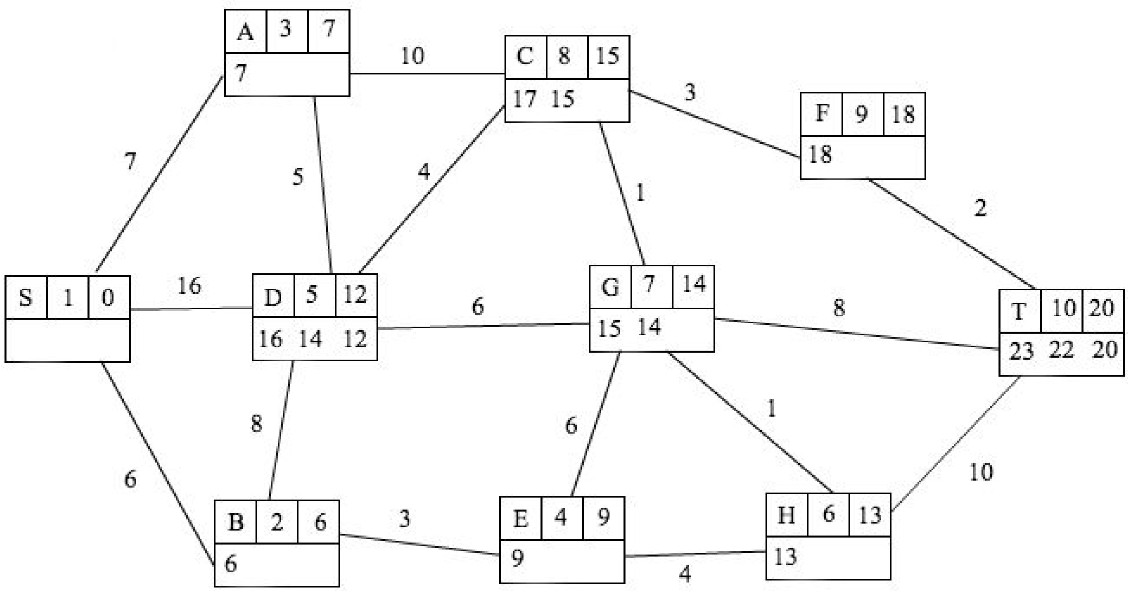
Exercise 3E

1 a Use Dijkstra's algorithm to construct the following graph

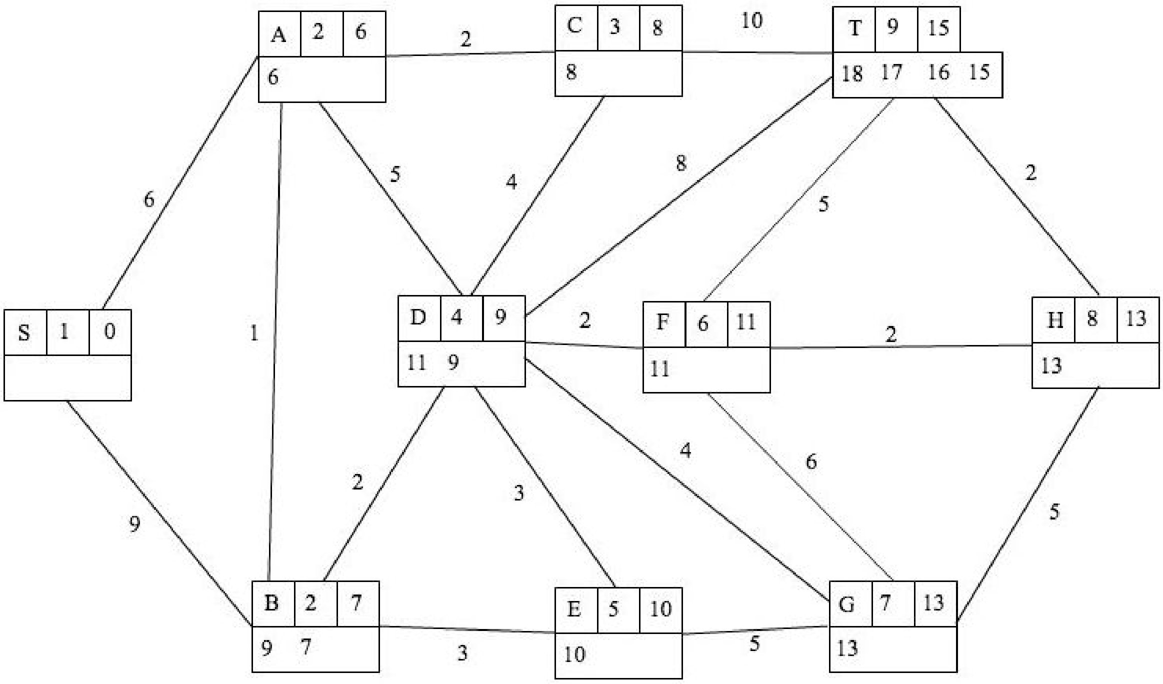


So the shortest route from *S* to *T* has length 20. Now, to find this route, work backwards from *T*:

