

Distance Vector Routing Protocol

Name: Kaushal Lodd

Roll No: BT19CSE052

Instructions to run the code

- Download the zip folder of the codebase and unzip.
 - Open terminal and change directory to that of the codebase folder.
 - type `python dvr.py <input_filename.txt>` in the terminal to run the code.
 - There are two input files, namely `input1.txt` and `input2.txt`
-

Working of the code

- Designed and implemented a `Router` class which contains attributes and methods like -
 - Name of the Router
 - List of Neighbouring Routers
 - Routing Table
 - The Routing Table of each router contains all the destination routers, the minimum cost required to route to the destination routers and the path via which it routes to the destination routers.
 - Shared queue (containing routing tables from other routers)
 - `queue_lock` to acquire and release the lock for the shared queue
- The `input_parser` function parses the input text file and creates Router instances and assigns cost to the links between the routers. It implements the Initialization phase of the DV (Distance Vector) Algorithm.
- The `threaded` function is the function invoked when a thread is created for each instance of a router. This method invokes the `add_to_queue` method which shares

the current router's routing table to all other routers. The `threaded` function then waits for the current router to receive the routing tables from other routers'.

- After storing the router tables of other routers in `Router.queue`, we will apply Bellman-Ford's equation to calculate least-costly path from the current router to all other routers in the network.
 - We then update the current routing table according to Bellman-Ford's equation and wait for 2 seconds before running the next iteration of the algorithm on each router.
 - **Note:** Each iteration of the algorithm is run on each router concurrently (parallelly) using multiple threads.
 - Finally, over multiple iterations (in our case 4), the least-cost required to route to all the other routers is calculated and stored in the Routing Table.
-

Testing

- The DV Algorithm was tested on two input files, namely `input1.txt` and `input2.txt`
- `input1.txt` contains three routers with two links and is a fairly simple network. This was the example network provided to us in the Problem Statement. The algorithm calculates the least-cost path for all routers to all other routers in two iterations.
- The below screenshots show the Routing Table of each Router in various iterations.

