E-fólio B | Instruções para a realização do E-fólio

INTRODUCTION TO ARTIFICIAL INTELLIGENCE | 21071 | 2023/2024

Folio B is inspired by the railway boom in the Wild West. The introduction of railways transformed the landscape, replacing horses as the main means of transport and facilitating the movement of people and goods. This technological advance not only brought communities closer together, but also promoted territorial expansion.

Let's consider territories of NxM zones, with the number of families in each zone, in the same way as in e-folio A. We need to identify the locations of the stations in order to best serve the population. The route of the railway to connect the stations, as well as the connection to other territories, is up to the railway company's engineers, once the location of the stations has been decided.

After much consideration, the following criteria were used to assess station locations, to be minimised:

- A - number of stations
- B - average cost of travelling to the nearest station

For travel costs, the following unit costs were adopted, based on the distance (horizontal, vertical and diagonal movements) to the nearest station:

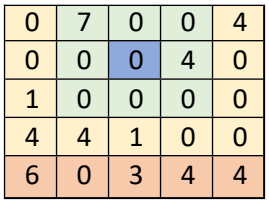
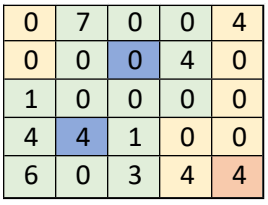
Distance	0 or 1	2	3	4	5	6 or higher
Cost	0	1	2	4	8	10

The aim is to minimise the number of stations first, and only then the average travel cost. However, we don't want solutions whose average travel cost is equal to or greater than 3. Therefore, a single formula was defined to minimise the cost of the solution, rounded down to the nearest integer:

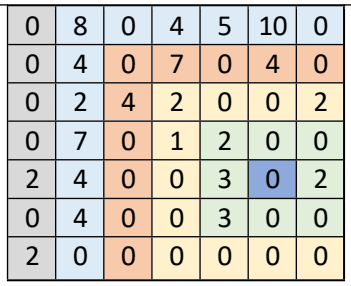
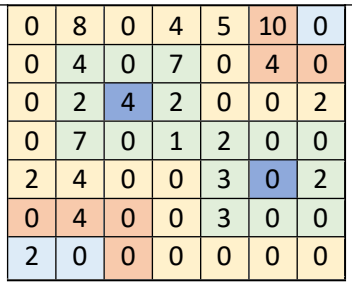
Minimise Cost = 1000A + 100B

As the value of B will not be greater than 3, the three least significant digits will not be greater than 300, so this formula allows you to do what you want, minimise the number of stations, and between solutions with the same number of stations, minimise the average cost of travel. The distance is the shortest path between zones, and a move can take place from one zone to any of the adjacent ones (connected by side or corner).

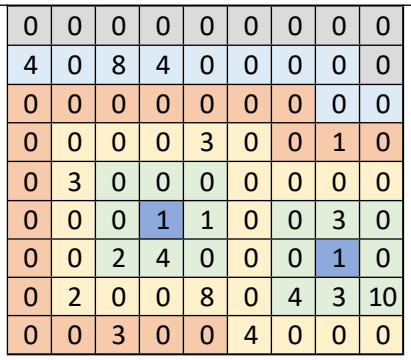
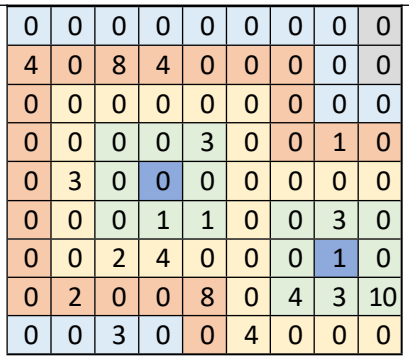
Example instance 1 (two solutions):

 Cost: 1114	 Cost: 2064
In this solution for this territory there is only one station in the blue zone. The green zone is 1 kilometre away, so there is no travel cost. The yellow zones are 2 zones away from the station and have a unit cost of 1. As there are 14 families, they contribute to a total travel cost of 14. The orange zones are 3 zones away from the station and have a unit cost of 2. As there are 17 families, the total cost is 34. Adding up the travel costs gives 48. As there are 42 families, the average travel cost is **B=48/42=1.142**. So, as there is one station, **A=1,** the value of the solution according to the single formula to be minimised is 1000*1+1.**142*100=1114**. The solution is valid because the average cost of travelling is less than 3.	By adding one more station to the solution next to it, we get 8 families in yellow areas, with a travel cost of 8. In orange there are only 4 families, with a unit cost of 2, so they have a travel cost of 8. So the total travel cost is 16, divided by 25 gives **B=0.**64. As there are two stations, **A=2,** the cost of the solution is **2064**. This change is not valued, as the cost of the solution has increased compared to the solution next door. 16/42=0.38 being the cost 2038

Example instance 3 (its solutions)

 Cost: 1315	 Cost: 2084
In this solution for this territory there is only one station in the blue zone. The green zone has a cost of 0. In yellow are cost zones 1, with 5 families, with a travel cost of 5. In orange are cost zones 2, with 15 families, contributing 30 to the travel cost. In light blue are cost zones 4, with 48 families. Thus, these zones contribute to the cost of travelling by 192. Finally, the grey areas have a cost of 8, with 4 families contributing 32 to the cost of commuting. This leaves a total travel cost of 259. As there are 82 families, the average travel cost is **B=3.158**. Since this value is equal to or greater than 3, this solution is not a good solution. valid.	Another station was added to the previous solution. This leaves us with 25 families in the yellow zone, with a unit cost of 1. In the orange zone, with a unit cost of 2, we have 18 families, contributing 36 to the cost of travelling. Finally, in the blue zone we only have 2 families, with a unit cost of 4, contributes to a travel cost of 8. The total travel cost is 69, as there are 82 families the average travel cost is **B=0.841**. As there are two stations, **A=2,** leaving the cost of the solution at **2084**. This change is valued, although it has a higher value, it is now a valid solution, while the solution next door is invalid.