20 – 2 = 18 *TF*
18 – 3 = 15 *FC*
15 – 1 = 14 *CG*
14 – 1 = 13 *GH*
13 – 4 = 9 *HE*
9 – 3 = 6 *EB*
6 – 6 = 0 *BS*

Thus, the shortest route: *SBEHGCFT*.
Length of shortest route: 20

1 b Use Dijkstra's algorithm to construct the following graph.

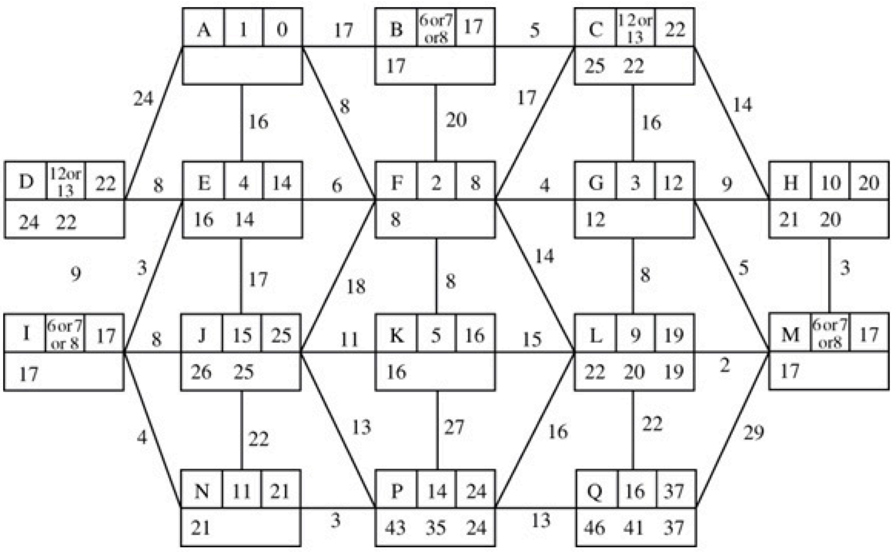


So the shortest route from S to T has length 15. To find the route, work backwards from T:

