Allocation problems Mixed exercise

1 $\begin{pmatrix} 322 & 326 & 326 & 328 \\ 318 & 325 & 324 & 325 \\ 315 & 319 & 317 & 320 \\ 323 & 322 & 319 & 321 \end{pmatrix}$ reducing rows $\begin{pmatrix} 0 & 4 & 4 & 6 \\ 0 & 7 & 6 & 7 \\ 0 & 4 & 2 & 5 \\ 4 & 3 & 0 & 2 \end{pmatrix}$

reducing columns $\begin{pmatrix} 0 & 1 & 4 & 4 \\ 0 & 4 & 6 & 5 \\ 0 & 1 & 2 & 3 \\ 4 & 0 & 0 & 0 \end{pmatrix}$

Minimum uncovered element is 1 $\begin{pmatrix} 0 & 0 & 3 & 3 \\ 0 & 3 & 5 & 4 \\ 0 & 0 & 1 & 2 \\ 5 & 0 & 0 & 0 \end{pmatrix}$

Minimum uncovered element is 1 $\begin{pmatrix} 0 & 0 & 2 & 2 \\ 0 & 3 & 4 & 3 \\ 0 & 0 & 0 & 1 \\ 6 & 1 & 0 & 0 \end{pmatrix}$

Bring-it – Depot (326)

Collect-it – Airport (318)

Fetch-it – Docks (317) Cost: £1282

Haul-it – Station (321)

2 a $\begin{pmatrix} 18 & 20 & 19 & 14 \\ 19 & 21 & 19 & 14 \\ 17 & 20 & 20 & 16 \\ 20 & 21 & 20 & 15 \end{pmatrix} \rightarrow$ reducing rows $\begin{pmatrix} 4 & 6 & 5 & 0 \\ 5 & 7 & 5 & 0 \\ 1 & 4 & 4 & 0 \\ 5 & 6 & 5 & 0 \end{pmatrix}$

reducing columns $\begin{pmatrix} 3 & 2 & 1 & 0 \\ 4 & 3 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 4 & 2 & 1 & 0 \end{pmatrix}$

Minimum uncovered element is 1 $\begin{pmatrix} 2 & 1 & 0 & 0 \\ 3 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 3 & 1 & 0 & 0 \end{pmatrix}$

Minimum uncovered element is 1 $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 2 & 0 & 0 & 0 \end{pmatrix}$

2 a continued

J – Br (20)	J – C (14)	J – Br (20)	J – Bu (19)
K – Bu (19)	K – Bu (19)	K – Cr (14)	K – Cr (14)
L – Ba (17)	L – Ba (17)	L – Ba (17)	L – Ba (17)
M – C (15)	M – Br (21)	M – Bu (20)	M – Br (21)

b There are 4 solutions each of duration 71 seconds

c An allocation not given in part a above.

3 a

$$\begin{pmatrix} 8 & 19 & 11 & 14 & 12 \\ 12 & 17 & 14 & 18 & 20 \\ 10 & 22 & 18 & 14 & 19 \\ 9 & 15 & 16 & 15 & 21 \\ 14 & 23 & 20 & 20 & 19 \end{pmatrix} \text{reducing rows} \begin{pmatrix} 0 & 11 & 3 & 6 & 4 \\ 0 & 5 & 2 & 5 & 8 \\ 0 & 12 & 8 & 7 & 9 \\ 0 & 6 & 7 & 6 & 12 \\ 0 & 9 & 6 & 6 & 5 \end{pmatrix}$$

$$\text{reducing columns} \begin{pmatrix} 0 & 6 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 & 4 \\ 0 & 7 & 6 & 0 & 5 \\ 0 & 1 & 5 & 2 & 8 \\ 0 & 4 & 4 & 2 & 1 \end{pmatrix}$$

$$\text{Minimum uncovered element is 1} \begin{pmatrix} 0 & 5 & 0 & 2 & 0 \\ 1 & 0 & 0 & 2 & 5 \\ 0 & 6 & 5 & 0 & 5 \\ 0 & 0 & 4 & 2 & 8 \\ 0 & 3 & 3 & 2 & 1 \end{pmatrix}$$