Example instance 6 (its solutions)

 Cost: 2133	 Cost: 2114
In this solution for this territory there are two stations in dark blue. We have 20 families in yellow, contributing 20 to the cost of travelling. In orange there are 4 families, contributing 8 to the cost of travelling. In light blue there are zones at cost 4, with 16 families, contributing 64 to the cost of travelling. Finally, in grey there are zones with a cost of 8, but there are no families in these zones. So the total cost of travelling is 92, divided by 69 families in the territory we get an average cost of **B=1.333**. As there are 2 stations, **A=2,** so the cost of the solution is **2133**. The solution is valid given that the average cost is less than 3.	In the solution opposite, we have moved one of the stations up one zone, and there have been changes in the travel costs. In yellow there are 13 families, in orange there are 27 families, with a travel cost of 54, and in blue there are 3 families, giving a travel cost of 12. So the total travel cost is 79, and the average travel cost is **B=1.144**. The cost of the solution therefore stands at **2114**. This change is valued because the cost of the solution has decreased compared to the solution on the other side.

Consider the 20 instances in the annex (10 are the same as e-folio A).

A text format can be used to report a valid solution, as shown below for the examples given above, for the solution on the left:

```
  0   7   0   0   4
  0   0  0#   4   0
  1   0   0   0   0
  4   4   1   0   0
  6   0   3   4   4
 Cost: 1114

  0   8   0   4   5  10   0
  0   4   0   7   0   4   0
  0   2   4   2   0   0   2
  0   7   0   1   2   0   0
  2   4   0   0   3  0#   2
  0   4   0   0   3   0   0
  2   0   0   0   0   0   0
Cost: 1315

  0   0   0   0   0   0   0   0   0
  4   0   8   4   0   0   0   0   0
  0   0   0   0   0   0   0   0   0
  0   0   0   0   3   0   0   1   0
  0   3   0   0   0   0   0   0   0
  0   0   0  1#   1   0   0   3   0
  0   0   2   4   0   0   0  1# 0
  0   2   0   0   8   0   4   3 10
  0   0   3   0   0   4   0   0   0
 Cost: 2133
```

The zones with stations should be marked and the cost calculated according to the formula given. Alternatively, the solutions can be presented graphically, via a spreadsheet, as used in this statement. The format can be changed, as long as the content is maintained: **position of the stations** and calculation of the **cost**.

You must use informed searches to solve the e-folio, using only the algorithms taught. If you use an algorithm that has not been taught, you will be penalised 0.5 points in the algorithms criterion.

You must deliver:

- Report;
- Source code for the implemented algorithms.

The report must contain a table with the results of running the algorithms/configurations tested against the instances provided. For each algorithm/instance it should show:

- Number of reviews (maximum 100,000 reviews);
- Cost;
- Time spent (maximum 1 minute).

For each instance, it must have the best information obtained considering all the executions, namely the value of the best solution. This solution must be presented in an annex, in a format identical to the one used in this statement.

You should set a stop criterion on the time spent and the number of evaluations, so that you get runs with a maximum of 1 minute, and a maximum of 100,000 evaluations (whichever comes first). Naturally, you don't need to check the stop criterion at every instant, so if one of these limits is exceeded slightly, there's no problem.

Template for the results table:

	Instance	1	2	…	20
Algorithm 1 / configurations 1	Reviews				
	Generations				
	Cost				
	Time (msec)				
…					
	Best result				

The best solution obtained for each solved instance must be attached, in a format identical to that presented for the example instance (you can omit the colours).

Correction criteria (4 points):

- **Analysis** (1 value): Reference to important aspects of the problem in the report, revealing whether or not you implemented them, that you were aware of them, as well as the choices made in implementation and their justification.
- **Algorithms** (1 value): Clear identification of the algorithms you have implemented according to the nomenclature of the book and the CU, together with the configurations used. Case

does not have a functional implementation, or has implemented an algorithm not taught, or has realised the e-folio manually, this criterion can be valued at half, depending on the work done.

- **Results** (2 values): This criterion is only assessed if there is a table of results, and the solutions are attached in the report, with each instance worth 0.1 values (0.05 for instances with solutions costing 1000, above the best solution). An instance is considered solved if a solution is obtained by an algorithm, even if executed manually, and the solution is correctly attached. In instances with ID 11 to 20 in which the best solution has been obtained from among the valid solutions submitted, the instance is valued at 0.15 values, up to a maximum of 2 values for this criterion.

The work is individual, but if students wish, they can share the final cost results of the solutions obtained, but not the solutions themselves, nor the algorithms and approaches used.

