Floyd's algorithm

Worked solutions

Further Maths topic practice

This topic online

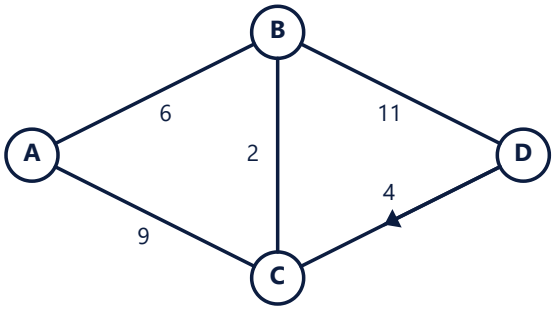
Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

1 The network below shows the roads between four farms, A, B, C and D. The number on each arc is the length of the road in km. The road from D to C is one-way, as shown by the arrow.

Floyd's algorithm is to be used to find the shortest distances between every pair of farms.

a. Complete the initial distance table and the initial route table.

b. Explain why the first iteration of Floyd's algorithm makes no changes to either table.



Initial tables

Distance table

	A	B	C	D
A	-	6	9	∞
B	6	-	2	11
C	9	2	-	∞
D	∞	11	4	-

Route table

	A	B	C	D
A	A	B	C	D
B	A	B	C	D
C	A	B	C	D
D	A	B	C	D

Diagram NOT accurately drawn

(4 marks)

a)

Distance table: the direct distance from the row vertex to the column vertex, ∞ where there is no direct arc

A one-way arc only gives an entry in the direction it can be travelled

A: - 6 9 ∞

B: 6 - 2 11

C: 9 2 - ∞

D: ∞ 11 4 -

Route table: every entry in column X is X to begin with

A: A B C D

B: A B C D

C: A B C D

D: A B C D

b)

Iteration 1 replaces an entry only if going via A is shorter

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Arcs at A: AB = 6 and AC = 9, and there is no arc between A and D

BA + AC = 6 + 9 = 15

15 > BC = 2, so no change

CA + AB = 9 + 6 = 15

15 > CB = 2, so no change

Row D and column D have ∞ in the A position, so no sum via A involving D is finite

No route via A is shorter than the current entry, so neither table changes

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

2 The initial distance table and initial route table for a directed network with vertices A, B, C and D are shown below.

Perform the first iteration of Floyd's algorithm. You should show the distance table and the route table after this iteration.

Initial tables

Distance table

	A	B	C	D
A	-	5	∞	3
B	5	-	4	∞
C	2	∞	-	8
D	3	∞	8	-

Route table

	A	B	C	D
A	A	B	C	D
B	A	B	C	D
C	A	B	C	D
D	A	B	C	D

After the first iteration

Distance table

	A	B	C	D
A	-	5	∞	3
B	5	-	4	**8**
C	2	**7**	-	**5**
D	3	**8**	8	-

Route table

	A	B	C	D
A	A	B	C	D
B	A	B	C	**A**
C	A	**A**	C	**A**
D	A	**A**	C	D

(3 marks)

Iteration 1: row A and column A are copied; for every other cell compare (row to A) + (A to column) with the current entry

BO: BA + AO = 5 + 3 = 8 < ∞, so 8, route A

CB: CA + AB = 2 + 5 = 7 < ∞, so 7, route A

CO: CA + AO = 2 + 3 = 5 < 8, so 5, route A

OB: OA + AB = 3 + 5 = 8 < ∞, so 8, route A

All other entries stay the same

For example BC: BA + AC = 5 + ∞, not smaller than 4, so unchanged

BO = 8, CB = 7, CO = 5, OB = 8, route entries A

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

3 Floyd's algorithm has been applied to a network of roads between five towns, A, B, C, D and E. Some of the roads are one-way. The final distance table, in miles, and the final route table are shown below.

a. Use the final route table to find the shortest route from A to E, explaining how you use the table. State the length of this route.

b. Find the shortest route from E to D and its length.