output.txt – DVR Assignment

```

EXPLORER      dvr.py      output.txt
> input2 ss
DVR Assignment.pdf
dvr.py
figure.txt
input1.txt
input2.txt
output.txt

1  **** Iteration 0 *****
2
3
4 A => Router Object
5 Name: A
6 Neighbours: ['B', 'C']
7 Routing Table: {
8     A: 0          via: None
9     B: 5          via: B
10    C: 2          via: C
11 }
12
13 B => Router Object
14 Name: B
15 Neighbours: ['A']
16 Routing Table: {
17     A: 5          via: A
18     B: 0          via: None
19     C: inf        via: [no path]
20 }
21
22 C => Router Object
23 Name: C
24 Neighbours: ['A']
25 Routing Table: {
26     A: 2          via: A
27     B: inf        via: [no path]
28     C: 0          via: None
29 }
30
31 **** Iteration 1 *****
32
33 A => Router Object
34 Name: A
35 Neighbours: ['B', 'C']
36 Routing Table: {
37     A: 0          via: None
38     B: 5          via: B
39     C: 2          via: C
40 }
41
42
43 B => Router Object
44 Name: B
45 Neighbours: ['A']
46 Routing Table: {
47     A: 5          via: A
48     B: 0          via: None
49     *C: 7         via: A
50 }
51
52 C => Router Object
53 Name: C
54 Neighbours: ['A']
55 Routing Table: {
56     A: 2          via: A
57     *B: 7         via: A
58     C: 0          via: None
59 }
60
61 **** Iteration 2 *****
62
63 A => Router Object
64 Name: A
65 Neighbours: ['B', 'C']
66 Routing Table: {
67     A: 0          via: None
68     B: 5          via: B
69     C: 2          via: C
70 }
71
72 **** Iteration 3 *****
73
74 A => Router Object
75 Name: A
76 Neighbours: ['B', 'C']
77 Routing Table: {
78     A: 0          via: None
79     B: 5          via: B
80     C: 2          via: C
81 }
82
83 **** Iteration 4 *****
84
85 A => Router Object
86 Name: A
87 Neighbours: ['B', 'C']
88 Routing Table: {
89     A: 0          via: None
90     B: 5          via: B
91     C: 2          via: C
92 }
93
94 **** Iteration 5 *****
95
96 A => Router Object
97 Name: A
98 Neighbours: ['B', 'C']
99 Routing Table: {
100    A: 0         via: None
101    B: 5         via: B
102    C: 2         via: C
103 }
104
105 **** Iteration 6 *****
106
107 A => Router Object
108 Name: A
109 Neighbours: ['B', 'C']
110 Routing Table: {
111     A: 0         via: None
112     B: 5         via: B
113     C: 2         via: C
114 }
115
116 **** Iteration 7 *****
117
118 A => Router Object
119 Name: A
120 Neighbours: ['B', 'C']
121 Routing Table: {
122     A: 0         via: None
123     B: 5         via: B
124     C: 2         via: C
125 }
126
127 **** Iteration 8 *****
128
129 A => Router Object
130 Name: A
131 Neighbours: ['B', 'C']
132 Routing Table: {
133     A: 0         via: None
134     B: 5         via: B
135     C: 2         via: C
136 }
137
138 **** Iteration 9 *****
139
140 A => Router Object
141 Name: A
142 Neighbours: ['B', 'C']
143 Routing Table: {
144     A: 0         via: None
145     B: 5         via: B
146     C: 2         via: C
147 }
148
149 **** Iteration 10 *****
150
151 A => Router Object
152 Name: A
153 Neighbours: ['B', 'C']
154 Routing Table: {
155     A: 0         via: None
156     B: 5         via: B
157     C: 2         via: C
158 }
159
160 **** Iteration 11 *****
161
162 A => Router Object
163 Name: A
164 Neighbours: ['B', 'C']
165 Routing Table: {
166     A: 0         via: None
167     B: 5         via: B
168     C: 2         via: C
169 }
170
171 **** Iteration 12 *****
172
173 A => Router Object
174 Name: A
175 Neighbours: ['B', 'C']
176 Routing Table: {
177     A: 0         via: None
178     B: 5         via: B
179     C: 2         via: C
180 }
181
182 **** Iteration 13 *****
183
184 A => Router Object
185 Name: A
186 Neighbours: ['B', 'C']
187 Routing Table: {
188     A: 0         via: None
189     B: 5         via: B
190     C: 2         via: C
191 }
192
193 **** Iteration 14 *****
194
195 A => Router Object
196 Name: A
197 Neighbours: ['B', 'C']
198 Routing Table: {
199     A: 0         via: None
200     B: 5         via: B
201     C: 2         via: C
202 }
203
204 **** Iteration 15 *****
205
206 A => Router Object
207 Name: A
208 Neighbours: ['B', 'C']
209 Routing Table: {
210     A: 0         via: None
211     B: 5         via: B
212     C: 2         via: C
213 }
214
215 **** Iteration 16 *****
216
217 A => Router Object
218 Name: A
219 Neighbours: ['B', 'C']
220 Routing Table: {
221     A: 0         via: None
222     B: 5         via: B
223     C: 2         via: C
224 }
225
226 **** Iteration 17 *****
227
228 A => Router Object
229 Name: A
230 Neighbours: ['B', 'C']
231 Routing Table: {
232     A: 0         via: None
233     B: 5         via: B
234     C: 2         via: C
235 }
236
237 **** Iteration 18 *****
238
239 A => Router Object
240 Name: A
241 Neighbours: ['B', 'C']
242 Routing Table: {
243     A: 0         via: None
244     B: 5         via: B
245     C: 2         via: C
246 }
247