15 – 2 = 13	*TH*
13 – 2 = 11	*HF*
11 – 2 = 9	*FD*
9 – 2 = 7	*DB*
9 – 9 = 0	*BS*

Thus, the shortest route: *SABDFHT*. Length of shortest route: 15

2



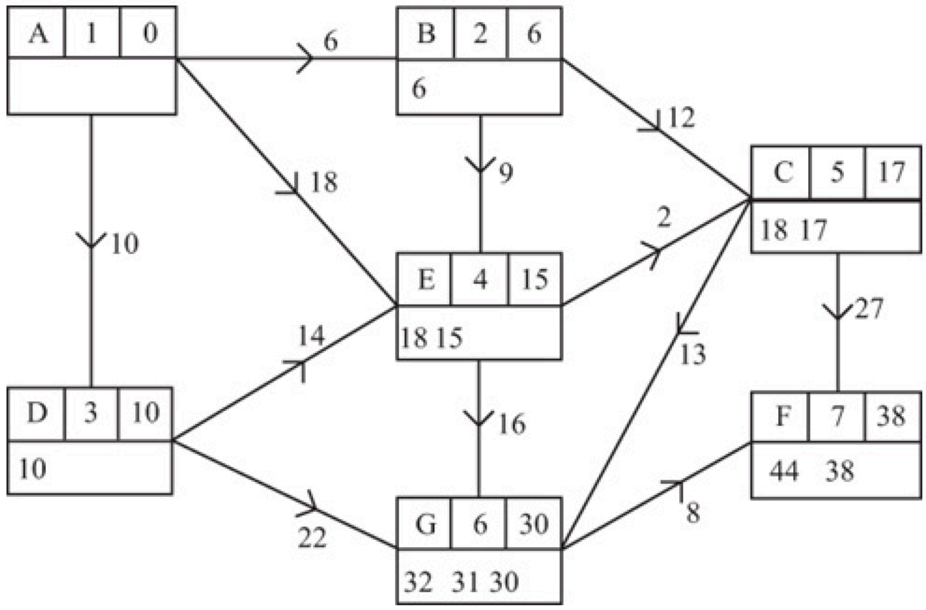
a *A* to *Q* *AFEINPQ* Length 37

b *A* to *L* *AFGML* Length 19

c *M* to *A* *MGFA* Length 17

d *P* to *A* *PNIEFA* Length 24

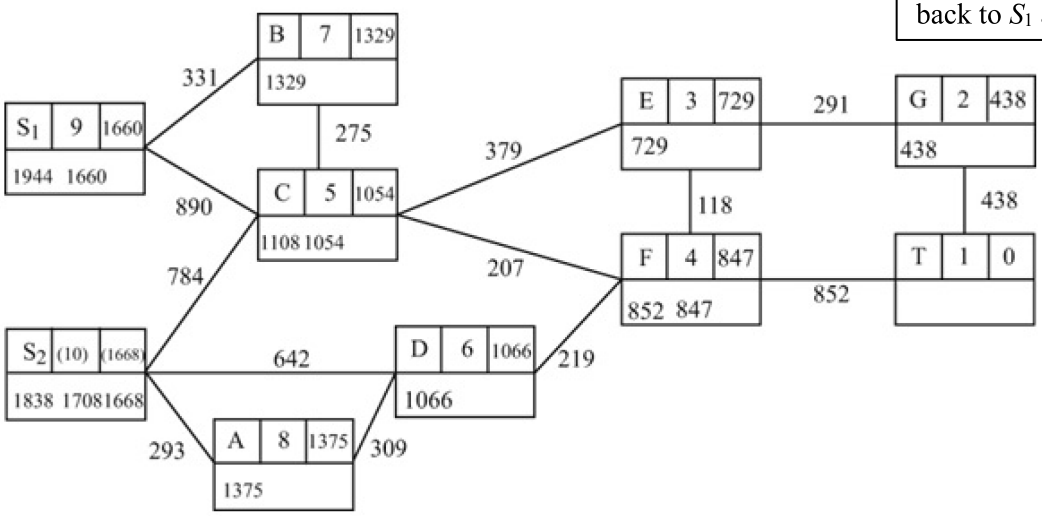
3



Shortest route: *ABECGF*
Length 38

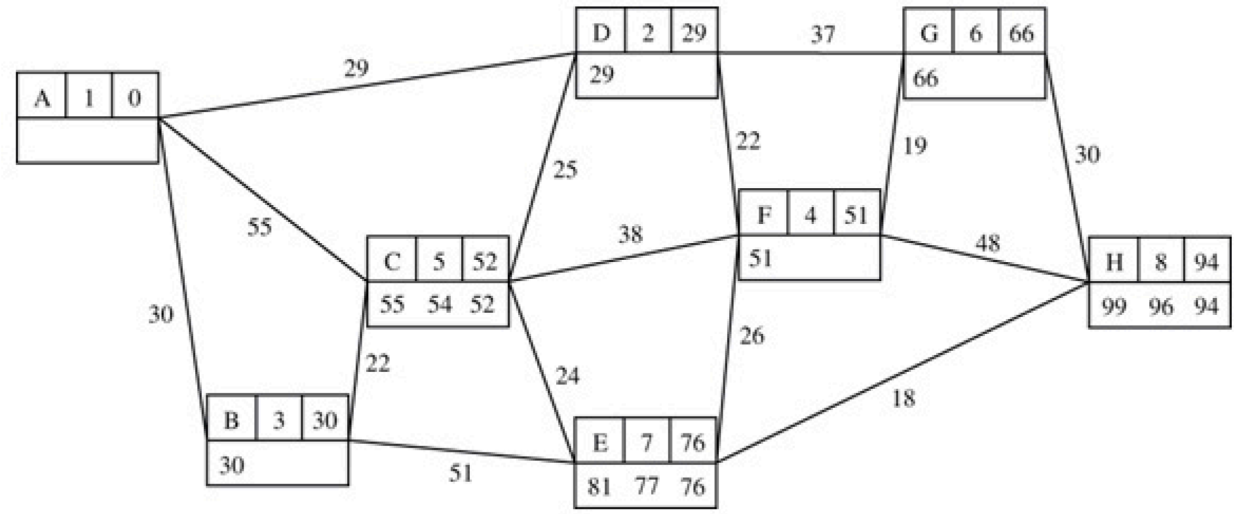
4

Start at T and work back to S_1 and S_2.