Appendix - instances to be used in the e-folio

ID1

0	7	0	0	4
0	0	0	4	0
1	0	0	0	0
4	4	1	0	0
6	0	3	4	4

ID2

4	0	0	10	1
1	0	0	0	0
0	0	1	6	3
0	4	0	0	2
8	0	6	3	0

ID3

0	8	0	4	5	10	0
0	4	0	7	0	4	0
0	2	4	2	0	0	2
0	7	0	1	2	0	0
2	4	0	0	3	0	2
0	4	0	0	3	0	0
2	0	0	0	0	0	0

ID4

0	0	1	0	7	0	1
0	1	4	0	0	0	4
0	0	0	0	2	0	0
3	1	0	8	5	7	7
0	4	0	3	0	0	0
0	0	0	3	2	4	2
0	8	3	6	3	0	0

ID5

6	7	2	0	0	0	0	0	0
3	3	6	0	8	4	3	1	0
0	0	8	0	0	0	2	4	0
0	0	0	1	0	3	2	0	0
0	0	0	7	4	0	1	0	0
12	8	0	5	4	1	4	3	4
8	0	1	2	4	3	3	0	0
1	1	0	0	0	0	5	0	0
4	0	0	0	4	6	0	13	2

ID6

0	0	0	0	0	0	0	0	0
4	0	8	4	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	3	0	0	1	0
0	3	0	0	0	0	0	0	0
0	0	0	1	1	0	0	3	0
0	0	2	4	0	0	0	1	0
0	2	0	0	8	0	4	3	10
0	0	3	0	0	4	0	0	0

ID7

0	0	0	0	0	3	0	0	0	0	0
0	0	11	2	0	0	9	3	0	0	3
0	0	0	3	1	0	2	0	0	0	0
4	1	2	3	0	4	0	0	4	0	0
5	0	0	0	4	0	1	0	4	3	0
0	0	0	7	4	0	1	0	0	7	0
0	8	0	0	0	0	3	0	1	0	3
0	3	0	0	5	2	3	0	0	0	2
0	0	0	3	1	0	2	8	0	0	0
0	3	4	0	7	0	0	7	0	0	0
4	2	0	4	0	3	0	0	5	7	0

ID8

1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	10	10	0	0	0	4	5	0	0
0	4	1	0	8	0	0	0	0	0	5
8	0	0	0	0	0	6	0	0	0	0
0	0	0	0	13	0	0	0	2	0	3
0	0	0	0	4	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0
0	0	4	0	0	0	0	3	0	0	0
4	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

ID9

2	4	0	0	6	7	3	4	0	0	3	0	1
0	0	2	0	3	0	0	6	0	0	8	11	3
0	3	0	8	0	0	2	0	0	0	0	0	4
2	0	0	0	0	0	0	0	0	3	2	0	0
0	6	0	8	0	3	0	0	0	0	0	0	1
0	3	0	2	0	0	9	0	0	0	0	5	6
1	9	4	0	0	2	4	0	0	0	3	2	0
2	3	0	4	0	0	0	6	2	0	1	0	3
0	0	0	0	0	6	0	0	0	2	2	0	8
7	2	4	2	0	0	6	4	1	0	0	0	7
0	0	0	11	0	0	0	0	3	4	0	9	0
0	0	0	0	1	4	3	4	0	0	0	3	11
0	0	4	7	7	0	0	2	0	2	5	0	1

ID10

0	0	1	4	0	0	9	0	0	0	12	0	1
0	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	2	0	0	2	0	0	0
0	0	0	0	0	9	4	0	0	0	6	0	0
0	6	9	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	6	10	0	1	4
0	3	0	0	0	1	0	0	0	0	0	2	0
0	0	0	1	3	0	0	0	0	9	0	0	0
9	0	0	3	3	0	0	0	0	3	4	0	0
0	1	4	0	0	0	0	0	0	5	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	3	3	0	0	0	0	0	10
0	0	0	0	0	0	0	0	0	4	0	0	0

ID11

0	0	0	4	0	0	0	6	0	0	0	0	2	2	0
0	2	12	0	3	0	0	0	0	26	0	0	0	0	4
0	0	0	0	0	0	0	0	0	0	0	0	2	2	0
0	0	0	0	0	0	0	3	3	1	0	0	0	0	0
0	0	0	0	0	1	3	0	0	6	4	0	0	0	0
0	0	0	0	0	0	0	5	4	0	0	3	0	0	0
9	12	0	0	0	4	1	6	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	4	0	0	0	0	0	0
0	3	0	0	0	2	0	0	0	7	0	4	0	0	0
0	0	2	0	0	9	2	0	0	0	0	0	0	0	0
0	2	0	0	2	16	0	8	0	2	0	0	0	0	7
0	0	5	0	6	0	0	0	0	0	8	0	0	0	0
0	4	0	0	0	0	0	0	1	2	3	0	0	0	0

ID12

0	0	0	0	0	0	0	10	3	0	0	0	0	2	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	1	0	3	0	0	0	0	4	0	0	0	4	0	0
0	0	0	10	3	8	11	0	0	0	0	0	2	0	0
0	4	0	0	0	0	0	0	0	0	0	2	0	0	1
0	4	2	0	0	0	4	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	5	0	10	0	0	1	0	0
0	0	0	0	0	0	0	0	3	0	0	2	8	0	15
0	1	0	0	0	0	0	0	0	0	0	0	11	0	0
0	0	0	0	0	3	0	0	0	0	0	1	0	2	0
0	0	0	11	0	0	0	0	0	0	0	0	0	0	2
8	0	0	0	0	4	0	0	0	0	0	4	2	0	4
0	0	0	0	0	0	0	0	0	0	0	0	0	8	1