Final tables

Distance table

	A	B	C	D	E
A	-	8	11	5	15
B	8	-	3	9	7
C	11	3	-	6	4
D	5	9	6	-	10
E	7	7	4	10	-

Route table

	A	B	C	D	E
A	A	(B)	(B)	D	(C)
B	A	B	(C)	C	C
C	B	B	C	(D)	(E)
D	A	C	C	D	C
E	A	C	(C)	(C)	E

(4 marks)

a)

Read row A, column E of the final route table and split the route at that vertex until every piece is direct

Row A, column E is C: the route from A to E goes via C

Row A, column C is B: the route from A to C goes via B

Row A, column B is B: A to B is direct

Row B, column C is C: B to C is direct

Row C, column E is E: C to E is direct

Length from the final distance table: row A, column E = 15

ABCE, 15 miles

b)

Read row E, column D of the final route table and split the route at that vertex until every piece is direct

Row E, column D is C: the route from E to D goes via C

Row E, column C is C: E to C is direct

Row C, column D is D: C to D is direct

Length from the final distance table: row E, column D = 10

ECD, 10 miles

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

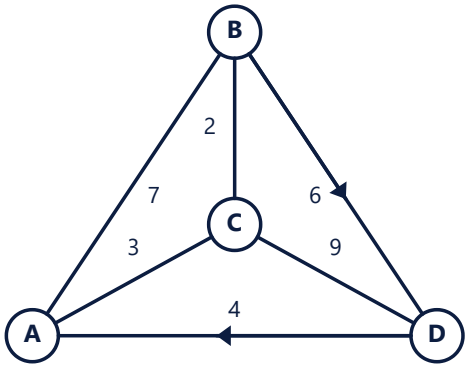
4 The network below shows the paths between four huts, A, B, C and D, on a nature reserve. The number on each arc is the walking time in minutes. The paths from B to D and from D to A are one-way, as shown by the arrows.

The initial distance and route tables are shown below the network.

a. Use Floyd's algorithm to find the shortest walking times between every pair of huts. You should show the distance table and the route table after each iteration.

b. Use your final route table to find the quickest route from D to B, and state its time.

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.



Initial tables

Distance table

	A	B	C	D
A	-	7	3	∞
B	7	-	2	6
C	3	2	-	9
D	4	∞	9	-

Route table

	A	B	C	D
A	A	B	C	D
B	A	B	C	D
C	A	B	C	D
D	A	B	C	D

(9 marks)

First iteration

Distance table

	A	B	C	D
A	-	7	3	∞
B	7	-	2	6
C	3	2	-	9
D	4	**11**	**7**	-

Route table

	A	B	C	D
A	A	B	C	D
B	A	B	C	D
C	A	B	C	D
D	A	**A**	**A**	D

Second iteration

Distance table

	A	B	C	D
A	-	7	3	**13**
B	7	-	2	6
C	3	2	-	**8**
D	4	11	7	-

Route table

	A	B	C	D
A	A	B	C	**B**
B	A	B	C	D
C	A	B	C	**B**
D	A	A	A	D

Third iteration

Distance table

	A	B	C	D
A	-	**5**	3	**11**
B	**5**	-	2	6
C	3	2	-	8
D	4	**9**	7	-

Route table

	A	B	C	D
A	A	**C**	C	**C**
B	**C**	B	C	D
C	A	B	C	B
D	A	**C**	A	D

Fourth iteration

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Distance table

	A	B	C	D
A	-	5	3	11
B	5	-	2	6
C	3	2	-	8
D	4	9	7	-

Route table

	A	B	C	D
A	A	C	C	C
B	C	B	C	D
C	A	B	C	B
D	A	C	A	D

Diagram NOT accurately drawn

a)

Iteration 1: row A and column A are copied; for every other cell compare (row to A) + (A to column) with the current entry

OB: OA + AB = 4 + 7 = 11 < ∞, so 11, route A

OC: OA + AC = 4 + 3 = 7 < 9, so 7, route A

All other entries stay the same

Iteration 2: row B and column B are copied; for every other cell compare (row to B) + (B to column) with the current entry

AO: AB + BO = 7 + 6 = 13 < ∞, so 13, route B

CO: CB + BO = 2 + 6 = 8 < 9, so 8, route B

All other entries stay the same

Iteration 3: row C and column C are copied; for every other cell compare (row to C) + (C to column) with the current entry