248 **** Iteration 19 *****
249
250 A => Router Object
251 Name: A
252 Neighbours: ['B', 'C']
253 Routing Table: {
254     A: 0         via: None
255     B: 5         via: B
256     C: 2         via: C
257 }
258
259 **** Iteration 20 *****
260
261 A => Router Object
262 Name: A
263 Neighbours: ['B', 'C']
264 Routing Table: {
265     A: 0         via: None
266     B: 5         via: B
267     C: 2         via: C
268 }
269
270 **** Iteration 21 *****
271
272 A => Router Object
273 Name: A
274 Neighbours: ['B', 'C']
275 Routing Table: {
276     A: 0         via: None
277     B: 5         via: B
278     C: 2         via: C
279 }
280
281 **** Iteration 22 *****
282
283 A => Router Object
284 Name: A
285 Neighbours: ['B', 'C']
286 Routing Table: {
287     A: 0         via: None
288     B: 5         via: B
289     C: 2         via: C
290 }
291
292 **** Iteration 23 *****
293
294 A => Router Object
295 Name: A
296 Neighbours: ['B', 'C']
297 Routing Table: {
298     A: 0         via: None
299     B: 5         via: B
300     C: 2         via: C
301 }
302
303 **** Iteration 24 *****
304
305 A => Router Object
306 Name: A
307 Neighbours: ['B', 'C']
308 Routing Table: {
309     A: 0         via: None
310     B: 5         via: B
311     C: 2         via: C
312 }
313
314 **** Iteration 25 *****
315
316 A => Router Object
317 Name: A
318 Neighbours: ['B', 'C']
319 Routing Table: {
320     A: 0         via: None
321     B: 5         via: B
322     C: 2         via: C
323 }
324
325 **** Iteration 26 *****
326
327 A => Router Object
328 Name: A
329 Neighbours: ['B', 'C']
330 Routing Table: {
331     A: 0         via: None
332     B: 5         via: B
333     C: 2         via: C
334 }
335
336 **** Iteration 27 *****
337
338 A => Router Object
339 Name: A
340 Neighbours: ['B', 'C']
341 Routing Table: {
342     A: 0         via: None
343     B: 5         via: B
344     C: 2         via: C
345 }
346
347 **** Iteration 28 *****
348
349 A => Router Object
350 Name: A
351 Neighbours: ['B', 'C']
352 Routing Table: {
353     A: 0         via: None
354     B: 5         via: B
355     C: 2         via: C
356 }
357
358 **** Iteration 29 *****
359
360 A => Router Object
361 Name: A
362 Neighbours: ['B', 'C']
363 Routing Table: {
364     A: 0         via: None
365     B: 5         via: B
366     C: 2         via: C
367 }
368
369 **** Iteration 30 *****
370
371 A => Router Object
372 Name: A
373 Neighbours: ['B', 'C']
374 Routing Table: {
375     A: 0         via: None
376     B: 5         via: B
377     C: 2         via: C
378 }
379
380 **** Iteration 31 *****
381
382 A => Router Object
383 Name: A
384 Neighbours: ['B', 'C']
385 Routing Table: {
386     A: 0         via: None
387     B: 5         via: B
388     C: 2         via: C
389 }
390
391 **** Iteration 32 *****
392
393 A => Router Object
394 Name: A
395 Neighbours: ['B', 'C']
396 Routing Table: {
397     A: 0         via: None
398     B: 5         via: B
399     C: 2         via: C
400 }
401
402 **** Iteration 33 *****
403
404 A => Router Object
405 Name: A
406 Neighbours: ['B', 'C']
407 Routing Table: {
408     A: 0         via: None
409     B: 5         via: B
410     C: 2         via: C
411 }
412
413 **** Iteration 34 *****
414
415 A => Router Object
416 Name: A
417 Neighbours: ['B', 'C']
418 Routing Table: {
419     A: 0         via: None
420     B: 5         via: B
421     C: 2         via: C
422 }
423
424 **** Iteration 35 *****
425
426 A => Router Object
427 Name: A
428 Neighbours: ['B', 'C']
429 Routing Table: {
430     A: 0         via: None
431     B: 5         via: B
432     C: 2         via: C
433 }
434
435 **** Iteration 36 *****
436
437 A => Router Object
438 Name: A
439 Neighbours: ['B', 'C']
440 Routing Table: {
441     A: 0         via: None
442     B: 5         via: B
443     C: 2         via: C
444 }
445
446 **** Iteration 37 *****
447
448 A => Router Object
449 Name: A
450 Neighbours: ['B', 'C']
451 Routing Table: {
452     A: 0         via: None
453     B: 5         via: B
454     C: 2         via: C
455 }
456
457 **** Iteration 38 *****
458
459 A => Router Object
460 Name: A
461 Neighbours: ['B', 'C']
462 Routing Table: {
463     A: 0         via: None
464     B: 5         via: B
465     C: 2         via: C
466 }
467
468 **** Iteration 39 *****
469
470 A => Router Object
471 Name: A
472 Neighbours: ['B', 'C']
473 Routing Table: {
474     A: 0         via: None
475     B: 5         via: B
476     C: 2         via: C
477 }
478
479 **** Iteration 40 *****
480
481 A => Router Object
482 Name: A
483 Neighbours: ['B', 'C']
484 Routing Table: {
485     A: 0         via: None
486     B: 5         via: B
487     C: 2         via: C
488 }
489
490 **** Iteration 41 *****
491
492 A => Router Object
493 Name: A
494 Neighbours: ['B', 'C']
495 Routing Table: {
496     A: 0         via: None