Solution

Alf - Kitchen (12)

Betty - Gallery (14)

Charlie - Bedroom (14)

Donna - Dinning room (15)

Eve - Grand Hall (14) Minimum time 69 minutes

3 b

Subtracting all terms from 23 $\begin{pmatrix} 15 & 4 & 12 & 9 & 11 \\ 11 & 6 & 9 & 5 & 3 \\ 13 & 1 & 5 & 9 & 4 \\ 14 & 8 & 7 & 8 & 2 \\ 9 & 0 & 3 & 3 & 4 \end{pmatrix}$

reducing rows $\begin{pmatrix} 11 & 0 & 8 & 5 & 7 \\ 8 & 3 & 6 & 2 & 0 \\ 12 & 0 & 4 & 8 & 3 \\ 12 & 6 & 5 & 6 & 0 \\ 9 & 0 & 3 & 3 & 4 \end{pmatrix}$ reducing columns $\begin{pmatrix} 3 & 0 & 5 & 3 & 7 \\ 0 & 3 & 3 & 0 & 0 \\ 4 & 0 & 1 & 6 & 3 \\ 4 & 6 & 2 & 4 & 0 \\ 1 & 0 & 0 & 1 & 4 \end{pmatrix}$

Minimum uncovered element is 1 $\begin{pmatrix} 2 & 0 & 5 & 2 & 6 \\ 0 & 4 & 4 & 0 & 0 \\ 3 & 0 & 1 & 5 & 2 \\ 4 & 7 & 3 & 4 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{pmatrix}$

Minimum uncovered element is 1 $\begin{pmatrix} 1 & 0 & 4 & 1 & 6 \\ 0 & 5 & 4 & 0 & 1 \\ 2 & 0 & 0 & 4 & 2 \\ 3 & 7 & 2 & 3 & 0 \\ 0 & 1 & 0 & 0 & 4 \end{pmatrix}$

Solutions

Alf – Dining room (19)		Alf – Dining room (19)
Betty – Hall (12)		Betty – Bedroom (18)
Charlie – Gallery (18)	or	Charlie – Gallery (18)
Donna – Kitchen (21)		Donna – Kitchen (21)
Eve – Bedroom (20)		Eve – Hall (14)

Maximum time 90 minutes

4 a There are 4 chauffeurs but only 3 tasks.

b $\begin{pmatrix} 245 & 378 & 459 & 0 \\ 250 & 387 & 467 & 0 \\ 224 & 350 & 442 & 0 \\ 231 & 364 & 453 & 0 \end{pmatrix}$ reducing columns $\begin{pmatrix} 21 & 38 & 17 & 0 \\ 26 & 37 & 25 & 0 \\ 0 & 0 & 0 & 0 \\ 7 & 14 & 11 & 0 \end{pmatrix}$

Minimum uncovered element is 7 $\begin{pmatrix} 14 & 21 & 10 & 0 \\ 19 & 30 & 18 & 0 \\ 0 & 0 & 0 & 7 \\ 0 & 7 & 4 & 0 \end{pmatrix}$

Minimum uncovered element is 10 $\begin{pmatrix} 4 & 11 & 0 & 0 \\ 9 & 20 & 8 & 0 \\ 0 & 0 & 0 & 17 \\ 0 & 7 & 4 & 10 \end{pmatrix}$

Solution:

D – Party (459)

E – Dummy Cost: £1040

F – Film (350)

G – Award (231)

5 The initial cost matrix is shown below:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	No	863	636	628	739	634
Green	562	796	583	478	674	No
Orange	No	825	672	583	756	710
Red	635	881	650	538	No	685
Teal	688	934	No	554	No	742
Yellow	624	835	580	No	712	No

In the matrix a 'No' indicates that the company cannot offer the service. These are known as forbidden allocations. Therefore, for all such entries we replace the 'No' with the value of infinity (∞) to make the assignments 'unattractive'. Therefore, the matrix becomes:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	863	636	628	739	634
Green	562	796	583	478	674	∞
Orange	∞	825	672	583	756	710
Red	635	881	650	538	∞	685
Teal	688	934	∞	554	∞	742
Yellow	624	835	580	∞	712	∞

As the table above is an n by n matrix (with $n = 6$), we do not need to add any additional dummy rows or columns.