ID13

0	0	0	3	0	0	0	0	5	0	0	0	0	0	0	0	0
0	0	0	0	3	0	0	0	4	2	0	3	0	0	0	0	0
6	0	3	0	0	0	6	0	30	0	1	8	6	10	0	0	0
0	7	0	1	4	0	0	1	0	0	0	0	0	0	0	4	4
0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3
0	8	0	0	0	0	0	3	0	0	36	0	1	0	0	2	0
6	0	0	0	8	2	8	0	0	2	0	0	0	0	0	0	0
8	1	0	0	0	0	4	1	0	0	0	0	0	6	7	0	0
3	5	0	0	0	0	0	0	0	0	5	0	0	4	0	0	1
3	0	0	2	0	4	0	0	0	0	9	0	0	0	8	16	24
0	1	0	0	1	1	0	0	2	0	0	0	0	0	6	1	0
0	3	4	0	3	4	0	10	0	0	0	0	5	5	8	4	4
8	0	0	0	0	0	17	0	0	10	0	2	0	0	2	0	0

ID14

0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	4	0	0	0
0	6	0	0	0	0	8	0	10	0	0	0	0	2	2	3	0
0	0	0	0	0	4	0	8	3	0	0	0	0	0	0	0	0
0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0
4	0	8	1	0	0	7	0	0	0	0	0	5	3	0	0	0
0	0	3	0	1	0	0	3	0	0	3	0	3	0	8	0	0
0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	1	0
0	6	0	0	0	0	0	1	0	2	0	1	0	0	0	0	0
0	0	2	0	1	3	0	1	0	4	0	0	6	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
4	8	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	3	0	0	6	3

ID15

0	0	0	0	4	0	0	4	0	0	8	0	6	0	0	0	0	0	4
0	0	0	0	0	2	0	6	0	0	0	0	0	0	0	0	0	3	1
2	0	8	3	0	0	0	5	0	4	0	0	0	0	0	2	1	4	0
0	0	1	0	4	0	0	0	0	0	0	1	0	0	0	18	10	0	0
0	0	1	0	0	0	0	3	0	2	0	0	0	7	4	0	0	4	3
0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	3	2	0	2
0	0	0	1	0	1	0	0	0	2	2	0	0	4	0	0	10	1	0
3	0	0	0	0	0	0	4	1	0	0	0	0	0	4	0	0	1	0
2	0	2	0	0	0	0	1	0	0	4	1	0	3	0	0	0	3	3
0	0	0	0	4	0	1	1	3	0	0	0	0	0	0	0	0	0	0
0	0	0	4	0	0	4	2	4	0	0	0	0	0	0	4	0	0	0
0	0	2	0	3	22	0	0	0	0	0	2	7	0	0	0	0	0	1
0	9	0	0	6	0	0	0	0	0	0	0	0	0	5	1	4	0	8

ID16

0	0	0	0	0	0	0	3	0	0	0	2	0	4	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0
5	0	0	3	0	0	0	0	0	0	0	2	1	0	0	0	0	0	0
0	0	4	0	0	0	0	0	14	0	0	0	0	2	0	7	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	2
0	0	3	3	0	0	0	0	0	3	0	0	0	0	0	0	2	0	0
5	0	0	0	0	0	6	0	2	2	0	0	0	0	0	0	0	0	0
0	0	2	0	0	0	0	0	4	0	0	5	0	0	0	0	0	4	1
3	0	4	0	0	0	0	0	0	0	0	7	2	0	0	0	1	0	3
0	0	1	0	0	4	11	0	3	0	0	0	0	11	3	0	0	0	0
1	0	2	8	0	0	0	0	0	0	4	0	0	3	1	0	0	0	0
3	0	11	0	0	0	0	0	0	0	0	0	0	3	0	7	0	0	0
0	0	0	0	0	2	0	0	0	4	0	0	0	0	0	0	0	1	0

ID17

0	0	0	0	0	0	0	0	16	1	0	5	0	3	0	0	0	4	0
0	2	3	0	0	5	0	0	0	0	0	0	0	6	0	0	0	1	0
0	9	1	0	0	0	0	4	2	2	0	1	8	2	0	4	24	10	13
0	0	3	0	0	0	0	2	0	0	4	0	11	0	0	0	2	1	1
0	4	0	1	0	0	0	0	0	0	0	0	0	0	0	4	0	0	2
0	3	12	0	4	0	0	0	0	0	0	0	10	0	0	0	0	0	0
0	0	2	0	0	0	9	0	0	0	0	0	8	4	0	0	0	0	0
3	0	0	0	0	2	0	0	6	0	3	0	6	0	0	0	0	0	0
1	0	0	0	0	0	0	2	5	0	0	12	2	4	0	0	7	0	1
6	4	4	0	0	8	0	3	2	0	0	0	9	0	0	0	0	0	0
0	0	0	0	0	0	4	8	0	0	2	0	0	8	0	0	0	0	2
0	4	18	0	0	0	0	0	0	0	4	1	2	0	0	0	0	8	3
2	0	7	0	7	0	0	9	0	0	0	0	0	0	0	2	0	0	0
0	0	0	0	7	2	0	0	1	0	0	0	0	0	0	11	0	30	0
1	0	0	0	0	0	0	7	0	0	0	3	0	0	0	0	0	0	0