Shortest route: $S_1BCFEGT$
Length of shortest route: 1660

5

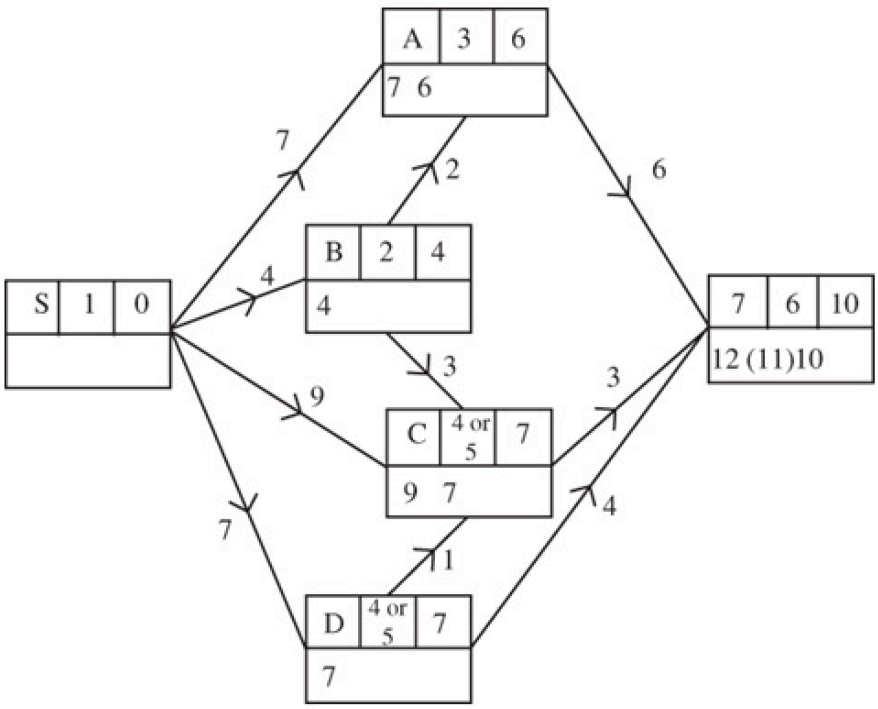


a 94 – 18 = 76 *EH*
76 – 24 = 52 *CE*
52 – 22 = 30 *BC*
30 – 30 = 0 *AB*
Shortest route *A* to *H*: *ABCEH*
Length 94

b Shortest route *A* to *H* via *G*: *ADGH*
Length 96

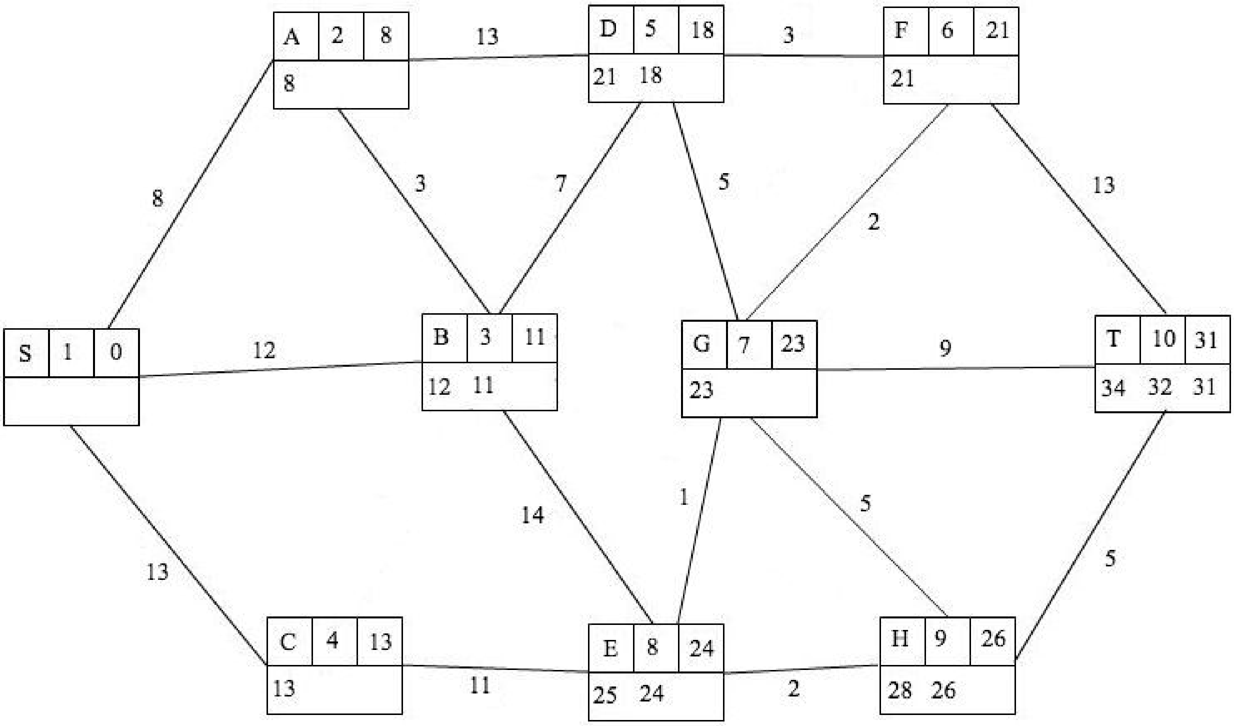
c Shortest route *A* to *H* not using *CE*: *ADFEH*
Length 95