AB: AC + CB = 3 + 2 = 5 < 7, so 5, route C

AO: AC + CO = 3 + 8 = 11 < 13, so 11, route C

BA: BC + CA = 2 + 3 = 5 < 7, so 5, route C

OB: OC + CB = 7 + 2 = 9 < 11, so 9, route C

All other entries stay the same

Iteration 4: row O and column O are copied; for every other cell compare (row to O) + (O to column) with the current entry

No sum through O is smaller than the current entry, so both tables are unchanged

After the fourth iteration the tables are final

b)

Read row O, column B of the final route table and split the route at that vertex until every piece is direct

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Row 0, column B is C: the route from 0 to B goes via C

Row 0, column C is A: the route from 0 to C goes via A

Row 0, column A is A: 0 to A is direct

Row A, column C is C: A to C is direct

Row C, column B is B: C to B is direct

Length from the final distance table: row 0, column B = 9

$4 + 3 + 2 = 9$

OACB, 9 minutes

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

5 Floyd's algorithm is being used to find the shortest times, in minutes, between five stations, A, B, C, D and E, on a tram network in which some links run in one direction only. The distance table and route table after the second iteration are shown below.

a. Perform the third, fourth and fifth iterations of Floyd's algorithm. You should show the distance table and the route table after each iteration.

b. Use your final tables to find the quickest journey from C to D and its time.

After the second iteration

Distance table

	A	B	C	D	E
A	-	4	7	11	14
B	4	-	3	7	18
C	10	14	-	21	5
D	9	7	10	-	2
E	14	18	5	2	-

Route table

	A	B	C	D	E
A	A	B	B	B	E
B	A	B	C	D	A
C	A	A	C	B	E
D	A	B	B	D	E
E	A	A	C	D	E

Third iteration

Distance table

	A	B	C	D	E
A	-	4	7	11	**12**
B	4	-	3	7	**8**
C	10	14	-	21	5
D	9	7	10	-	2
E	14	18	5	2	-

Route table

	A	B	C	D	E
A	A	B	B	B	**C**
B	A	B	C	D	**C**
C	A	A	C	B	E
D	A	B	B	D	E
E	A	A	C	D	E

Fourth iteration

Distance table

	A	B	C	D	E
A	-	4	7	11	12
B	4	-	3	7	8
C	10	14	-	21	5
D	9	7	10	-	2
E	**11**	**9**	5	2	-

Route table

	A	B	C	D	E
A	A	B	B	B	C
B	A	B	C	D	C
C	A	A	C	B	E
D	A	B	B	D	E
E	**D**	**D**	C	D	E

Fifth iteration

Distance table

	A	B	C	D	E
A	-	4	7	11	12
B	4	-	3	7	8
C	10	14	-	**7**	5
D	9	7	**7**	-	2
E	11	9	5	2	-

Route table

	A	B	C	D	E
A	A	B	B	B	C
B	A	B	C	D	C
C	A	A	C	**E**	E
D	A	B	**E**	D	E
E	D	D	C	D	E

(7 marks)

a)

Iteration 3: row C and column C are copied; for every other cell compare (row to C) + (C to column) with the current entry

AE: AC + CE = 7 + 5 = 12 < 14, so 12, route C

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

BE: BC + CE = 3 + 5 = 8 < 18, so 8, route C

All other entries stay the same

Iteration 4: row 0 and column 0 are copied; for every other cell compare (row to 0) + (0 to column) with the current entry

EA: E0 + 0A = 2 + 9 = 11 < 14, so 11, route 0

EB: E0 + 0B = 2 + 7 = 9 < 18, so 9, route 0

All other entries stay the same

Iteration 5: row E and column E are copied; for every other cell compare (row to E) + (E to column) with the current entry

C0: CE + E0 = 5 + 2 = 7 < 21, so 7, route E

0C: 0E + EC = 2 + 5 = 7 < 10, so 7, route E

All other entries stay the same

b)

Read row C, column 0 of the final route table and split the route at that vertex until every piece is direct

Row C, column 0 is E: the route from C to 0 goes via E

Row C, column E is E: C to E is direct

Row E, column 0 is 0: E to 0 is direct

Length from the final distance table: row C, column 0 = 7

CE0, 7 minutes

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

6 The initial distance table below gives the lengths, in metres, of the walkways between five buildings, A, B, C, D and E, on a hospital site. Some walkways are one-way.

