

```

# 計算された一筆書きの経路 (path は directedE 関数のグローバル変数として利用可能)

print(f"一筆書きの経路 (辺の順序) : {path[1:]}") # dummy index 0 を除く


# 経路をステップバイステップで可視化

for i, edge_idx in enumerate(path[1:]):

    plt.figure(figsize=(10, 7))

    # ノードを描画

    nx.draw_networkx_nodes(G, pos, node_color='skyblue', node_size=700)

    nx.draw_networkx_labels(G, pos, font_size=10, font_color='black')

    # すべての辺を描画 (初期状態は灰色)

    nx.draw_networkx_edges(G, pos, edgelist=all_edges, edge_color='gray', arrows=True, arrowsize=20,
width=1.5)

    # 辺のラベルを描画

    nx.draw_networkx_edge_labels(G, pos, edge_labels=edge_labels, font_color='red')

    # 現在の経路の辺を強調表示

    highlight_edge_u = start[edge_idx]

    highlight_edge_v = end[edge_idx]

    nx.draw_networkx_edges(G, pos, edgelist=[(highlight_edge_u, highlight_edge_v)], edge_color='red',
arrows=True, arrowsize=30, width=3.0)

    plt.title(f'Step {i+1}: Traversing Edge {edge_idx} ({highlight_edge_u} -> {highlight_edge_v})')

```

```
plt.axis('off')
```

```
plt.show()
```

<結果> :

一筆書きの経路 (辺の順番): [1, 2, 3, 7, 8, 5, 6, 4]

```
# 計算された一筆書きの経路 (path は directedE 関数のグローバル変数として利用可能)
```

```
print(f"一筆書きの経路 (辺の順序) : {path[1:]}") # dummy index 0 を除く
```

```
# 経路をステップバイステップで可視化
```

```
for i, edge_idx in enumerate(path[1:]):
```

```
    plt.figure(figsize=(10, 7))
```

```
    # ノードを描画
```

```
    nx.draw_networkx_nodes(G, pos, node_color='skyblue', node_size=700)
```

```
    nx.draw_networkx_labels(G, pos, font_size=10, font_color='black')
```

```
    # すべての辺を描画 (初期状態は灰色)
```

```
    nx.draw_networkx_edges(G, pos, edgelist=all_edges, edge_color='gray', arrows=True, arrowsize=20,
width=1.5)
```

```
    # 辺のラベルを描画
```

```
    nx.draw_networkx_edge_labels(G, pos, edge_labels=edge_labels, font_color='red')
```

```
    # 現在の経路の辺を強調表示
```

```
    highlight_edge_u = start[edge_idx]
```

```
highlight_edge_v = end[edge_idx]
```

```
nx.draw_networkx_edges(G, pos, edgelist=[(highlight_edge_u, highlight_edge_v)], edge_color='red',  
arrows=True, arrowsize=30, width=3.0)
```

```
plt.title(f'Step {i+1}: Traversing Edge {edge_idx} ({highlight_edge_u} -> {highlight_edge_v})')
```

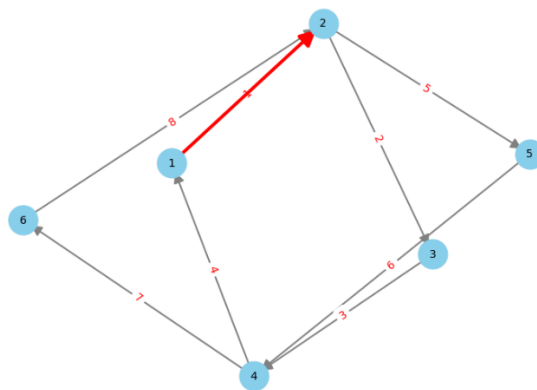
```
plt.axis('off')
```

```
plt.show()
```