6



Shortest route: *SBCT*
Length of shortest route: 10

7 a Use Dijkstra's algorithm to construct the following graph

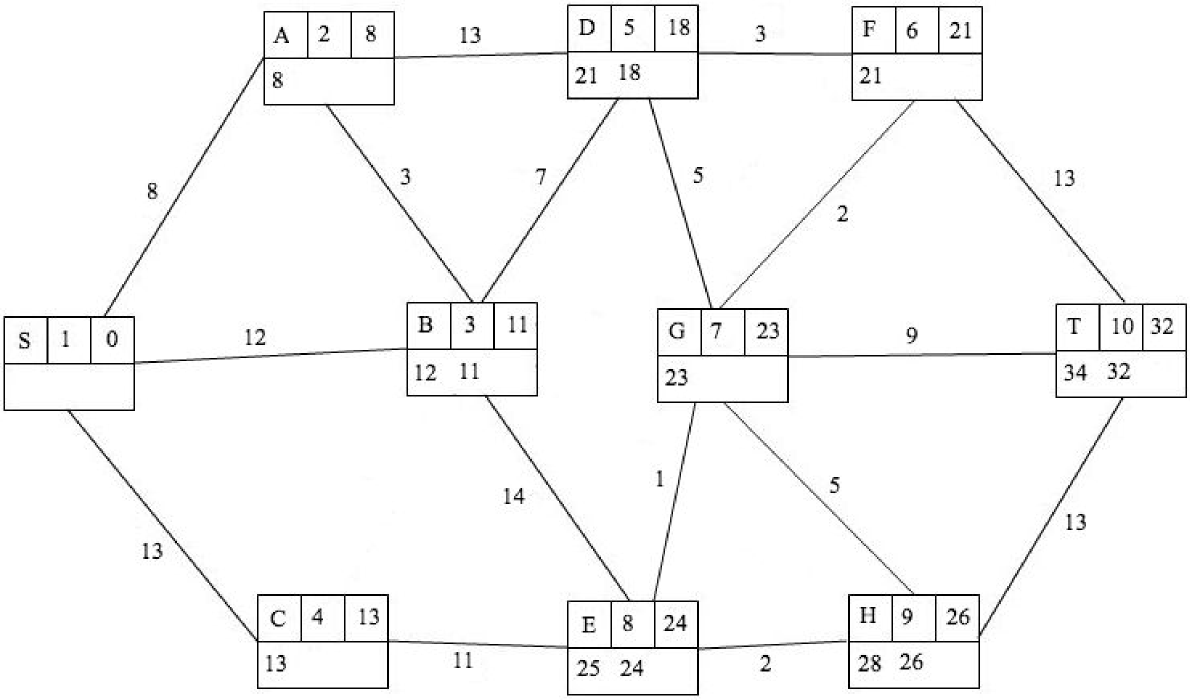


So the quickest route has length 31 minutes. To find the route, work backwards from T:

31 – 5 = 26	*TH*		
26 – 2 = 24	*HE*		
24 – 1 = 23	*EG*	or	24 – 11 = 13 *EC*
23 – 2 = 21	*GF*	or	13 – 13 = 0 *CS*
21 – 3 = 18	*FD*		
18 – 7 = 11	*DB*		
11 – 3 = 8	*BA*		
8 – 8 = 0	*AS*		

So there are 2 routes of the same, shortest time: *SCEHT* and *SABDFGEHT*.
Shortest time = 31 min.