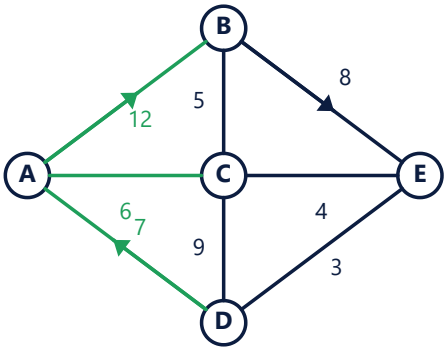
	A	**B**	**C**	**D**	**E**
A	–	12	6	∞	∞
B	∞	–	5	∞	8
C	6	5	–	9	4
D	7	∞	9	–	3
E	∞	∞	4	3	–

The drawing of the network below is incomplete.

a. By adding the arcs that join vertex A to the other vertices, with their weights and any arrows needed, complete the drawing of the network.

b. Perform the first two iterations of Floyd's algorithm, starting from the initial tables given below the network. You should show the distance table and the route table after each iteration.

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.



(6 marks)

Initial tables

Distance table

	A	B	C	D	E
A	-	12	6	∞	∞
B	∞	-	5	∞	8
C	6	5	-	9	4
D	7	∞	9	-	3
E	∞	∞	4	3	-

Route table

	A	B	C	D	E
A	A	B	C	D	E
B	A	B	C	D	E
C	A	B	C	D	E
D	A	B	C	D	E
E	A	B	C	D	E

First iteration

Distance table

	A	B	C	D	E
A	-	12	6	∞	∞
B	∞	-	5	∞	8
C	6	5	-	9	4
D	7	**19**	9	-	3
E	∞	∞	4	3	-

Route table

	A	B	C	D	E
A	A	B	C	D	E
B	A	B	C	D	E
C	A	B	C	D	E
D	A	**A**	C	D	E
E	A	B	C	D	E

Second iteration

Distance table

	A	B	C	D	E
A	-	12	6	∞	**20**
B	∞	-	5	∞	8
C	6	5	-	9	4
D	7	19	9	-	3
E	∞	∞	4	3	-

Route table

	A	B	C	D	E
A	A	B	C	D	**B**
B	A	B	C	D	E
C	A	B	C	D	E
D	A	A	C	D	E
E	A	B	C	D	E

Diagram NOT accurately drawn

a)

Compare row A with column A: equal entries give a two-way arc, a single finite entry gives a one-way arc

AB = 12, BA = ∞: arc from A to B, weight 12, one-way

AC = 6, CA = 6: two-way arc AC, weight 6

AD = ∞, DA = 7: arc from D to A, weight 7, one-way

AE = ∞, EA = ∞: no arc

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

b)

Iteration 1: row A and column A are copied; for every other cell compare (row to A) + (A to column) with the current entry

OB: OA + AB = 7 + 12 = 19 < ∞, so 19, route A

All other entries stay the same

Iteration 2: row B and column B are copied; for every other cell compare (row to B) + (B to column) with the current entry

AE: AB + BE = 12 + 8 = 20 < ∞, so 20, route B

All other entries stay the same

Iteration 1: OB = 19 (route A); iteration 2: AE = 20 (route B)

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

7 a. Explain the difference between the information given by Dijkstra's algorithm and the information given by Floyd's algorithm.

b. Explain why, in the kth iteration of Floyd's algorithm, the entries in the kth row and the kth column of the distance table never change.

c. Floyd's algorithm has cubic order. A computer takes 2.4 seconds to apply Floyd's algorithm to a network with 40 vertices. Estimate the time it would take to apply the algorithm to a network with 100 vertices.

