Name: Srujan R
Roll No: BT18CSE041

CN Assignment: 4

Details about files :

- BT18CSE041_dvr.py : contains the main program
- ex1.txt : contains the first test case
- ex2.txt : contains the second test case

Working of the code :

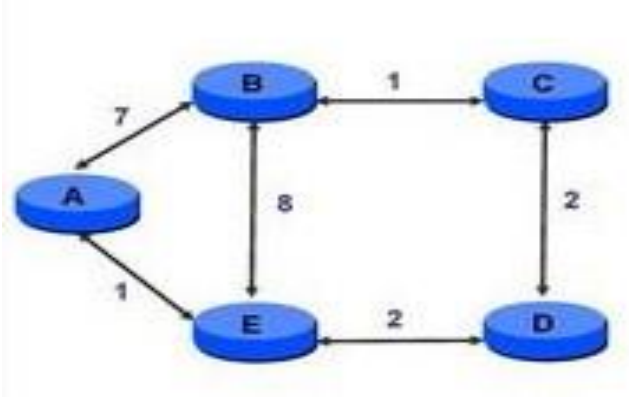
1. The main thread is responsible for printing a global dictionary(tables) containing the distance table of all routers and does this 5 times (initial condition and 4 rounds).
2. The threads are made to run the function calculateDistances() which has the implementation of DVR algorithm and each thread accesses a specific router object from router_object_list and updates that router's distance table, the global dictionary table and goes to sleep for 2 seconds.
3. The threads can run the function only when its router object's queue has the adjacent routers' distance table, so it waits till the queue is filled by the other threads.
4. New threads are created for each router object, in every iteration in the main thread and the main thread waits till all threads finish, to print the global dictionary(tables).

To execute :

- python BT18CSE041_dvr.py (input file-path).
    - Ex : to execute first testcase file -> python BT18CSE041_dvr.py ex1.txt

Test Cases and their Output Snippets :

