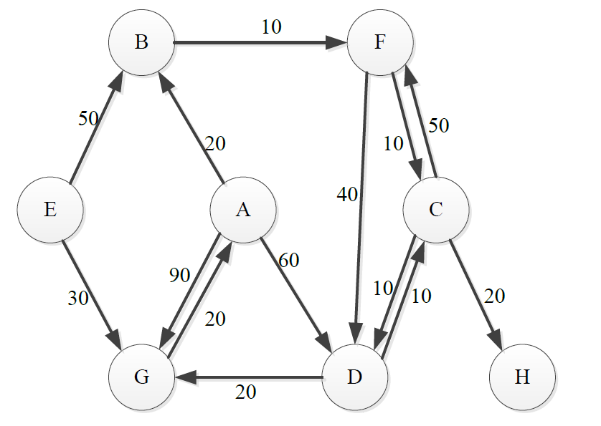
Question 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** |
| **A** |  |  |  |  |  |
| **B** |  |  |  |  |  |
| **C** |  |  |  |  |  |
| **D** |  |  |  |  |  |
| **E** |  |  |  |  |  |

Question 2.



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** |
| **A** |  |  |  |  |  |  |  |  |
| **B** |  |  |  |  |  |  |  |  |
| **C** |  |  |  |  |  |  |  |  |
| **D** |  |  |  |  |  |  |  |  |
| **E** |  |  |  |  |  |  |  |  |
| **F** |  |  |  |  |  |  |  |  |
| **G** |  |  |  |  |  |  |  |  |
| **H** |  |  |  |  |  |  |  |  |

Dijkstra’s algorithm:

* Go to the nearest vertex that has not been explored -> Red
* Try to update the path distances, by comparing the current path distance against the path transiting through the selected vertex.

Let say going from A to a target through B:

If update -> **Bold**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Step |  |  | A | B | C | D | E | F | G | H |
| 1 | - |  |  |  |  |  |  |  |  |  |
| 2 | B |  |  |  |  |  |  |  |  |  |
| 3 | F |  |  |  |  |  |  |  |  |  |
| 4 | C |  |  |  |  |  |  |  |  |  |
| 5 | D |  |  |  |  |  |  |  |  |  |
| 6 | H |  |  |  |  |  |  |  |  |  |
| 7 | G |  |  |  |  |  |  |  |  |  |

Backtrack to find the best path:

* Find the last time a node was updated
* Append the node to at that step:

: Last updated in step 1 : Last updated in step 3

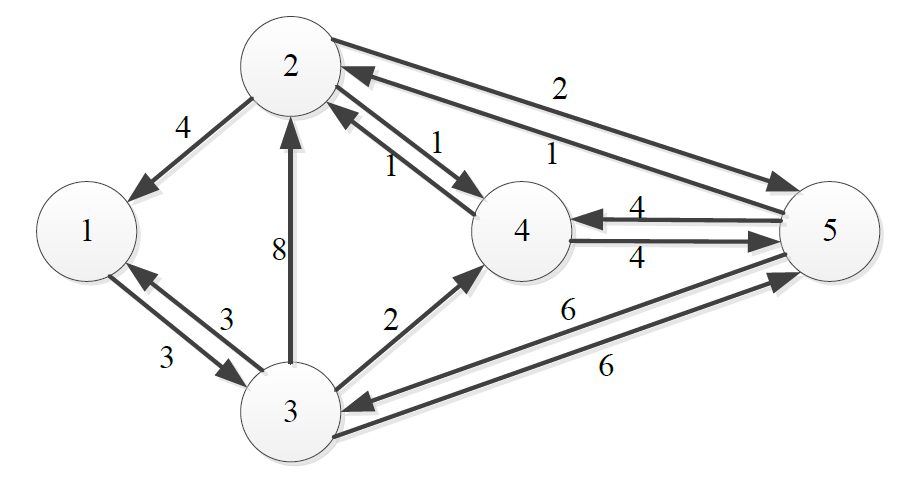
: Last updated in step 4

: No updates

: Last updated in step 2

: Last updated in step 6

: Last updated in step 5

Question 3:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

FloydAlgorithm:

* Go through each node, and ask
  + For any node i, and for any node j, is it better to go from i to j with k in between?

E.g.: Is where is the current best path going from to .

For

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

For

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

For

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

For

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

For

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

The shortest distances going from each vertex to each other vertex is defined by the distance matrix:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |

T1, 100

T2, 30, T1

T3, 50, T2, T5

T4, 90, T1, T7

T5, 70, T2, T4

T6, 55, T5

T7, 50

Identify all vertices for the matrix’s header.

Convert from a header to an index, for example:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | T3 | T1 | T7 | T6 | T2 | T5 | T4 |
| T3 |  |  |  |  |  |  |  |
| T1 |  |  |  |  |  |  |  |
| T7 |  |  |  |  |  |  |  |
| T6 |  |  |  |  |  |  |  |
| T2 |  |  |  |  |  |  |  |
| T5 |  |  |  |  |  |  |  |
| T4 |  |  |  |  |  |  |  |