497     B: 5         via: B
498     C: 2         via: C
499 }
500
501 **** Iteration 42 *****
502
503 A => Router Object
504 Name: A
505 Neighbours: ['B', 'C']
506 Routing Table: {
507     A: 0         via: None
508     B: 5         via: B
509     C: 2         via: C
510 }
511
512 **** Iteration 43 *****
513
514 A => Router Object
515 Name: A
516 Neighbours: ['B', 'C']
517 Routing Table: {
518     A: 0         via: None
519     B: 5         via: B
520     C: 2         via: C
521 }
522
523 **** Iteration 44 *****
524
525 A => Router Object
526 Name: A
527 Neighbours: ['B', 'C']
528 Routing Table: {
529     A: 0         via: None
530     B: 5         via: B
531     C: 2         via: C
532 }
533
534 **** Iteration 45 *****
535
536 A => Router Object
537 Name: A
538 Neighbours: ['B', 'C']
539 Routing Table: {
540     A: 0         via: None
541     B: 5         via: B
542     C: 2         via: C
543 }
544
545 **** Iteration 46 *****
546
547 A => Router Object
548 Name: A
549 Neighbours: ['B', 'C']
550 Routing Table: {
551     A: 0         via: None
552     B: 5         via: B
553     C: 2         via: C
554 }
555
556 **** Iteration 47 *****
557
558 A => Router Object
559 Name: A
560 Neighbours: ['B', 'C']
561 Routing Table: {
562     A: 0         via: None
563     B: 5         via: B
564     C: 2         via: C
565 }
566
567 **** Iteration 48 *****
568
569 A => Router Object
570 Name: A
571 Neighbours: ['B', 'C']
572 Routing Table: {
573     A: 0         via: None
574     B: 5         via: B
575     C: 2         via: C
576 }
577
578 **** Iteration 49 *****
579
580 A => Router Object
581 Name: A
582 Neighbours: ['B', 'C']
583 Routing Table: {
584     A: 0         via: None
585     B: 5         via: B
586     C: 2         via: C
587 }
588
589 **** Iteration 50 *****
590
591 A => Router Object
592 Name: A
593 Neighbours: ['B', 'C']
594 Routing Table: {
595     A: 0         via: None
596     B: 5         via: B
597     C: 2         via: C
598 }
599
600 **** Iteration 51 *****
601
602 A => Router Object
603 Name: A
604 Neighbours: ['B', 'C']
605 Routing Table: {
606     A: 0         via: None
607     B: 5         via: B
608     C: 2         via: C
609 }
610
611 **** Iteration 52 *****
612
613 A => Router Object
614 Name: A
615 Neighbours: ['B', 'C']
616 Routing Table: {
617     A: 0         via: None
618     B: 5         via: B
619     C: 2         via: C
620 }
621
622 **** Iteration 53 *****
623
624 A => Router Object
625 Name: A
626 Neighbours: ['B', 'C']
627 Routing Table: {
628     A: 0         via: None
629     B: 5         via: B
630     C: 2         via: C
631 }
632
633 **** Iteration 54 *****
634
635 A => Router Object
636 Name: A
637 Neighbours: ['B', 'C']
638 Routing Table: {
639     A: 0         via: None
640     B: 5         via: B
641     C: 2         via: C
642 }
643
644 **** Iteration 55 *****
645
646 A => Router Object
647 Name: A
648 Neighbours: ['B', 'C']
649 Routing Table: {
650     A: 0         via: None
651     B: 5         via: B
652     C: 2         via: C
653 }
654
655 **** Iteration 56 *****
656
657 A => Router Object
658 Name: A
659 Neighbours: ['B', 'C']
660 Routing Table: {
661     A: 0         via: None
662     B: 5         via: B
663     C: 2         via: C
664 }
665
666 **** Iteration 57 *****
667
668 A => Router Object
669 Name: A
670 Neighbours: ['B', 'C']
671 Routing Table: {
672     A: 0         via: None
673     B: 5         via: B
674     C: 2         via: C
675 }
676
677 **** Iteration 58 *****
678
679 A => Router Object
680 Name: A
681 Neighbours: ['B', 'C']
682 Routing Table: {
683     A: 0         via: None
684     B: 5         via: B
685     C: 2         via: C
686 }
687
688 **** Iteration 59 *****
689
690 A => Router Object
691 Name: A
692 Neighbours: ['B', 'C']
693 Routing Table: {
694     A: 0         via: None
695     B: 5         via: B
696     C: 2         via: C
697 }
698
699 **** Iteration 60 *****
700
701 A => Router Object
702 Name: A
703 Neighbours: ['B', 'C']
704 Routing Table: {
705     A: 0         via: None
706     B: 5         via: B
707     C: 2         via: C
708 }
709
710 **** Iteration 61 *****
711
712 A => Router Object
713 Name: A
714 Neighbours: ['B', 'C']
715 Routing Table: {
716     A: 0         via: None
717     B: 5         via: B
718     C: 2         via: C
719 }
720
721 **** Iteration 62 *****
722
723 A => Router Object
724 Name: A
725 Neighbours: ['B', 'C']
726 Routing Table: {
727     A: 0         via: None
728     B: 5         via: B
729     C: 2         via: C
730 }
731
732 **** Iteration 63 *****
733
734 A => Router Object
735 Name: A
736 Neighbours: ['B', 'C']
737 Routing Table: {
738     A: 0         via: None
739     B: 5         via: B
740     C: 2         via: C
741 }
742
743 **** Iteration 64 *****
744
745 A => Router Object
746 Name: A
747 Neighbours: ['B', 'C']