(5 marks)

a)

Dijkstra: shortest distances from ONE vertex to ALL the other vertices

Floyd: shortest distances between EVERY PAIR of vertices

b)

Iteration k compares $d(i,j)$ with $d(i,k) + d(k,j)$

Row k: $i = k$ gives $d(k,k) + d(k,j) = 0 + d(k,j) = d(k,j)$

Column k: $j = k$ gives $d(i,k) + d(k,k) = d(i,k) + 0 = d(i,k)$

The sum equals the current entry, never smaller, so nothing changes

c)

Cubic order: time $\propto n^3$

$$\text{time} = 2.4 \times \left(\frac{100}{40}\right)^3$$

$$\text{time} = 2.4 \times 15.625$$

$$\text{time} = 37.5$$

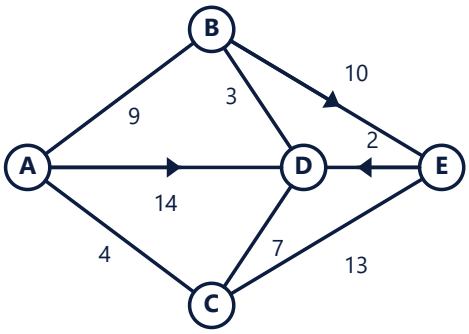
About 37.5 seconds

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

8 The network below shows the cycle routes between five villages, A, B, C, D and E. The number on each arc is the length of the route in km. The routes from A to D, from B to E and from E to D are one-way, as shown by the arrows.

a. Complete the initial distance table and the initial route table.

b. Perform the first two iterations of Floyd's algorithm. You should show the distance table and the route table after each iteration.



Initial tables

Distance table

	A	B	C	D	E
A	-	9	4	14	∞
B	9	-	∞	3	10
C	4	∞	-	7	13
D	∞	3	7	-	∞
E	∞	∞	13	2	-

Route table

	A	B	C	D	E
A	A	B	C	D	E
B	A	B	C	D	E
C	A	B	C	D	E
D	A	B	C	D	E
E	A	B	C	D	E

First iteration

Distance table

	A	B	C	D	E
A	-	9	4	14	∞
B	9	-	**13**	3	10
C	4	**13**	-	7	13
D	∞	3	7	-	∞
E	∞	∞	13	2	-

Route table

	A	B	C	D	E
A	A	B	C	D	E
B	A	B	**A**	D	E
C	A	**A**	C	D	E
D	A	B	C	D	E
E	A	B	C	D	E

Second iteration

Distance table

	A	B	C	D	E
A	-	9	4	**12**	**19**
B	9	-	13	3	10
C	4	13	-	7	13
D	**12**	3	7	-	**13**
E	∞	∞	13	2	-

Route table

	A	B	C	D	E
A	A	B	C	**B**	**B**
B	A	B	A	D	E
C	A	A	C	D	E
D	**B**	B	C	D	**B**
E	A	B	C	D	E

Diagram NOT accurately drawn

(6 marks)

a)

Distance table: the direct distance from the row vertex to the column vertex, ∞ where there is no direct arc

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

A one-way arc only gives an entry in the direction it can be travelled

A: - 9 4 14 ∞

B: 9 - ∞ 3 10

C: 4 ∞ - 7 13

D: ∞ 3 7 - ∞

E: ∞ ∞ 13 2 -

Route table: every entry in column X is X to begin with

A: A B C D E

B: A B C D E

C: A B C D E

D: A B C D E

E: A B C D E

b)

Iteration 1: row A and column A are copied; for every other cell compare (row to A) + (A to column) with the current entry

BC: BA + AC = 9 + 4 = 13 < ∞, so 13, route A

CB: CA + AB = 4 + 9 = 13 < ∞, so 13, route A

All other entries stay the same

Iteration 2: row B and column B are copied; for every other cell compare (row to B) + (B to column) with the current entry

AD: AB + BD = 9 + 3 = 12 < 14, so 12, route B

AE: AB + BE = 9 + 10 = 19 < ∞, so 19, route B

DA: DB + BA = 3 + 9 = 12 < ∞, so 12, route B

DE: DB + BE = 3 + 10 = 13 < ∞, so 13, route B

All other entries stay the same

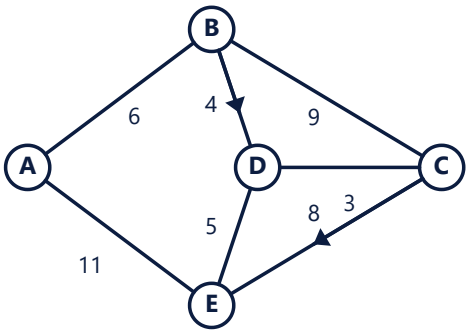
Iteration 1: BC = CB = 13; iteration 2: AD = 12, AE = 19, DA = 12, DE = 13

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

9 The network below shows the times, in minutes, to drive along the roads between five depots, A, B, C, D and E. The roads from B to D and from C to E are one-way. Floyd's algorithm has been applied to the network, with the vertices taken in the order A, B, C, D, E. The final distance table and a partially completed final route table are shown below.

a. Complete the final route table, explaining how you found each missing entry.

b. Use the final route table to find the quickest route from E to C and its time.