5 continued

The smallest numbers in rows 1, 2, 3, 4, 5 and 6 are 628, 478, 583, 538, 554 and 580. We reduce rows first by subtracting these numbers from each element in the row. The forbidden allocations remain unaltered by row and column operations and always have the value of ∞. The table becomes:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	235	8	0	111	6
Green	84	318	105	0	196	∞
Orange	∞	242	89	0	173	127
Red	97	343	112	0	∞	147
Teal	134	380	∞	0	∞	188
Yellow	44	255	0	∞	132	∞

The smallest numbers in columns 1, 2, 3, 4, 5 and 6 are 44, 235, 0, 0, 111 and 6. We reduce columns by subtracting these numbers from each element in the column. Therefore, the reduced cost matrix is:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	0	0	0
Green	40	83	105	0	85	∞
Orange	∞	7	89	0	62	121
Red	53	108	112	0	∞	141
Teal	90	145	∞	0	∞	182
Yellow	0	20	0	∞	21	∞

We now apply the Hungarian algorithm.
We can cover all the zeros in three lines, so the solution is not optimal:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	0	0	0
Green	40	83	105	0	85	∞
Orange	∞	7	89	0	62	121
Red	53	108	112	0	∞	141
Teal	90	145	∞	0	∞	182
Yellow	0	20	0	∞	21	∞

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	0	0	0
Green	40	83	105	0	85	∞
Orange	∞	7	89	0	62	121
Red	53	108	112	0	∞	141
Teal	90	145	∞	0	∞	182
Yellow	0	20	0	∞	21	∞

5 continued

The smallest uncovered element is 7, so we augment the matrix as follows:

- Add 7 to the elements covered by two lines circled in red
- Leave the elements covered by just one line unchanged
- Subtract 7 from the uncovered elements

This gives the following matrix:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	7	0	0
Green	33	76	98	0	78	∞
Orange	∞	0	82	0	55	114
Red	46	101	105	0	∞	134
Teal	83	138	∞	0	∞	175
Yellow	0	20	0	∞	21	∞

We can cover all the zeros in four lines, so the solution is not optimal:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	7	0	0
Green	33	76	98	0	78	∞
Orange	∞	0	82	0	55	114
Red	46	101	105	0	∞	134
Teal	83	138	∞	0	∞	175
Yellow	0	20	0	∞	21	∞

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	7	0	0
Green	33	76	98	0	78	∞
Orange	∞	0	82	0	55	114
Red	46	101	105	0	∞	134
Teal	83	138	∞	0	∞	175
Yellow	0	20	0	∞	21	∞

The smallest uncovered element is 33, so we augment the matrix as follows:

- Add 33 to the elements covered by two lines circled in red
- Leave the elements covered by just one line unchanged
- Subtract 33 from the uncovered elements

5 continued

This gives the following matrix:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	40	0	0
Green	0	43	65	0	45	∞
Orange	∞	0	82	33	55	114
Red	13	68	72	0	∞	101
Teal	50	105	∞	0	∞	142
Yellow	0	20	0	∞	21	∞

We can cover all the zeros in five lines, so the solution is not optimal:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	40	0	0
Green	0	43	65	0	45	∞
Orange	∞	0	82	33	55	114
Red	13	68	72	0	∞	101
Teal	50	105	∞	0	∞	142
Yellow	0	20	0	∞	21	∞