ID18

0	0	0	0	14	0	0	0	0	0	0	0	3	7	0	0	0	0	0
0	2	5	7	2	0	0	0	6	0	0	0	1	0	0	3	0	0	1
0	7	0	2	0	0	0	0	0	0	0	10	0	0	4	2	0	0	0
0	0	0	4	0	0	0	5	0	0	0	0	0	0	0	0	0	2	0
0	0	5	0	4	0	0	3	4	0	0	0	3	0	0	0	0	7	0
0	0	0	0	0	0	3	0	6	0	0	5	0	4	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	3	0
0	0	7	0	4	0	0	0	0	0	1	0	0	0	8	0	0	0	0
4	0	0	0	0	0	0	0	7	0	7	0	0	0	0	0	8	0	3
8	0	0	0	0	0	2	6	2	0	0	0	0	0	3	0	0	0	0
0	0	0	0	0	0	0	12	0	0	0	4	0	0	8	0	0	0	0
0	0	4	0	0	0	0	0	0	13	0	2	0	0	0	0	0	1	0
0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	3	0	2	0
0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	7	0
0	0	0	0	0	0	4	0	0	0	2	0	0	0	0	0	2	0	3

ID19

0	2	0	0	0	4	4	0	0	4	0	0	1	6	0	1	4	0	0
1	0	0	2	0	0	0	0	0	0	0	0	9	3	0	0	0	0	0
3	0	0	4	0	9	1	0	0	1	0	0	0	6	0	0	0	0	0
0	4	0	0	4	4	0	0	0	0	0	0	12	0	0	0	0	1	0
0	0	0	3	0	6	0	0	0	0	3	0	0	11	17	0	0	0	0
6	0	1	0	0	6	0	0	1	0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0	6	0	0	1	0	0	0	2	0
1	0	10	0	0	2	2	0	3	4	8	0	0	9	11	1	0	16	0
3	0	0	0	0	0	0	0	4	0	0	7	0	0	7	0	0	0	0
0	6	0	1	0	0	0	0	3	5	0	0	2	4	0	0	0	0	0
0	0	0	0	0	6	0	3	6	0	10	6	0	0	0	0	0	0	2
3	0	0	4	4	0	2	0	0	0	1	0	0	1	2	16	11	0	0
7	0	0	3	0	0	0	0	0	10	12	0	0	0	0	0	0	0	0
0	2	0	0	0	0	2	2	0	0	0	0	0	0	0	4	1	0	1
0	0	0	0	0	0	0	1	2	6	3	0	0	0	0	0	7	0	0
0	0	1	0	4	8	0	0	0	0	0	6	0	0	0	6	0	0	0
0	0	0	0	0	0	0	2	0	2	0	0	0	7	0	0	0	0	2

ID20

3	4	0	0	3	0	0	0	0	6	0	4	4	0	0	0	4	0	0
4	0	0	5	0	0	0	0	7	3	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	1	0
0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	1	4	0	0	0	0	0	0	0	0	2	0	0	0	0	9	0	0
0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	1	0	4	0
4	0	0	5	0	0	0	0	0	0	0	11	0	0	0	0	0	3	0
2	0	7	0	0	11	0	0	0	0	0	0	5	0	7	0	0	0	0
9	0	0	0	1	0	1	15	0	0	0	0	1	0	0	0	1	4	3
0	3	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	0	0
0	0	2	4	0	0	0	0	0	0	0	0	4	0	1	0	0	0	0
0	2	0	0	0	7	0	4	0	0	0	0	0	0	0	0	0	9	0
0	0	6	0	0	0	2	0	1	0	0	0	0	0	0	0	0	1	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	3
0	0	0	0	0	1	0	0	0	0	0	0	9	0	0	0	7	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	6	3	0	0	0	0	0	0	1	0	0