- **5 , [A B C D E] , [A B 7, A E 1, B E 8, B C 1, C D 2, D E 2**



ITERATION -> 0

ROUTER A

Node	Distance	Next Hop
B	7	B
C	INF	NAN
D	INF	NAN
E	1	E

ROUTER B

Node	Distance	Next Hop
A	7	A
C	1	C
D	INF	NAN
E	8	E

ROUTER C

Node	Distance	Next Hop
A	INF	NAN
B	1	B
D	2	D
E	INF	NAN

ROUTER D

Node	Distance	Next Hop
A	INF	NAN
B	INF	NAN
C	2	C
E	2	E

ROUTER E

Node	Distance	Next Hop
A	1	A
B	8	B
C	INF	NAN
D	2	D

ITERATION -> 1

ROUTER A

Node	Distance	Next Hop
B	7	B
C	8 *	B *
D	3 *	E *
E	1	E

ROUTER B

Node	Distance	Next Hop
A	7	A
C	1	C
D	3 *	C *
E	8	E

ROUTER C

Node	Distance	Next Hop
A	8 *	B *
B	1	B
D	2	D
E	4 *	D *

ROUTER D

Node	Distance	Next Hop
A	3 *	E *
B	3 *	C *
C	2	C
E	2	E

ROUTER E

Node	Distance	Next Hop
A	1	A
B	5 *	D *
C	4 *	D *
D	2	D

ITERATION -> 2

ROUTER A

Node	Distance	Next Hop
B	6 *	E *
C	5 *	E *
D	3	E
E	1	E

ROUTER B

Node	Distance	Next Hop
A	7	A
C	1	C
D	3 *	C *
E	5 *	C *

ROUTER C

Node	Distance	Next Hop
A	5 *	D *
B	1	B
D	2	D
E	4 *	D *

ROUTER D

Node	Distance	Next Hop
A	3 *	E *
B	3	C
C	2	C
E	2	E

ROUTER E

Node	Distance	Next Hop
A	1	A
B	5	D
C	4 *	D *
D	2	D

ITERATION -> 3

ROUTER A

Node	Distance	Next Hop
B	6	E
C	5 *	E *
D	3	E
E	1	E

ROUTER B

Node	Distance	Next Hop
A	6 *	C *
C	1	C
D	3	C
E	5	C

ROUTER C

Node	Distance	Next Hop
A	5	D
B	1	B
D	2	D
E	4	D

ROUTER D

Node	Distance	Next Hop
A	3	E
B	3	C
C	2	C
E	2	E

ROUTER E

Node	Distance	Next Hop
A	1	A
B	5	D
C	4 *	D *
D	2	D

ITERATION -> 4

ROUTER A

Node	Distance	Next Hop
B	6	E
C	5	E
D	3	E
E	1	E

ROUTER B

Node	Distance	Next Hop
A	6	C
C	1	C
D	3	C
E	5	C

ROUTER C

Node	Distance	Next Hop
A	5	D
B	1	B
D	2	D
E	4	D

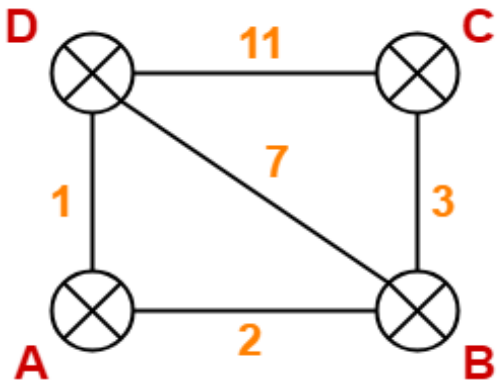
ROUTER D

Node	Distance	Next Hop
A	3	E
B	3	C
C	2	C
E	2	E

ROUTER E

Node	Distance	Next Hop
A	1	A
B	5	D
C	4	D
D	2	D

- **4, [A B C D], [A D 1, A B 2, B D 7, B C 3, C D 11]**



ITERATION -> 0

ROUTER A

Node	Distance	Next Hop
B	2	B
C	INF	NAN
D	1	D

ROUTER B

Node	Distance	Next Hop
A	2	A
C	3	C
D	7	D

ROUTER C

Node	Distance	Next Hop
A	INF	NAN
B	3	B
D	11	D

ROUTER D

Node	Distance	Next Hop
A	1	A
B	7	B
C	11	C

ITERATION -> 1

ROUTER A

Node	Distance	Next Hop
B	2	B
C	5 *	B *
D	1	D

ROUTER B

Node	Distance	Next Hop
A	2	A
C	3	C
D	3 *	A *

ROUTER C

Node	Distance	Next Hop
A	5 *	B *
B	3	B
D	6 *	B *

ROUTER D

Node	Distance	Next Hop
A	1	A
B	3 *	A *
C	6 *	A *

ITERATION -> 2

ROUTER A

Node	Distance	Next Hop
B	2	B
C	5 *	B *
D	1	D

ROUTER B

Node	Distance	Next Hop
A	2	A
C	3	C
D	3	A

ROUTER C

Node	Distance	Next Hop
A	5	B
B	3	B
D	6	B

ROUTER D

Node	Distance	Next Hop
A	1	A
B	3	A
C	6	A

ITERATION -> 3

ROUTER A

Node	Distance	Next Hop
B	2	B
C	5	B
D	1	D

ROUTER B

Node	Distance	Next Hop
A	2	A
C	3	C
D	3	A

ROUTER C

Node	Distance	Next Hop
A	5	B
B	3	B
D	6	B

ROUTER D

Node	Distance	Next Hop
A	1	A
B	3	A
C	6	A

ITERATION -> 4

ROUTER A

Node	Distance	Next Hop
B	2	B
C	5	B
D	1	D

ROUTER B

Node	Distance	Next Hop
A	2	A
C	3	C
D	3	A

ROUTER C

Node	Distance	Next Hop
A	5	B
B	3	B
D	6	B

ROUTER D

Node	Distance	Next Hop
A	1	A
B	3	A
C	6	A