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	0	8	40	0	0
Green	0	43	65	0	45	∞
Orange	∞	0	82	33	55	114
Red	13	68	72	0	∞	101
Teal	50	105	∞	0	∞	142
Yellow	0	20	0	∞	21	∞

The smallest uncovered element is 45, so we augment the matrix as follows:

- Add 45 to the elements covered by two lines circled in red
- Leave the elements covered by just one line unchanged
- Subtract 45 from the uncovered elements

This gives the following matrix:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	45	8	85	0	0
Green	0	43	20	0	0	∞
Orange	∞	0	37	33	10	69
Red	13	68	27	0	∞	56
Teal	50	105	∞	0	∞	97
Yellow	45	65	0	∞	21	∞

5 continued

We can cover all the zeros in five lines, so the solution is not optimal:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	45	8	85	0	0
Green	0	43	20	0	0	∞
Orange	∞	0	37	33	10	69
Red	13	68	27	0	∞	56
Teal	50	105	∞	0	∞	97
Yellow	45	65	0	∞	21	∞

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	45	8	85	0	0
Green	0	43	20	0	0	∞
Orange	∞	0	37	33	10	69
Red	13	68	27	0	∞	56
Teal	50	105	∞	0	∞	97
Yellow	45	65	0	∞	21	∞

The smallest uncovered element is 13, so we augment the matrix as follows:

- Add 13 to the elements covered by two lines
- Leave the elements covered by just one line unchanged
- Subtract 13 from the uncovered elements

This gives the following matrix:

	Catering	Cleaning	Computer	Copying	Post	Maintenance
Blue	∞	45	21	98	0	0
Green	0	43	33	13	0	∞
Orange	∞	0	50	46	10	69
Red	0	55	27	0	∞	43
Teal	37	92	∞	0	∞	84
Yellow	32	52	0	∞	8	∞

We can cover all the zeros with six lines, so we have found our optimal solution of:

Blue – Maintenance (634)
Green – Post (674)
Orange – Cleaning (825)
Red – Catering (635)
Teal – Copying (554)
Yellow – Computer (580)

Minimum cost = 634 + 674 + 825 + 635 + 554 + 580 = £3902.

6 $\begin{pmatrix} 834 & 365 & 580 & 648 \\ 874 & 375 & 2000 & 593 \\ 743 & 289 & 2000 & 665 \\ 899 & 500 & 794 & 2000 \end{pmatrix}$ Subtracting all terms from 1000 $\begin{pmatrix} 166 & 635 & 420 & 352 \\ 126 & 625 & 1000 & 407 \\ 257 & 711 & 1000 & 335 \\ 101 & 500 & 206 & 1000 \end{pmatrix}$

reducing rows $\begin{pmatrix} 0 & 469 & 254 & 186 \\ 0 & 499 & 874 & 281 \\ 0 & 454 & 743 & 78 \\ 0 & 399 & 105 & 899 \end{pmatrix}$ reducing columns $\begin{pmatrix} 0 & 70 & 149 & 108 \\ 0 & 100 & 769 & 203 \\ 0 & 55 & 638 & 0 \\ 0 & 0 & 0 & 821 \end{pmatrix}$

Minimum uncovered element is 55 $\begin{pmatrix} 0 & 15 & 94 & 108 \\ 0 & 45 & 714 & 203 \\ 0 & 0 & 583 & 0 \\ 55 & 0 & 0 & 876 \end{pmatrix}$

Minimum uncovered element is 15 $\begin{pmatrix} 0 & 0 & 79 & 93 \\ 0 & 30 & 699 & 188 \\ 15 & 0 & 583 & 0 \\ 70 & 0 & 0 & 876 \end{pmatrix}$

Ghost train – Coffee shop (365)

Log flume – Cafe (874)

Roller coaster – Snack shop (665)

Teddie's adventure – Restaurant (794)

Profit £2698

7 The initial cost matrix is shown below:

	1	**2**	**3**	**4**	**5**
A	128	142	153	133	155
B	–	138	147	139	147
C	135	144	144	130	158
D	141	156	154	142	–
E	150	141	157	145	160
F	132	149	140	140	157

