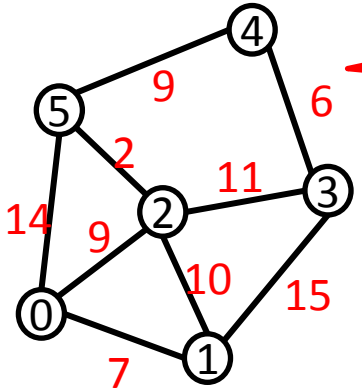
Quiz 9

(1) Make a program of Dijkstra's algorithm without built-in functions (dijkstra_path & dijkstra_path_length).

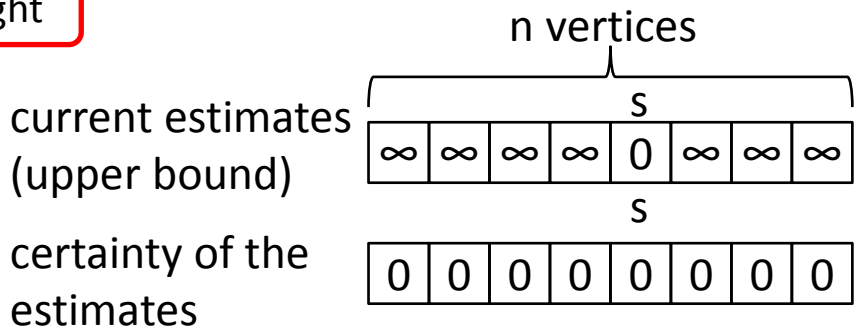
(2) Show all the statuses of current estimates and their certainties while Dijkstra's algorithm is performed from vertex 0.

(3) Start from vertex 1 and show all the statuses & their certainties.

(4) Explain the reasons why Dijkstra's algorithm does not work for negative weight edges.



edge weight



- Submit from Tokyo Tech OCW-i
- Deadline: ??:??(Japan Standard Time) on Jan. 16(Wed)
- Files should be MS Word, PDF or Zipped Jupyter notebook.

```python
import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import sys
import functools
import operator

G = nx.Graph()
G.add_nodes_from(range(0, 5))
G.add_weighted_edges_from([(0, 1, 7), (0, 2, 9), (0, 5, 14), (1, 2, 10), (1, 3, 15), (2, 3, 11), (2, 5, 2), (3, 4, 6), (4, 5, 9)])

plt.figure(figsize=(5, 5))
pos = nx.spring_layout(G)
nx.draw_networkx_edges(G, pos)
nx.draw_networkx_nodes(G, pos)
nx.draw_networkx_edge_labels(G, pos, font_size=16, edge_labels={(u, v): d["weight"] for u, v, d in G.edges(data=True)})
nx.draw_networkx_labels(G, pos)
plt.axis('off')
plt.show()

dist_estimate = [sys.maxsize] * nx.number_of_nodes(G)
dist_certainty = [0] * nx.number_of_nodes(G)
dist_estimate[0] = 0

print(dist_estimate)
print(dist_certainty)

while functools.reduce(operator.mul, dist_certainty) == 0 :
```

```
[0, 9223372036854775807, 9223372036854775807, 9223372036854775807, 9223372036854775807, 9223372036854775807]
[0, 0, 0, 0, 0, 0]
[0, 9223372036854775807, 9223372036854775807, 9223372036854775807, 9223372036854775807, 9223372036854775807]
[0, 0, 0, 0, 0, 0]
[0, 7, 9, 9223372036854775807, 9223372036854775807, 14]
[1, 0, 0, 0, 0, 0]
[0, 7, 9, 22, 9223372036854775807, 14]
[1, 1, 0, 0, 0, 0]
[0, 7, 9, 20, 9223372036854775807, 11]
[1, 1, 1, 0, 0, 0]
[0, 7, 9, 20, 20, 11]
[1, 1, 1, 0, 0, 1]
[0, 7, 9, 20, 20, 11]
[1, 1, 1, 0, 1, 1]
[0, 7, 9, 20, 20, 11]
[1, 1, 1, 1, 1, 1]
```