7 b i Use Dijkstra's algorithm to create the following graph

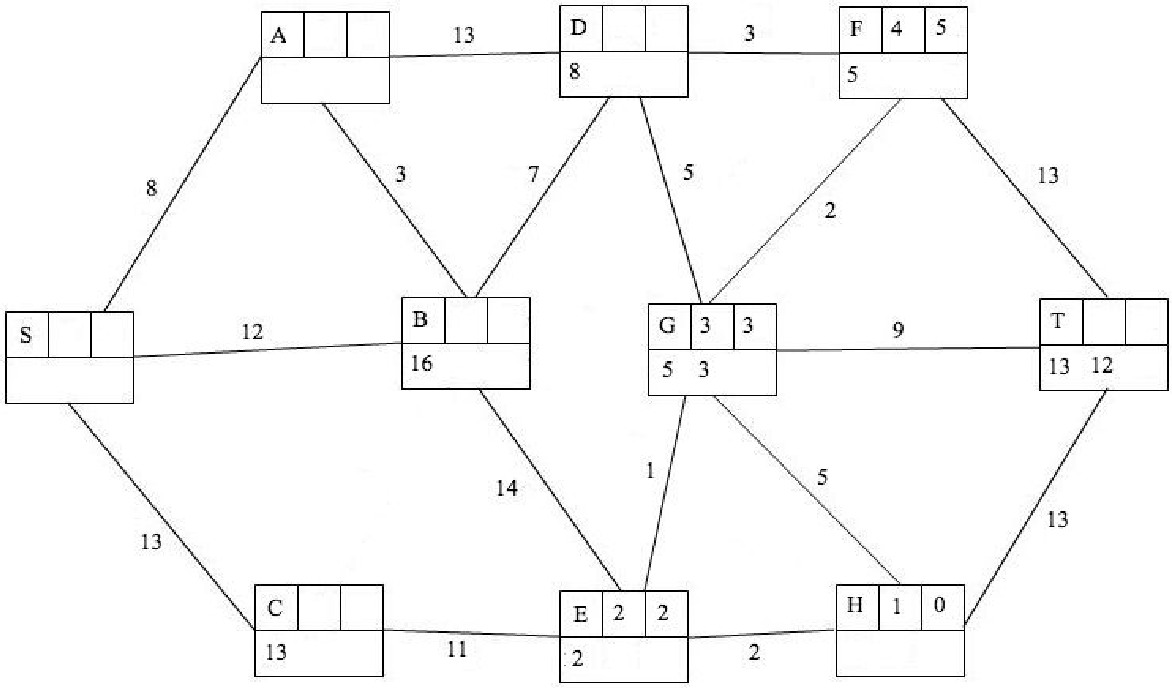


So the length of the journey changes to 32 minutes. To find the route, work backwards from *T*:

32 – 9 = 23	*TG*			
23 – 2 = 21	*GF*	or	23 – 5 = 18	*GD*
21 – 3 = 18	*FD*	We reached point *D* so both routes coincide again		
18 – 7 = 11	*DB*			
11 – 3 = 8	*BA*			
8 – 8 = 0	*AS*			

So the route changes to *SABDFGT* or *SABDGT*, both of length 32 minutes.

7 b ii If the driver finds out about the change at *H,* that is his penultimate stop, we can consider the following graph



Now, even though the graph is unfinished (we have not considered what happens at vertices *A* through to *D*), we have exhausted all vertices directly connected to *T* and so any other route would eventually have to reach *F, G* or *H*. This means that any other route would necessarily be longer than what we can construct at the moment. Hence, the quickest route from *H* to *T* is 12 minutes. It can be found by working backwards from *T:*

12 – 9 = 3 *TG*
3 – 1 = 2 *GE*
2 – 2 = 0 *EH*

So instead of going from *H* to *T* directly, the driver should turn back and go *HEGT* to save 1 minute. This will make the total journey time 38 minutes (since from part **i** we know that the driver took 26 minutes to get to *H* and then from the graph above we have another 12 minutes to get from *H* to *T*.