Introducing a dummy 6^{th} task makes the matrix n by n as follows:

	1	**2**	**3**	**4**	**5**	**6**
A	128	142	153	133	155	0
B	–	138	147	139	147	0
C	135	144	144	130	158	0
D	141	156	154	142	–	0
E	150	141	157	145	160	0
F	132	149	140	140	157	0

7 continued

To maximise the profit with incomplete data, first subtract every number from the largest value in the table. Then replace the '–' forbidden allocations with the value of infinity (∞) to make the assignments 'unattractive'.
The largest value in the table is 160. Subtracting every value from 160 the table becomes:

	1	**2**	**3**	**4**	**5**	**6**
A	32	18	7	27	5	160
B	–	22	13	21	13	160
C	25	16	16	30	2	160
D	19	4	6	18	–	160
E	10	19	3	15	0	160
F	28	11	20	20	3	160

Replacing the '–' entries with the value of ∞ gives:

	1	**2**	**3**	**4**	**5**	**6**
A	32	18	7	27	5	160
B	∞	22	13	21	13	160
C	25	16	16	30	2	160
D	19	4	6	18	∞	160
E	10	19	3	15	0	160
F	28	11	20	20	3	160

The smallest numbers in rows 1, 2, 3, 4, 5 and 6 are 5, 13, 2, 4, 0 and 3. We reduce rows first by subtracting these numbers from each element in the row. The forbidden allocations remain unaltered by row and column operations and always have a value of ∞. The table becomes:

	1	**2**	**3**	**4**	**5**	**6**
A	27	13	2	22	0	155
B	∞	9	0	8	0	147
C	23	14	14	28	0	158
D	15	0	2	14	∞	156
E	10	19	3	15	0	160
F	25	8	17	17	0	157

The smallest numbers in columns 1, 2, 3, 4, 5 and 6 are 10, 0, 0, 8, 0 and 147. We reduce columns by subtracting these numbers from each element in the column. Therefore, the reduced cost matrix is:

7 continued

	1	**2**	**3**	**4**	**5**	**6**
A	17	13	2	14	0	8
B	∞	9	0	0	0	0
C	13	14	14	20	0	11
D	5	0	2	6	∞	9
E	0	19	3	7	0	13
F	15	8	17	9	0	10

We now apply the Hungarian algorithm.
We can cover all the zeros in four lines, so the solution is not optimal:

	1	**2**	**3**	**4**	**5**	**6**
A	17	13	2	14	0	8
B	∞	9	0	0	0	0
C	13	14	14	20	0	11
D	5	0	2	6	∞	9
E	0	19	3	7	0	13
F	15	8	17	9	0	10

	1	**2**	**3**	**4**	**5**	**6**
A	17	13	2	14	0	8
B	∞	9	0	0	0	0
C	13	14	14	20	0	11
D	5	0	2	6	∞	9
E	0	19	3	7	0	13
F	15	8	17	9	0	10

The smallest uncovered element is 2, so we augment the matrix as follows:

- Add 2 to the elements covered by two lines circled in red
- Leave the elements covered by just one line unchanged
- Subtract 2 from the uncovered elements

This gives the following matrix:

	1	**2**	**3**	**4**	**5**	**6**
A	17	13	0	12	0	6
B	∞	11	0	0	2	0
C	13	14	12	18	0	9
D	5	0	0	4	∞	7
E	0	19	1	5	0	11
F	15	8	15	7	0	8

7 continued

We can cover all the zeros in five lines, so the solution is not optimal:

	1	**2**	**3**	**4**	**5**	**6**
A	17	13	0	12	0	6
B	∞	11	0	0	2	0
C	13	14	12	18	0	9
D	5	0	0	4	∞	7
E	0	19	1	5	0	11
F	15	8	15	7	0	8