748 Routing Table: {
749     A: 0         via: None
750     B: 5         via: B
751     C: 2         via: C
752 }
753
754 **** Iteration 65 *****
755
756 A => Router Object
757 Name: A
758 Neighbours: ['B', 'C']
759 Routing Table: {
760     A: 0         via: None
761     B: 5         via: B
762     C: 2         via: C
763 }
764
765 **** Iteration 66 *****
766
767 A => Router Object
768 Name: A
769 Neighbours: ['B', 'C']
770 Routing Table: {
771     A: 0         via: None
772     B: 5         via: B
773     C: 2         via: C
774 }
775
776 **** Iteration 67 *****
777
778 A => Router Object
779 Name: A
780 Neighbours: ['B', 'C']
781 Routing Table: {
782     A: 0         via: None
783     B: 5         via: B
784     C: 2         via: C
785 }
786
787 **** Iteration 68 *****
788
789 A => Router Object
790 Name: A
791 Neighbours: ['B', 'C']
792 Routing Table: {
793     A: 0         via: None
794     B: 5         via: B
795     C: 2         via: C
796 }
797
798 **** Iteration 69 *****
799
800 A => Router Object
801 Name: A
802 Neighbours: ['B', 'C']
803 Routing Table: {
804     A: 0         via: None
805     B: 5         via: B
806     C: 2         via: C
807 }
808
809 **** Iteration 70 *****
810
811 A => Router Object
812 Name: A
813 Neighbours: ['B', 'C']
814 Routing Table: {
815     A: 0         via: None
816     B: 5         via: B
817     C: 2         via: C
818 }
819
820 **** Iteration 71 *****
821
822 A => Router Object
823 Name: A
824 Neighbours: ['B', 'C']
825 Routing Table: {
826     A: 0         via: None
827     B: 5         via: B
828     C: 2         via: C
829 }
830
831 **** Iteration 72 *****
832
833 A => Router Object
834 Name: A
835 Neighbours: ['B', 'C']
836 Routing Table: {
837     A: 0         via: None
838     B: 5         via: B
839     C: 2         via: C
840 }
841
842 **** Iteration 73 *****
843
844 A => Router Object
845 Name: A
846 Neighbours: ['B', 'C']
847 Routing Table: {
848     A: 0         via: None
849     B: 5         via: B
850     C: 2         via: C
851 }
852
853 **** Iteration 74 *****
854
855 A => Router Object
856 Name: A
857 Neighbours: ['B', 'C']
858 Routing Table: {
859     A: 0         via: None
860     B: 5         via: B
861     C: 2         via: C
862 }
863
864 **** Iteration 75 *****
865
866 A => Router Object
867 Name: A
868 Neighbours: ['B', 'C']
869 Routing Table: {
870     A: 0         via: None
871     B: 5         via: B
872     C: 2         via: C
873 }
874
875 **** Iteration 76 *****
876
877 A => Router Object
878 Name: A
879 Neighbours: ['B', 'C']
880 Routing Table: {
881     A: 0         via: None
882     B: 5         via: B
883     C: 2         via: C
884 }
885
886 **** Iteration 77 *****
887
888 A => Router Object
889 Name: A
890 Neighbours: ['B', 'C']
891 Routing Table: {
892     A: 0         via: None
893     B: 5         via: B
894     C: 2         via: C
895 }
896
897 **** Iteration 78 *****
898
899 A => Router Object
900 Name: A
901 Neighbours: ['B', 'C']
902 Routing Table: {
903     A: 0         via: None
904     B: 5         via: B
905     C: 2         via: C
906 }
907
908 **** Iteration 79 *****
909
910 A => Router Object
911 Name: A
912 Neighbours: ['B', 'C']
913 Routing Table: {
914     A: 0         via: None
915     B: 5         via: B
916     C: 2         via: C
917 }
918
919 **** Iteration 80 *****
920
921 A => Router Object
922 Name: A
923 Neighbours: ['B', 'C']
924 Routing Table: {
925     A: 0         via: None
926     B: 5         via: B
927     C: 2         via: C
928 }
929
930 **** Iteration 81 *****
931
932 A => Router Object
933 Name: A
934 Neighbours: ['B', 'C']
935 Routing Table: {
936     A: 0         via: None
937     B: 5         via: B
938     C: 2         via: C
939 }
940
941 **** Iteration 82 *****
942
943 A => Router Object
944 Name: A
945 Neighbours: ['B', 'C']
946 Routing Table: {
947     A: 0         via: None
948     B: 5         via: B
949     C: 2         via: C
950 }
951
952 **** Iteration 83 *****
953
954 A => Router Object
955 Name: A
956 Neighbours: ['B', 'C']
957 Routing Table: {
958     A: 0         via: None
959     B: 5         via: B
960     C: 2         via: C
961 }
962
963 **** Iteration 84 *****
964
965 A => Router Object
966 Name: A
967 Neighbours: ['B', 'C']
968 Routing Table: {
969     A: 0         via: None
970     B: 5         via: B
971     C: 2         via: C
972 }
973
974 **** Iteration 85 *****
975
976 A => Router Object
977 Name: A
978 Neighbours: ['B', 'C']
979 Routing Table: {
980     A: 0         via: None
981     B: 5         via: B
982     C: 2         via: C
983 }
984
985 **** Iteration 86 *****
986
987 A => Router Object
988 Name: A
989 Neighbours: ['B', 'C']
990 Routing Table: {
991     A: 0         via: None
992     B: 5         via: B
993     C: 2         via: C
994 }
995
996 **** Iteration 87 *****
