**04. Constrained Shortest Path**

**Condition:**

Given a weighted directed graph with N vertices and M edges. Each vertex has a cost to visit, and each edge has a weight. Your goal is to find the shortest path from a given starting vertex to a given ending vertex, minimizing the sum of the edge weights and the costs of the vertices along the path. You must not pass through vertices with a cost greater than a given value K.

**Input:**

1. The first line contains three integers: the number of vertices N (1 ≤ N ≤ 1000), the number of edges M (1 ≤ M ≤ 5000), and the maximum allowable cost K (1 ≤ K ≤ 1000).
2. The second line contains N integers: the prices of the vertices (the cost of visiting each vertex). The prices are in the interval [-1000, 1000].
3. Next are M rows, each containing three integers u, v, and w (1 ≤ u, v ≤ N, -1000 ≤ w ≤ 1000) - an edge from vertex u to vertex v with weight w.
4. The last line contains two integers: the starting vertex S and the ending vertex T (1 ≤ S, T ≤ N).

**Output:**

* If there exists a path from S to T, derive the minimum sum of the edge weights and vertex prices along that path.
* If no such path exists, output -1.

**Examples:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5 7 10  3 2 4 5 1  1 2 2  1 3 4  2 3 1  2 4 7  3 4 3  3 5 1  4 5 2  1 5 | 9 |