	1	**2**	**3**	**4**	**5**	**6**
A	17	13	0	12	0	6
B	∞	11	0	0	2	0
C	13	14	12	18	0	9
D	5	0	0	4	∞	7
E	0	19	1	5	0	11
F	15	8	15	7	0	8

The smallest uncovered element is 6, so we augment the matrix as follows:

- Add 6 to the elements covered by two lines circled in red
- Leave the elements covered by just one line unchanged
- Subtract 6 from the uncovered elements

This gives the following matrix:

	1	**2**	**3**	**4**	**5**	**6**
A	11	7	0	6	0	0
B	∞	11	6	0	8	0
C	7	8	12	12	0	3
D	5	0	6	4	∞	7
E	0	19	7	5	6	11
F	9	2	15	1	0	2

We can cover all the zeros in five lines, so the solution is not optimal:

	1	**2**	**3**	**4**	**5**	**6**
A	11	7	0	6	0	0
B	∞	11	6	0	8	0
C	7	8	12	12	0	3
D	5	0	6	4	∞	7
E	0	19	7	5	6	11
F	9	2	15	1	0	2

7 continued

	1	**2**	**3**	**4**	**5**	**6**
A	11	7	0	6	0	0
B	∞	11	6	0	8	0
C	7	8	12	12	0	3
D	5	0	6	4	∞	7
E	0	19	7	5	6	11
F	9	2	15	1	0	2

The smallest uncovered element is 1, so we augment the matrix as follows:

- Add 1 to the elements covered by two lines circled in red
- Leave the elements covered by just one line unchanged
- Subtract 1 from the uncovered elements

This gives the following matrix:

	1	**2**	**3**	**4**	**5**	**6**
A	12	8	0	6	1	0
B	∞	12	6	0	9	0
C	7	8	11	11	0	2
D	5	0	5	3	∞	6
E	0	19	6	4	6	10
F	9	2	14	0	0	1

We can cover all the zeros with six lines, so we have found our optimal solution of:

A – 3 (153)
B – 6 (0)
C – 5 (158)
D – 2 (156)
E – 1 (150)
F – 4 (140)

Maximum profit = 153 + 0 + 158 + 156 + 150 + 140 = £757

8 $\begin{pmatrix} 13 & 17 & 15 & 18 \\ 15 & 19 & 12 & 19 \\ 16 & 20 & 13 & 22 \\ 14 & 15 & 17 & 24 \end{pmatrix}$

Let x_{ij} be 0 or 1

$$x_{ij} \begin{cases} 1 \text{ if worker } i \text{ does task } j \\ 0 \text{ otherwise} \end{cases}$$

where $i \in \{\text{P, Q, R, S}\}$ and $j \in \{\text{A, B, C, D}\}$

$$\begin{aligned} \text{Maximise } P = & 13x_{\text{PA}} + 17x_{\text{PB}} + 15x_{\text{PC}} + 18x_{\text{PD}} \\ & + 15x_{\text{QA}} + 19x_{\text{QB}} + 12x_{\text{QC}} + 19x_{\text{QD}} \\ & + 16x_{\text{RA}} + 20x_{\text{RB}} + 13x_{\text{RC}} + 22x_{\text{RD}} \\ & + 14x_{\text{SA}} + 15x_{\text{SB}} + 17x_{\text{SC}} + 14x_{\text{SD}} \end{aligned}$$

$$\begin{aligned} \text{Subject to: } & \sum x_{\text{Pj}} = 1 \\ & \sum x_{\text{Qj}} = 1 \\ & \sum x_{\text{Rj}} = 1 \\ & \sum x_{\text{Sj}} = 1 \\ & \sum x_{\text{iA}} = 1 \\ & \sum x_{\text{iB}} = 1 \\ & \sum x_{\text{iC}} = 1 \\ & \sum x_{\text{iD}} = 1 \end{aligned}$$

9 The initial cost matrix is shown below (with the numbers representing costs in £):

	1	**2**	**3**	**4**
P	143	243	247	475
Q	132	238	–	437
R	126	207	197	408
S	–	222	238	445

Replacing the '–' entries with the value of ∞ gives:

	1	**2**	**3**	**4**
P	143	243	247	475
Q	132	238	∞	437
R	126	207	197	408
S	∞	222	238	445

9 continued

The following linear programming problem can be formulated to minimise the total cost.