997
998 A => Router Object
999 Name: A
1000 Neighbours: ['B', 'C']
1001 Routing Table: {
1002     A: 0         via: None
1003     B: 5         via: B
1004     C: 2         via: C
1005 }
1006
1007 **** Iteration 88 *****
1008
1009 A => Router Object
1010 Name: A
1011 Neighbours: ['B', 'C']
1012 Routing Table: {
1013     A: 0         via: None
1014     B: 5         via: B
1015     C: 2         via: C
1016 }
1017
1018 **** Iteration 89 *****
1019
1020 A => Router Object
1021 Name: A
1022 Neighbours: ['B', 'C']
1023 Routing Table: {
1024     A: 0         via: None
1025     B: 5         via: B
1026     C: 2         via: C
1027 }
1028
1029 **** Iteration 90 *****
1030
1031 A => Router Object
1032 Name: A
1033 Neighbours: ['B', 'C']
1034 Routing Table: {
1035     A: 0         via: None
1036     B: 5         via: B
1037     C: 2         via: C
1038 }
1039
1040 **** Iteration 91 *****
1041
1042 A => Router Object
1043 Name: A
1044 Neighbours: ['B', 'C']
1045 Routing Table: {
1046     A: 0         via: None
1047     B: 5         via: B
1048     C: 2         via: C
1049 }
1050
1051 **** Iteration 92 *****
1052
1053 A => Router Object
1054 Name: A
1055 Neighbours: ['B', 'C']
1056 Routing Table: {
1057     A: 0         via: None
1058     B: 5         via: B
1059     C: 2         via: C
1060 }
1061
1062 **** Iteration 93 *****
1063
1064 A => Router Object
1065 Name: A
1066 Neighbours: ['B', 'C']
1067 Routing Table: {
1068     A: 0         via: None
1069     B: 5         via: B
1070     C: 2         via: C
1071 }
1072
1073 **** Iteration 94 *****
1074
1075 A => Router Object
1076 Name: A
1077 Neighbours: ['B', 'C']
1078 Routing Table: {
1079     A: 0         via: None
1080     B: 5         via: B
1081     C: 2         via: C
1082 }
1083
1084 **** Iteration 95 *****
1085
1086 A => Router Object
1087 Name: A
1088 Neighbours: ['B', 'C']
1089 Routing Table: {
1090     A: 0         via: None
1091     B: 5         via: B
1092     C: 2         via: C
1093 }
1094
1095 **** Iteration 96 *****
1096
1097 A => Router Object
1098 Name: A
1099 Neighbours: ['B', 'C']
1100 Routing Table: {
1101     A: 0         via: None
1102     B: 5         via: B
1103     C: 2         via: C
1104 }
1105
1106 **** Iteration 97 *****
1107
1108 A => Router Object
1109 Name: A
1110 Neighbours: ['B', 'C']
1111 Routing Table: {
1112     A: 0         via: None
1113     B: 5         via: B
1114     C: 2         via: C
1115 }
1116
1117 **** Iteration 98 *****
1118
1119 A => Router Object
1120 Name: A
1121 Neighbours: ['B', 'C']
1122 Routing Table: {
1123     A: 0         via: None
1124     B: 5         via: B
1125     C: 2         via: C
1126 }
1127
1128 **** Iteration 99 *****
1129
1130 A => Router Object
1131 Name: A
1132 Neighbours: ['B', 'C']
1133 Routing Table: {
1134     A: 0         via: None
1135     B: 5         via: B
1136     C: 2         via: C
1137 }
1138
1139 **** Iteration 100 *****
1140
1141 A => Router Object
1142 Name: A
1143 Neighbours: ['B', 'C']
1144 Routing Table: {
1145     A: 0         via: None
1146     B: 5         via: B
1147     C: 2         via: C
1148 }
1149
1150 **** Iteration 101 *****
1151
1152 A => Router Object
1153 Name: A
1154 Neighbours: ['B', 'C']
1155 Routing Table: {
1156     A: 0         via: None
1157     B: 5         via: B
1158     C: 2         via: C
1159 }
1160
1161 **** Iteration 102 *****
1162
1163 A => Router Object
1164 Name: A
1165 Neighbours: ['B', 'C']
1166 Routing Table: {
1167     A: 0         via: None
1168     B: 5         via: B
1169     C: 2         via: C
1170 }
1171
1172 **** Iteration 103 *****
1173
1174 A => Router Object
1175 Name: A
1176 Neighbours: ['B', 'C']
1177 Routing Table: {
1178     A: 0         via: None
1179     B: 5         via: B
1180     C: 2         via: C
1181 }
1182
1183 **** Iteration 104 *****
1184
1185 A => Router Object
1186 Name: A
1187 Neighbours: ['B', 'C']
1188 Routing Table: {
1189     A: 0         via: None
1190     B: 5         via: B
1191     C: 2         via: C
1192 }
1193
1194 **** Iteration 105 *****
1195
1196 A => Router Object
1197 Name: A
1198 Neighbours: ['B', 'C']
1199 Routing Table: {
1200     A: 0         via: None
1201     B: 5         via: B
1202     C: 2         via: C
1203 }
1204
1205 **** Iteration 106 *****
1206
1207 A => Router Object
1208 Name: A
1209 Neighbours: ['B', 'C']
1210 Routing Table: {
1211     A: 0         via: None
1212     B: 5         via: B
1213     C: 2         via: C
1214 }
1215
1216 **** Iteration 107 *****
1217
1218 A => Router Object
1219 Name: A
1220 Neighbours: ['B', 'C']
1221 Routing Table: {
1222     A: 0         via: None
1223     B: 5         via: B
1224     C: 2         via: C
1225 }
1226
1227 **** Iteration 108 *****
1228
1229 A => Router Object
1230 Name: A
1231 Neighbours: ['B', 'C']
1232 Routing Table: {
1233     A: 0        
```