Instance maps can be defined as matrices with a maximum value of 20x20, and be initialised statically in the code:

```
{
// 5x5
{
{0,7,0,0,4},
{0,0,0,4,0},
{1,0,0,0,0},
{4,4,1,0,0},
{6,0,3,4,4},
},
{
{4,0,0,10,1},
{1,0,0,0,0},
{0,0,1,6,3},
{0,4,0,0,2},
{8,0,6,3,0},
},
// 7x7
{
{0,8,0,4,5,10,0},
{0,4,0,7,0,4,0},
{0,2,4,2,0,0,2},
{0,7,0,1,2,0,0},
{2,4,0,0,3,0,2},
{0,4,0,0,3,0,0},
{2,0,0,0,0,0,0},
},
{
{0,0,1,0,7,0,1},
{0,1,4,0,0,0,4},
{0,0,0,0,2,0,0},
{3,1,0,8,5,7,7},
{0,4,0,3,0,0,0},
{0,0,0,3,2,4,2},
{0,8,3,6,3,0,0},
},
// 9x9
{
{6,7,2,0,0,0,0,0,0},
{3,3,6,0,8,4,3,1,0},
{0,0,8,0,0,0,2,4,0},
{0,0,0,1,0,3,2,0,0},
{0,0,0,7,4,0,1,0,0},
{12,8,0,5,4,1,4,3,4},
{8,0,1,2,4,3,3,0,0},
{1,1,0,0,0,0,5,0,0},
{4,0,0,0,4,6,0,13,2},
},
{
```

```
{0,0,0,0,0,0,0,0,0},
{4,0,8,4,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0},
{0,0,0,0,3,0,0,1,0},
{0,3,0,0,0,0,0,0,0},
{0,0,0,1,1,0,0,3,0},
{0,0,2,4,0,0,0,1,0},
{0,2,0,0,8,0,4,3,10},
{0,0,3,0,0,4,0,0,0},
},
// 11x11
{
{0,0,0,0,0,3,0,0,0,0,0},
{0,0,11,2,0,0,9,3,0,0,3},
{0,0,0,3,1,0,2,0,0,0,0},
{4,1,2,3,0,4,0,0,4,0,0},
{5,0,0,0,4,0,1,0,4,3,0},
{0,0,0,7,4,0,1,0,0,7,0},
{0,8,0,0,0,0,3,0,1,0,3},
{0,3,0,0,5,2,3,0,0,0,2},
{0,0,0,3,1,0,2,8,0,0,0},
{0,3,4,0,7,0,0,7,0,0,0},
{4,2,0,4,0,3,0,0,5,7,0},
},
{
{1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0},
{0,0,10,10,0,0,0,4,5,0,0},
{0,4,1,0,8,0,0,0,0,0,5},
{8,0,0,0,0,0,6,0,0,0,0},
{0,0,0,0,13,0,0,0,2,0,3},
{0,0,0,0,4,0,0,0,0,1,0},
{0,0,0,0,0,0,0,0,0,0,0},
{0,0,4,0,0,0,0,3,0,0,0},
{4,1,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0},
},
// 13x13
{
{2,4,0,0,6,7,3,4,0,0,3,0,1},
{0,0,2,0,3,0,0,6,0,0,8,11,3},
{0,3,0,8,0,0,2,0,0,0,0,0,4},
{2,0,0,0,0,0,0,0,0,3,2,0,0},
{0,6,0,8,0,3,0,0,0,0,0,0,1},
{0,3,0,2,0,0,9,0,0,0,0,5,6},
{1,9,4,0,0,2,4,0,0,0,3,2,0},
{2,3,0,4,0,0,0,6,2,0,1,0,3},
{0,0,0,0,0,6,0,0,0,2,2,0,8},
{7,2,4,2,0,0,6,4,1,0,0,0,7},
{0,0,0,11,0,0,0,0,3,4,0,9,0},
{0,0,0,0,1,4,3,4,0,0,0,3,11},
{0,0,4,7,7,0,0,2,0,2,5,0,1},
},
{
{0,0,1,4,0,0,9,0,0,0,12,0,1},
{0,0,0,0,0,0,0,0,0,1,0,0,0},
{1,0,0,0,0,0,2,0,0,2,0,0,0},
{0,0,0,0,0,9,4,0,0,0,6,0,0},
{0,6,9,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,1,6,10,0,1,4},
{0,3,0,0,0,1,0,0,0,0,0,2,0},
{0,0,0,1,3,0,0,0,0,9,0,0,0},
{9,0,0,3,3,0,0,0,0,3,4,0,0},
{0,1,4,0,0,0,0,0,0,5,0,1,0},
{0,0,0,0,0,0,0,0,0,0,0,0,0},
{2,0,0,0,0,3,3,0,0,0,0,0,10},
{0,0,0,0,0,0,0,0,0,4,0,0,0},
},
// 15x13
{
{0, 0, 0, 4, 0, 0, 0, 6, 0, 0, 0, 0, 2, 2, 0},
{ 0,2,12,0,3,0,0,0,0,26,0,0,0,0,4 },
{ 0,0,0,0,0,0,0,0,0,0,0,0,2,2,0 },
{ 0,0,0,0,0,0,0,3,3,1,0,0,0,0,0 },
{ 0,0,0,0,0,1,3,0,0,6,4,0,0,0,0 },
{ 0,0,0,0,0,0,0,5,4,0,0,3,0,0,0 },
{ 9,12,0,0,0,4,1,6,0,0,0,0,0,0,0 },
{ 0,0,0,0,0,0,0,0,4,0,0,0,0,0,0 },
{ 0,3,0,0,0,2,0,0,0,7,0,4,0,0,0 },
{ 0,0,2,0,0,9,2,0,0,0,0,0,0,0,0 },
{ 0,2,0,0,2,16,0,8,0,2,0,0,0,0,7 },
{ 0,0,5,0,6,0,0,0,0,0,8,0,0,0,0 },
