enter value for X 500

Enter value for Y 400

Enter value for Z 300

Enter value for P 0

Enter value for Q 0

Enter the time stamp where snap shot is to be taken 4

All WHITE MESSAGES RECEIVED BEFORE 4 th sec ARE RECORDED

PROCESSES TURN RED AFTER TAKING SNAPSHOT

PROCESSES AFTER 4 th sec ARE STILL WHITE AND TURN RED AFTER NEXT SNAPSHOT

MESSAGES RECEIVED AFTER 4 th sec ARE STILL WHITE AND TURN RED AFTER NEXT SNAPSHOT

Value of C12: 0

Value of C13: 0

Value of C14: 0

Value of C15: 0

Value of C21: 0

Value of C23: 0

Value of C24: 0

Value of C25: 0

Value of C31: 0

Value of C32: 0

Value of C34: 0

Value of C35: 0

Value of C41: 0

Value of C42: 0

Value of C43: 0

Value of C45: 0

Value of C51: 0

Value of C52: 0

Value of C53: 0

Value of C54: 0

----Value of X,Y,Z,P and Q at the time of snapshot----------

Current value of X is: 495

Current value of Y is: 385

Current value of Z is: 320

Current value of P is: 0

Current value of Q is: 0

Initial money in the system: 1200

Total money in the system after snapshot at 4 th sec is: 1200

Consistent Global State Recorded,Amount Preserved in system

In [2]: