3.4 Dijkstra's Algorithm

Let G be a weighted graph with vertices $v_1, v_2, v_3, \ldots, v_n$. If G contains any self-loop, discard it. If G has parallel edges between two vertices, discard all except the one with the least weight.

Let w_{ij} represent the weight of the edge $e_{ij} = (v_i, v_j) \geq 0$. We define:

$$w_{ij} = \begin{cases} 0 & \text{if } i = j, \\ \infty & \text{if there is no edge connecting } v_i \text{ and } v_j. \end{cases}$$

Steps of Dijkstra's Algorithm

Input: A connected weighted graph G and a source vertex v_s.
Output: Shortest path from v_s to all other vertices.

Step 1: **Initialize Distances:** Set the distance of the source vertex v_s to 0 and all other vertices to ∞ (infinity).

Step 2: **Select the Vertex with the Smallest Distance:** Choose the unvisited vertex with the smallest current distance as the current vertex.

Step 3: **Update Neighboring Vertices:** For each neighbor v_t of the current vertex, calculate the tentative distance from v_i to v_t as the sum of the current vertex's distance and the edge weight between them. If this tentative distance is less than the current known distance of v_t, update v_t's distance.

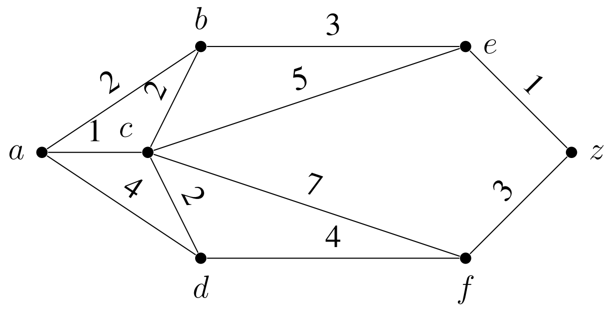
Step 4: **Mark as Visited:** Mark the current vertex as visited, as it has been processed.

Step 5: **Repeat Until All vertices Are Visited:** Go back to Step 2 if there are unvisited vertices remaining.

Step 6: **End:** Once all vertices have been visited, the shortest path from v_s to all vertices is obtained.

3.4.1 Illustrative Examples

Example 3.2 Apply shortest path algorithm to determine a shortest path between a to z in the following graph.



Here, a is the starting vertex and z is the terminating vertex.
The algorithm is displayed in the following table.

a	b	c	d	e	f	z	**Permanent vertex**
$\boxed{0}$ ✓	∞	∞	∞	∞	∞	∞	a
$\boxed{0}$	min(∞,0+2) $=2$	min(∞,0+1) $=\boxed{1}$ ✓	min(∞,0+4) $=4$	min(∞,0+∞) $=\infty$	min(∞,0+∞) $=\infty$	min(∞,0+∞) $=\infty$	c
$\boxed{0}$	min(2,1+2) $=\boxed{2}$ ✓	$\boxed{1}$	min(4,1+2) $=3$	min(∞,1+5) $=6$	min(∞,1+7) $=8$	min(∞,1+∞) $=\infty$	b
$\boxed{0}$	$\boxed{2}$	$\boxed{1}$	min(3,2+∞) $=\boxed{3}$ ✓	min(6,2+3) $=5$	min(8,2+∞) $=8$	min(∞,2+∞) $=\infty$	d
$\boxed{0}$	$\boxed{2}$	$\boxed{1}$	$\boxed{3}$	min(5,3+∞) $=\boxed{5}$ ✓	min(8,3+4) $=7$	min(∞,3+∞) $=\infty$	e
$\boxed{0}$	$\boxed{2}$	$\boxed{1}$	$\boxed{3}$	$\boxed{5}$	min(7,5+∞) $=7$	min(∞,5+1) $=\boxed{6}$ ✓	z

The permanent label of the destination vertex z is 6, so the shortest distance from a to z is 6.

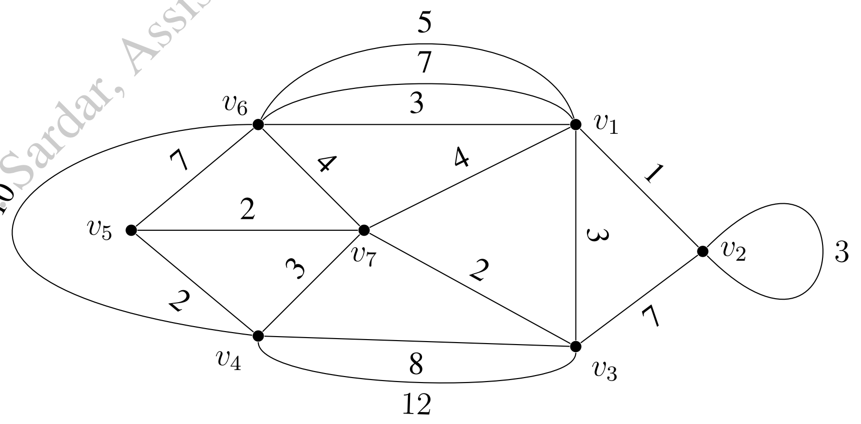
To find the shortest path, we use backtracking as follows:

Starting from the permanent label of z (i.e., $\boxed{6}$), we trace back along the temporary labels of z until we reach a change.

So we reach the temporary label ∞ (at 5th row,). In that row the newly permanent labeled vertex is e. Again we go back along the previously assigned temporary label of e, until we get a change. We reach the temporary label 6 (at 3rd row). In that row the recent permanent labeled vertex is b .Similarly tracking back we reach the vertex a.

Thus, the shortest path is $a - b - e - z$.

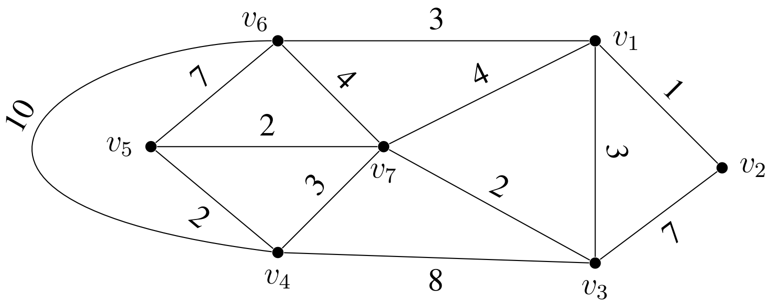
Example 3.3 By "Dijkstra's procedure," find the shortest path and the length of the shortest path from the vertex v_2 to v_5 in the following graph:



The given graph has one loop (at v_2) and 3 parallel edges connecting v_1 and v_6, as well as 2 parallel edges connecting v_3 and v_5. We discard:

- the loop at v_2,
- the two parallel edges with weights 5 and 7 between v_1 and v_6 (since the smallest weight among them is 3),
- the edge (v_3, v_4) with weight 12.

The resulting simplified graph becomes:



Here, v_2 is the starting vertex and v_5 is the terminating vertex.
The algorithm is displayed in the following table.

v_1	v_2	v_3	v_4	v_5	v_6	v_7	Permanent vertex
∞	$\boxed{0}$ ✓	∞	∞	∞	∞	∞	v_2
min(∞,0+1) $= \boxed{1}$ ✓	$\boxed{0}$	min(∞,0+7) $= 7$	min(∞,0+∞) $= \infty$	min(∞,0+∞) $= \infty$	min(∞,0+∞) $= \infty$	min(∞,0+∞) $= \infty$	v_1
$\boxed{1}$	$\boxed{0}$	min(∞,1+3) $= 4$	min(∞,1+∞) $= \infty$	min(∞,1+∞) $= \infty$	min(∞,1+3) $= \boxed{4}$ ✓	min(∞,1+4) $= 5$	v_6
$\boxed{1}$	$\boxed{0}$	min(4,4+∞) $= \boxed{4}$ ✓	min(∞,4+10) $= 14$	min(∞,4+7) $= 11$	$\boxed{4}$	min(5,4+4) $= 5$	v_3
$\boxed{1}$	$\boxed{0}$	$\boxed{4}$	min(14,4+8) $= 12$	min(11,4+∞) $= 11$	$\boxed{4}$	min(5,4+2) $= \boxed{5}$ ✓	v_7
$\boxed{1}$	$\boxed{0}$	$\boxed{4}$	min(12,5+3) $= 8$	min(11,5+2) $= \boxed{7}$ ✓	$\boxed{4}$	$\boxed{5}$	v_5

The permanent label of the destination vertex v_5 is 7, so the shortest distance from v_2 to u_5 is 7.

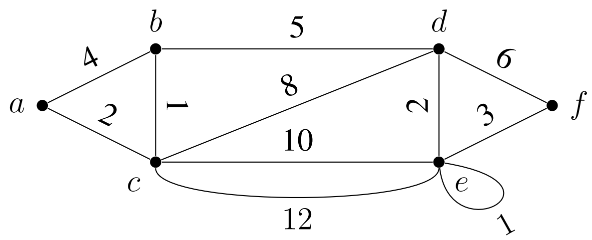
To find the shortest path, we use backtracking as follows:

Starting from the permanent label of v_5 (i.e., [7]), we trace back along the temporary labels of u_5 until we reach a change.

So we reach the temporary label 11 (at 5th row,). In that row the newly permanent labeled vertex is v_7. Again we go back along the previously assigned temporary label of v_7, until we get a change. We reach the temporary label ∞ (at 2nd row). In that row the recent permanent labeled vertex is v_1 .Similarly tracking back we reach the vertex v_2.

Thus, the shortest path is $v_2 - v_1 - v_7 - v_5$.

✍ **Exercise 3.2** Using Dijkstra's Algorithm find the length of the shortest path of the following graph from the vertex a to e:



Answer: a-c-b-d-e-f, SD=13