{ 0,4,0,0,0,0,0,0,1,2,3,0,0,0,0 },
},
{
{0,0,0,0,0,0,0,10,3,0,0,0,0,2,0},
{0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0},
```

```
{0,1,0,3,0,0,0,0,4,0,0,0,4,0,0},
{0,0,0,10,3,8,11,0,0,0,0,0,2,0,0},
{0,4,0,0,0,0,0,0,0,0,0,2,0,0,1},
{0,4,2,0,0,0,4,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,5,0,10,0,0,1,0,0},
{0,0,0,0,0,0,0,0,0,3,0,0,2,8,0,15},
{0,1,0,0,0,0,0,0,0,0,0,0,11,0,0},
{0,0,0,0,0,3,0,0,0,0,0,1,0,2,0},
{0,0,0,11,0,0,0,0,0,0,0,0,0,0,2},
{8,0,0,0,0,4,0,0,0,0,0,4,2,0,4},
{0,0,0,0,0,0,0,0,0,0,0,0,0,8,1},
},
// 17x13
{
{0,0,0,3,0,0,0,0,5,0,0,0,0,0,0,0,0},
{0,0,0,0,3,0,0,0,4,2,0,3,0,0,0,0,0},
{6,0,3,0,0,0,6,0,30,0,1,8,6,10,0,0,0,0},
{0,7,0,1,4,0,0,1,0,0,0,0,0,0,0,4,4},
{0,2,0,0,1,0,0,0,0,0,0,0,0,0,0,0,3},
{0,8,0,0,0,0,0,3,0,0,36,0,1,0,0,2,0},
{6,0,0,0,8,2,8,0,0,2,0,0,0,0,0,0,0},
{8,1,0,0,0,0,4,1,0,0,0,0,0,0,6,7,0,0},
{3,5,0,0,0,0,0,0,0,0,5,0,0,4,0,0,1},
{3,0,0,2,0,4,0,0,0,0,9,0,0,0,8,16,24},
{0,1,0,0,1,1,0,0,2,0,0,0,0,0,6,1,0},
{0,3,4,0,3,4,0,10,0,0,0,0,0,5,5,8,4,4},
{8,0,0,0,0,0,17,0,0,10,0,2,0,0,2,0,0},
},
{
{0,0,0,10,0,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,1,4,0,0,0},
{0,6,0,0,0,0,8,0,10,0,0,0,0,2,2,3,0},
{0,0,0,0,0,4,0,8,3,0,0,0,0,0,0,0,0},
{0,0,0,0,4,0,0,0,0,0,0,0,0,0,0,0,0},
{4,0,8,1,0,0,7,0,0,0,0,0,5,3,0,0,0},
{0,0,3,0,1,0,0,3,0,0,3,0,3,0,8,0,0},
{0,0,0,0,11,0,0,0,0,0,0,0,0,0,1,0},
{0,6,0,0,0,0,0,1,0,2,0,1,0,0,0,0,0},
{0,0,2,0,1,3,0,1,0,4,0,0,6,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0},
{4,8,0,0,0,0,0,0,0,2,0,0,0,0,0,0},
{4,0,0,0,0,0,0,0,0,0,0,3,0,0,6,3},
},
// 19x13
{
{0,0,0,0,4,0,0,4,0,0,8,0,6,0,0,0,0,0,4},
{0,0,0,0,0,2,0,6,0,0,0,0,0,0,0,0,0,3,1},
{2,0,8,3,0,0,0,5,0,4,0,0,0,0,0,2,1,4,0},
{0,0,1,0,4,0,0,0,0,0,0,1,0,0,0,18,10,0,0},
{0,0,1,0,0,0,0,3,0,2,0,0,0,7,4,0,0,4,3},
{0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,3,2,0,2},
{0,0,0,1,0,1,0,0,0,2,2,0,0,4,0,0,10,1,0},
{3,0,0,0,0,0,0,4,1,0,0,0,0,0,4,0,0,1,0},
{2,0,2,0,0,0,0,1,0,0,4,1,0,3,0,0,0,3,3},
{0,0,0,0,4,0,1,1,3,0,0,0,0,0,0,0,0,0,0},
{0,0,0,4,0,0,4,2,4,0,0,0,0,0,0,4,0,0,0},
{0,0,2,0,3,22,0,0,0,0,0,2,7,0,0,0,0,0,1},
{0,9,0,0,6,0,0,0,0,0,0,0,0,5,1,4,0,8},
},
{
{0,0,0,0,0,0,0,3,0,0,0,2,0,4,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0},
{5,0,0,3,0,0,0,0,0,0,0,2,1,0,0,0,0,0,0},
{0,0,4,0,0,0,0,0,0,14,0,0,0,0,2,0,7,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,0,0,0,2},
{0,0,3,3,0,0,0,0,0,3,0,0,0,0,0,0,2,0,0},
{5,0,0,0,0,0,6,0,2,2,0,0,0,0,0,0,0,0,0},
{0,0,2,0,0,0,0,0,0,4,0,0,5,0,0,0,0,0,4,1},
{3,0,4,0,0,0,0,0,0,0,0,7,2,0,0,0,1,0,3},
{0,0,1,0,0,4,11,0,3,0,0,0,0,11,3,0,0,0,0},
{1,0,2,8,0,0,0,0,0,0,4,0,0,3,1,0,0,0,0},
{3,0,11,0,0,0,0,0,0,0,0,0,0,0,3,0,7,0,0,0},
{0,0,0,0,0,2,0,0,0,4,0,0,0,0,0,0,0,1,0},
},
// 19x15
{
{0,0,0,0,0,0,0,0,16,1,0,5,0,3,0,0,0,4,0},
{0,2,3,0,0,5,0,0,0,0,0,0,0,6,0,0,0,1,0},