Decision variables:

Let $x_{ij} = \begin{cases} 1 \text{ if worker } i \text{ does task } j \\ 0 \text{ otherwise} \end{cases}$

where $i \in \{P, Q, R, S\}$ and $j \in \{1, 2, 3, 4\}$

Objective function:
To minimise the total cost P.

$$\begin{aligned} \text{Minimise } P = & \ 143x_{P1} + 243x_{P2} + 247x_{P3} + 475x_{P4} \\ & + 132x_{Q1} + 238x_{Q2} + \infty x_{Q3} + 437x_{Q4} \\ & + 126x_{R1} + 207x_{R2} + 197x_{R3} + 408x_{R4} \\ & + \infty x_{S1} + 222x_{S2} + 238x_{S3} + 445x_{S4} \end{aligned}$$

Subject to the following constraints:

Each worker can be assigned to at most one task:

$\sum x_{Pj} = 1, \sum x_{Qj} = 1, \sum x_{Rj} = 1, \sum x_{Sj} = 1$

Each task must be done by just one worker:

$\sum x_{i1} = 1, \sum x_{i2} = 1, \sum x_{i3} = 1, \sum x_{i4} = 1$

Challenge

The initial cost matrix is shown below (with the numbers representing costs in £):

	A	***B***	***C***	***D***	***E***	***F***	***G***	***H***
W	35	41	28	52	–	51	74	48
X	51	39	40	55	42	50	63	54
Y	38	45	39	50	48	47	65	50
Z	47	–	48	51	45	53	64	52

Replacing the '–' entries with the value of ∞ gives:

	A	***B***	***C***	***D***	***E***	***F***	***G***	***H***
W	35	41	28	52	∞	51	74	48
X	51	39	40	55	42	50	63	54
Y	38	45	39	50	48	47	65	50
Z	47	∞	48	51	45	53	64	52

The following linear programming problem can be formulated to minimise the total cost.

Decision variables:

Let $x_{ij} = \begin{cases} 1 \text{ if worker } i \text{ does task } j \\ 0 \text{ otherwise} \end{cases}$

where $i \in \{W, X, Y, Z\}$ and $j \in \{A, B, C, D, E, F, G, H\}$

Objective function:
To minimise the total cost P.

$$\begin{aligned} \text{Minimise } P = {} & 35x_{WA} + 41x_{WB} + 28x_{WC} + 52x_{WD} + \infty x_{WE} + 51x_{WF} + 74x_{WG} + 48x_{WH} \\ & + 51x_{XA} + 39x_{XB} + 40x_{XC} + 55x_{XD} + 42x_{XE} + 50x_{XF} + 63x_{XG} + 54x_{XH} \\ & + 38x_{YA} + 45x_{YB} + 39x_{YC} + 50x_{YD} + 48x_{YE} + 47x_{YF} + 65x_{YG} + 50x_{YH} \\ & + 47x_{ZA} + \infty x_{ZB} + 48x_{ZC} + 51x_{ZD} + 45x_{ZE} + 53x_{ZF} + 64x_{ZG} + 52x_{ZH} \end{aligned}$$

Subject to the following constraints:
No worker can do more than 3 tasks:

$\sum x_{Wj} \leqslant 3,\ \sum x_{Xj} \leqslant 3,\ \sum x_{Yj} \leqslant 3,\ \sum x_{Zj} \leqslant 3$

Each task must be assigned to one worker:

$\sum x_{iA} = 1,\ \sum x_{iB} = 1,\ \sum x_{iC} = 1,\ \sum x_{iD} = 1,\ \sum x_{iE} = 1,\ \sum x_{iF} = 1,\ \sum x_{iG} = 1,\ \sum x_{iH} = 1$