path. This network serves as a very good example to show how the shortest path isn't necessarily the least-cost path.

- The graphical (pictorial) representation of the network in `input2.txt` can be found in `figure.txt`

```

DVR ASSIGNMENT
  output.txt
  -----
  ***** Iteration 0 *****
  1 | 
  2 A => Router Object
  3 Name: A
  4 Neighbours: ['B', 'C', 'D']
  5 Routing Table: {
    6   A: 0      via: None
    7   B: 8      via: B
    8   C: 2      via: C
    9   D: 1      via: D
   10   E: inf    via: [no path]
  }
  11
  12
  13 }

  14
  15 B => Router Object
  16 Name: B
  17 Neighbours: ['A', 'D', 'E']
  18 Routing Table: {
    19   A: 8      via: A
    20   B: 0      via: None
    21   C: inf    via: [no path]
    22   D: 5      via: D
    23   E: 3      via: E
  }
  24
  25
  26 C => Router Object
  27 Name: C
  28 Neighbours: ['A']
  29 Routing Table: {
    30   A: 2      via: A
    31   B: inf    via: [no path]
    32   C: 0      via: None
    33   D: inf    via: [no path]
    34   E: inf    via: [no path]
  }
  35
  36
  37 D => Router Object
  38 Name: D
  39 Neighbours: ['A', 'B']
  40 Routing Table: {
    41   A: 1      via: A
  }

  -----
  1 5
  2 A B C D E
  3 A B B
  4 A C 2
  5 A D 1
  6 B D 5
  7 B E 3
  8 EOF
  