Final tables

Distance table

	A	B	C	D	E
A	-	6	13	10	11
B	6	-	7	4	9
C	15	9	-	3	8
D	16	12	3	-	5
E	11	17	8	5	-

Route table

	A	B	C	D	E
A	A	B	D	B	E
B	A	B	D	D	D
C	B	B	C	D	E
D	E	C	C	D	E
E	A	A	D	D	E

Diagram NOT accurately drawn

(6 marks)

a)

Find the shortest route that gives the final distance; the route entry is the vertex of the last iteration that reduced the entry, which for a unique shortest route is its intermediate vertex latest in the order A to E

AC = 13 = AB + BD + DC = 6 + 4 + 3

Route ABDC: intermediate vertices B then D, iteration D came last, so entry D

CA = 15 = CB + BA = 9 + 6

Route CBA: only intermediate vertex B, so entry B

DA = 16 = DE + EA = 5 + 11

Route DEA: only intermediate vertex E, so entry E

EB = 17 = EA + AB = 11 + 6

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Route EAB: only intermediate vertex A; iteration A sets EB = 17, route A

EOCB = 5 + 3 + 9 = 17 is also shortest, but in iteration O, EO + OB = 5 + 12 = 17 is not smaller than 17, so no change

So entry A

AC: O, CA: B, OA: E, EB: A

b)

Read row E, column C of the final route table and split the route at that vertex until every piece is direct

Row E, column C is O: the route from E to C goes via O

Row E, column O is O: E to O is direct

Row O, column C is C: O to C is direct

Length from the final distance table: row E, column C = 8

5 + 3 = 8

EOC, 8 minutes

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

10 A council runs a gritting service for six villages, A, B, C, D, E and F. Some of the roads between the villages are one-way. Floyd's algorithm has been used to find the shortest driving times, in minutes, between the villages. The final distance table and final route table are shown below.

The council will keep its gritting lorry at one of the villages. Each call-out is a return journey from that village to one other village and back again.

a. Explain why the time from A to C is not the same as the time from C to A.

b. Find the village at which the lorry should be kept so that the longest return journey is as short as possible. You must show your working.

c. For the village chosen in (b), find the route of its longest return journey.

Final tables

Distance table

	A	B	C	D	E	F
A	-	14	9	17	20	20
B	14	-	7	15	18	24
C	21	7	-	8	11	17
D	29	15	8	-	6	9
E	30	21	14	6	-	10
F	20	12	19	16	10	-

Route table

	A	B	C	D	E	F
A	A	B	C	C	C	F
B	A	B	C	C	C	D
C	B	B	C	D	E	D
D	C	C	C	D	E	F
E	F	D	D	D	E	F
F	A	B	B	E	E	F

(8 marks)

a)

AC = 9 but CA = 21

The road from A to C is one-way, so a journey from C to A has to use other roads

b)

Return journey X to Y and back = d(X,Y) + d(Y,X); add the table to its reflection

A: AB 28, AC 30, AD 46, AE 50, AF 40; Largest 50

B: BA 28, BC 14, BD 30, BE 39, BF 36; Largest 39

C: CA 30, CB 14, CD 16, CE 25, CF 36; Largest 36

D: DA 46, DB 30, DC 16, DE 12, DF 25; Largest 46

E: EA 50, EB 39, EC 25, ED 12, EF 20; Largest 50

F: FA 40, FB 36, FC 36, FD 25, FE 20; Largest 40

Smallest of the largest values: 36 at C

Keep the lorry at C (Longest return journey 36 minutes)

c)

The longest return journey from C is to F: 17 + 19 = 36

Read row C, column F of the final route table and split the route at that vertex until every piece is direct

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Row C, column F is O: the route from C to F goes via O

Row C, column O is O: C to O is direct

Row O, column F is F: O to F is direct

Length from the final distance table: row C, column F = 17

Read row F, column C of the final route table and split the route at that vertex until every piece is direct

Row F, column C is B: the route from F to C goes via B

Row F, column B is B: F to B is direct

Row B, column C is C: B to C is direct

Length from the final distance table: row F, column C = 19

C O F B C (36 minutes)

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

11 Floyd's algorithm has been applied to a network of roads between six towns, A, B, C, D, E and F. Some of the roads are one-way. The final table of shortest times, in minutes, and the final route table are shown below.