{0,9,1,0,0,0,0,4,2,2,0,1,8,2,0,4,24,10,13},
{0,0,3,0,0,0,0,2,0,0,4,0,11,0,0,0,2,1,1},
{0,4,0,1,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,2},
{0,3,12,0,4,0,0,0,0,0,0,0,10,0,0,0,0,0,0},
{0,0,2,0,0,0,9,0,0,0,0,0,8,4,0,0,0,0,0},
{3,0,0,0,0,2,0,0,6,0,3,0,6,0,0,0,0,0,0},
{1,0,0,0,0,0,0,0,2,5,0,0,12,2,4,0,0,7,0,1},
{6,4,4,0,0,8,0,3,2,0,0,0,9,0,0,0,0,0,0},
{0,0,0,0,0,0,4,8,0,0,2,0,0,8,0,0,0,0,2},
```

```
{0,4,18,0,0,0,0,0,0,0,4,1,2,0,0,0,0,8,3},
{2,0,7,0,7,0,0,9,0,0,0,0,0,0,0,2,0,0,0},
{0,0,0,0,7,2,0,0,1,0,0,0,0,0,0,11,0,30,0},
{1,0,0,0,0,0,0,7,0,0,0,3,0,0,0,0,0,0,0},
},
{
{0,0,0,0,14,0,0,0,0,0,0,0,3,7,0,0,0,0,0},
{0,2,5,7,2,0,0,0,6,0,0,0,1,0,0,3,0,0,1},
{0,7,0,2,0,0,0,0,0,0,0,10,0,0,4,2,0,0,0},
{0,0,0,4,0,0,0,5,0,0,0,0,0,0,0,0,0,2,0},
{0,0,5,0,4,0,0,3,4,0,0,0,3,0,0,0,0,7,0},
{0,0,0,0,0,0,3,0,6,0,0,5,0,4,0,0,0,0,0},
{2,0,0,0,0,0,0,0,0,0,0,3,0,0,0,0,0,0,3,0},
{0,0,7,0,4,0,0,0,0,0,1,0,0,0,8,0,0,0,0},
{4,0,0,0,0,0,0,0,7,0,7,0,0,0,0,0,8,0,3},
{8,0,0,0,0,0,2,6,2,0,0,0,0,0,3,0,0,0,0},
{0,0,0,0,0,0,0,0,12,0,0,0,4,0,0,8,0,0,0,0},
{0,0,4,0,0,0,0,0,0,13,0,2,0,0,0,0,0,1,0},
{0,0,0,0,0,0,3,0,0,0,0,0,0,0,0,3,0,2,0},
{0,0,0,0,0,0,0,0,5,0,0,0,0,0,0,0,0,7,0},
{0,0,0,0,0,0,4,0,0,0,2,0,0,0,0,0,2,0,3},
},
// 19x17
{
{0,2,0,0,0,4,4,0,0,4,0,0,1,6,0,1,4,0,0},
{1,0,0,2,0,0,0,0,0,0,0,0,0,9,3,0,0,0,0,0},
{3,0,0,4,0,9,1,0,0,1,0,0,0,6,0,0,0,0,0},
{0,4,0,0,4,4,0,0,0,0,0,0,12,0,0,0,0,1,0},
{0,0,0,3,0,6,0,0,0,0,3,0,0,11,17,0,0,0,0},
{6,0,1,0,0,6,0,0,1,0,0,0,0,0,0,0,0,0,0},
{0,1,0,1,0,0,0,0,0,0,6,0,0,1,0,0,0,2,0},
{1,0,10,0,0,2,2,0,3,4,8,0,0,9,11,1,0,16,0},
{3,0,0,0,0,0,0,0,4,0,0,7,0,0,7,0,0,0,0},
{0,6,0,1,0,0,0,0,0,3,5,0,0,2,4,0,0,0,0,0},
{0,0,0,0,0,0,6,0,3,6,0,10,6,0,0,0,0,0,0,2},
{3,0,0,4,4,0,2,0,0,0,1,0,0,1,2,16,11,0,0},
{7,0,0,3,0,0,0,0,0,10,12,0,0,0,0,0,0,0,0},
{0,2,0,0,0,0,0,2,2,0,0,0,0,0,0,0,0,4,1,0,1},
{0,0,0,0,0,0,0,0,1,2,6,3,0,0,0,0,0,0,7,0,0},
{0,0,1,0,4,8,0,0,0,0,0,6,0,0,0,6,0,0,0},
{0,0,0,0,0,0,0,2,0,2,0,0,0,7,0,0,0,0,2},
},
{
{3,4,0,0,3,0,0,0,0,6,0,4,4,0,0,0,4,0,0},
{4,0,0,5,0,0,0,0,7,3,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,4,0,0,0,0,0,0,0,0,0,1,0},
{0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0},
{0,1,4,0,0,0,0,0,0,0,0,2,0,0,0,0,9,0,0},
{0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,1,0,4,0},
{4,0,0,5,0,0,0,0,0,0,0,11,0,0,0,0,0,3,0},
{2,0,7,0,0,11,0,0,0,0,0,0,5,0,7,0,0,0,0},
{9,0,0,0,1,0,1,15,0,0,0,0,1,0,0,0,1,4,3},
{0,3,0,0,0,0,0,2,0,0,0,2,0,0,0,0,0,0,0},
{0,0,2,4,0,0,0,0,0,0,0,0,4,0,1,0,0,0,0},
{0,2,0,0,0,7,0,4,0,0,0,0,0,0,0,0,0,9,0},
{0,0,6,0,0,0,2,0,1,0,0,0,0,0,0,0,0,1,0},
{3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,0,3},
{0,0,0,0,0,1,0,0,0,0,0,0,9,0,0,0,7,0,0},
{0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0},
{0,0,0,0,0,0,0,0,0,6,3,0,0,0,0,0,0,1,0,0},
},
};
```