```

```

  1   A
  2   / \ (2)
  3   (8) / | \
  4   /   |   C
  5   B   |
  6   (3) / \ | (1)
  7   /   \ |
  8   E (5)\ |
  9   |   |
  
```

DVR ASSIGNMENT

input1.txt

input2.txt

output.txt

dvr.py

figure.txt

input1.txt

input2.txt

output.txt

output.txt — DVR Assignment

input2.txt — DVR Assignment

figure.txt — DVR Assignment

```
1 5
2 A B C D E
3 A B B
4 A C 2
5 A D 1
6 B D 5
7 B E 3
8 EOF

***** Iteration 1 *****
A => Router Object
Name: A
Neighbours: ['B', 'C', 'D']
Routing Table: {
    *A: 0      via: None
    *B: 6      via: D
    C: 2      via: C
    D: 1      via: D
    *E: 11     via: B
}

B => Router Object
Name: B
Neighbours: ['A', 'D', 'E']
Routing Table: {
    *A: 6      via: D
    B: 0      via: None
    *C: 10     via: A
    D: 5      via: D
    E: 3      via: E
}

C => Router Object
Name: C
Neighbours: ['A']
Routing Table: {
    A: 2      via: A
    *B: 10     via: A
    C: 0      via: None
    *D: 3      via: A
    E: inf     via: [no path]
}

D => Router Object
Name: D
```

DVR ASSIGNMENT

```
173 // output.txt
174
175
176 ***** Iteration 3 *****
177
178 A => Router Object
179 Name: A
180 Neighbours: ['B', 'C', 'D']
181 Routing Table: {
182     A: 0    via: None
183     B: 6    via: D
184     C: 2    via: C
185     D: 1    via: D
186     E: 9    via: D
187 }
188
189 B => Router Object
190 Name: B
191 Neighbours: ['A', 'D', 'E']
192 Routing Table: {
193     A: 5    via: D
194     B: 0    via: None
195     C: 8    via: D
196     D: 5    via: D
197     E: 3    via: E
198 }
199
200 C => Router Object
201 Name: C
202 Neighbours: ['A']
203 Routing Table: {
204     A: 2    via: A
205     B: 8    via: A
206     C: 0    via: None
207     D: 3    via: A
208     *E: 11   via: A
209 }
210
211 D => Router Object
212 Name: D
213 Neighbours: ['A', 'B']
```

output.txt — DVR Assignment

```
1 5
2 A B C D E
3 A B B
4 A C 2
5 A D 1
6 B D 5
7 B E 3
8 EOF
```

input2.txt — DVR Assignment

```
1
2 / \ (2)
3 (8) / \
4 | / | C
5 B |
6 (3) / \ ((1)
7 / \
8 E (5)\ |
9 | | D
```

figure.txt — DVR Assignment