Leila is a courier based in town A. She must deliver parcels to D and to E, in either order, and then return to A.

a. Find the order in which Leila should make her deliveries so that her total time is as short as possible. State the total time.

b. Use the final route table to write down the complete route Leila should take, listing every town she passes through.

c. Explain why her route passes through town F twice.

Final tables

Distance table

	A	B	C	D	E	F
A	-	12	7	15	22	11
B	12	-	6	9	17	22
C	7	6	-	10	15	18
D	17	16	10	-	8	13
E	16	21	15	8	-	5
F	11	20	14	4	12	-

Route table

	A	B	C	D	E	F
A	A	B	C	(F)	C	(F)
B	A	B	C	D	D	E
C	A	B	C	D	E	A
D	C	C	C	D	(E)	E
E	(F)	C	C	D	E	(F)
F	(A)	D	D	(D)	D	F

(8 marks)

a)

The table is not symmetric, so the two orders must both be checked

A→D→E→A = 15 + 8 + 16 = 39

A→E→D→A = 22 + 8 + 17 = 47

D then E, 39 minutes

b)

Read row A, column D of the final route table and split the route at that vertex until every piece is direct

Row A, column D is F: the route from A to D goes via F

Row A, column F is F: A to F is direct

Row F, column D is D: F to D is direct

Length from the final distance table: row A, column D = 15

Read row D, column E of the final route table and split the route at that vertex until every piece is direct

Row D, column E is E: D to E is direct

Length from the final distance table: row D, column E = 8

Read row E, column A of the final route table and split the route at that vertex until every piece is direct

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Row E, column A is F: the route from E to A goes via F

Row E, column F is F: E to F is direct

Row F, column A is A: F to A is direct

Length from the final distance table: row E, column A = 16

Join the stages: AFO + OE + EFA

A F O E F A

c)

A to O: quickest is A F O (11 + 4 = 15)

E to A: quickest is E F A (5 + 11 = 16)

Both stages use F, so F is visited twice

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

12 The network below shows the footpaths between five viewpoints, A, B, C, D and E, in a country park. The number on each arc is the walking time in minutes. The paths from A to C, from C to D and from E to A are one-way, as shown by the arrows.

a. Complete the initial distance table and the initial route table.

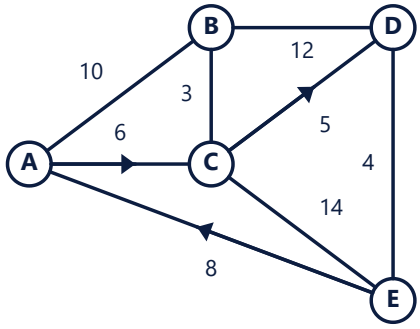
b. Perform the first three iterations of Floyd's algorithm. You should show the distance table and the route table after each iteration.

The final tables, after all five iterations, are shown below the tables for the iterations.

c. Use the final route table to find the quickest route from A to E and from E to B. State the time of each.

d. A new two-way path is to be built directly between B and E. Find the range of walking times for this path that would reduce the quickest time from B to E.

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.



Initial tables

Distance table

	A	B	C	D	E
A	-	10	6	∞	∞
B	10	-	3	12	∞
C	∞	3	-	5	14
D	∞	12	∞	-	4
E	8	∞	14	4	-

Route table

	A	B	C	D	E
A	A	B	C	D	E
B	A	B	C	D	E
C	A	B	C	D	E
D	A	B	C	D	E
E	A	B	C	D	E

(12 marks)

First iteration

Distance table

	A	B	C	D	E
A	-	10	6	∞	∞
B	10	-	3	12	∞
C	∞	3	-	5	14
D	∞	12	∞	-	4
E	8	**18**	14	4	-

Route table

	A	B	C	D	E
A	A	B	C	D	E
B	A	B	C	D	E
C	A	B	C	D	E
D	A	B	C	D	E
E	A	**A**	C	D	E

Second iteration

Distance table

	A	B	C	D	E
A	-	10	6	**22**	∞
B	10	-	3	12	∞
C	**13**	3	-	5	14
D	**22**	12	**15**	-	4
E	8	18	14	4	-

Route table

	A	B	C	D	E
A	A	B	C	**B**	E
B	A	B	C	D	E
C	**B**	B	C	D	E
D	**B**	B	**B**	D	E
E	A	A	C	D	E

Third iteration

Distance table

	A	B	C	D	E
A	-	**9**	6	**11**	**20**
B	10	-	3	**8**	**17**
C	13	3	-	5	14
D	22	12	15	-	4
E	8	**17**	14	4	-

Route table

	A	B	C	D	E
A	A	**C**	C	**C**	**C**
B	A	B	C	**C**	**C**
C	B	B	C	D	E
D	B	B	B	D	E
E	A	**C**	C	D	E

Final tables

Distance table

	A	B	C	D	E
A	-	9	6	11	15
B	10	-	3	8	17

Route table

	A	B	C	D	E
A	A	C	C	C	D
B	A	B	C	C	C

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

B	10	-	3	8	12
C	13	3	-	5	9
D	12	12	15	-	4
E	8	16	14	4	-

B	A	B	C	C	D
C	B	B	C	(D)	D
D	E	(B)	B	D	(E)
E	A	(D)	C	(D)	E

Diagram NOT accurately drawn

a)

Distance table: the direct distance from the row vertex to the column vertex, ∞ where there is no direct arc

A one-way arc only gives an entry in the direction it can be travelled

A: - 10 6 ∞ ∞

B: 10 - 3 12 ∞

C: ∞ 3 - 5 14

D: ∞ 12 ∞ - 4

E: 8 ∞ 14 4 -

Route table: every entry in column X is X to begin with

A: A B C D E

B: A B C D E

C: A B C D E

D: A B C D E

E: A B C D E

b)

Iteration 1: row A and column A are copied; for every other cell compare (row to A) + (A to column) with the current entry

EB: EA + AB = 8 + 10 = 18 < ∞, so 18, route A

All other entries stay the same

Iteration 2: row B and column B are copied; for every other cell compare (row to B) + (B to column) with the current entry

AD: AB + BD = 10 + 12 = 22 < ∞, so 22, route B

CA: CB + BA = 3 + 10 = 13 < ∞, so 13, route B

DA: DB + BA = 12 + 10 = 22 < ∞, so 22, route B

DC: DB + BC = 12 + 3 = 15 < ∞, so 15, route B

All other entries stay the same

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

Iteration 3: row C and column C are copied; for every other cell compare (row to C) + (C to column) with the current entry

AB: AC + CB = 6 + 3 = 9 < 10, so 9, route C

AD: AC + CD = 6 + 5 = 11 < 22, so 11, route C

AE: AC + CE = 6 + 14 = 20 < ∞, so 20, route C

BD: BC + CD = 3 + 5 = 8 < 12, so 8, route C

BE: BC + CE = 3 + 14 = 17 < ∞, so 17, route C

EB: EC + CB = 14 + 3 = 17 < 18, so 17, route C

All other entries stay the same

c)

Read row A, column E of the final route table and split the route at that vertex until every piece is direct

Row A, column E is D: the route from A to E goes via D

Row A, column D is C: the route from A to D goes via C

Row A, column C is C: A to C is direct

Row C, column D is D: C to D is direct

Row D, column E is E: D to E is direct

Length from the final distance table: row A, column E = 15

6 + 5 + 4 = 15

ACDE, 15 minutes

Read row E, column B of the final route table and split the route at that vertex until every piece is direct

Row E, column B is D: the route from E to B goes via D

Row E, column D is D: E to D is direct

Row D, column B is B: D to B is direct

Length from the final distance table: row E, column B = 16

4 + 12 = 16

EDB, 16 minutes

d)

Final distance table: row B, column E = 12

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.

The new path is used only if it is quicker than the current best

Walking time < 12 minutes

Free for classroom and home use: print, photocopy and share. Not for resale or re-hosting.