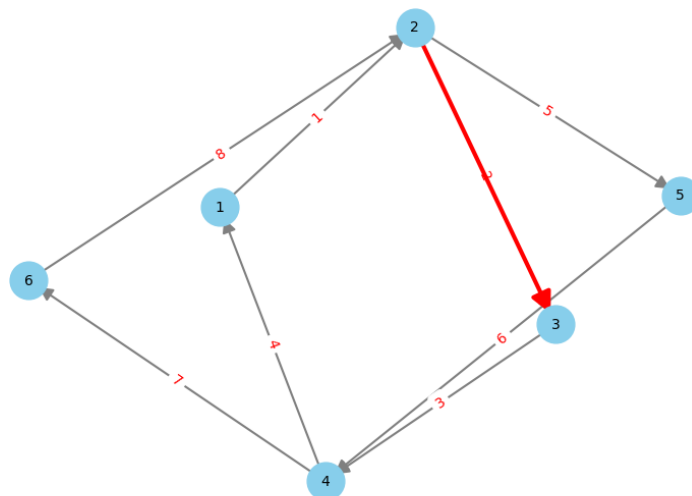
<結果> :

一筆書きの経路（辺の順序） : [1, 2, 3, 7, 8, 5, 6, 4]

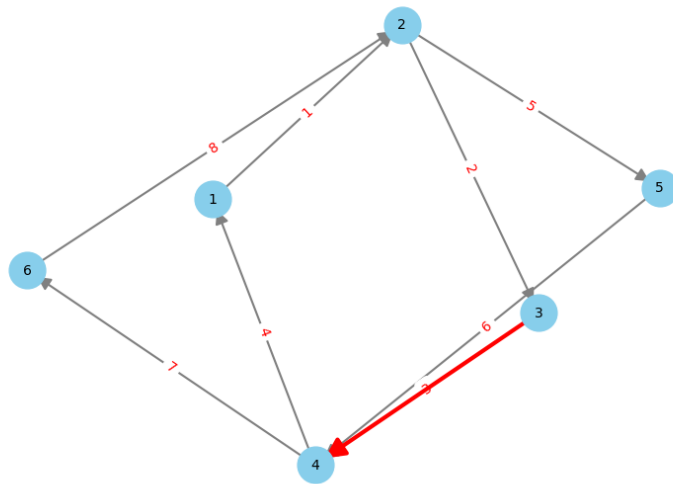
Step 1: Traversing Edge 1 (1 -> 2)



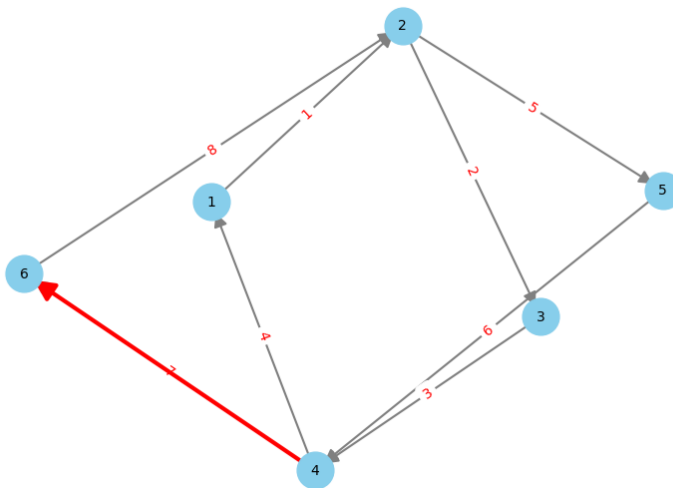
Step 2: Traversing Edge 2 (2 -> 3)



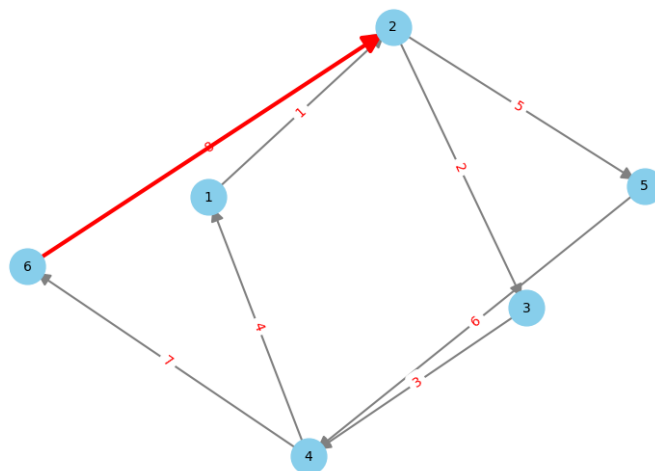
Step 3: Traversing Edge 3 (3 -> 4)



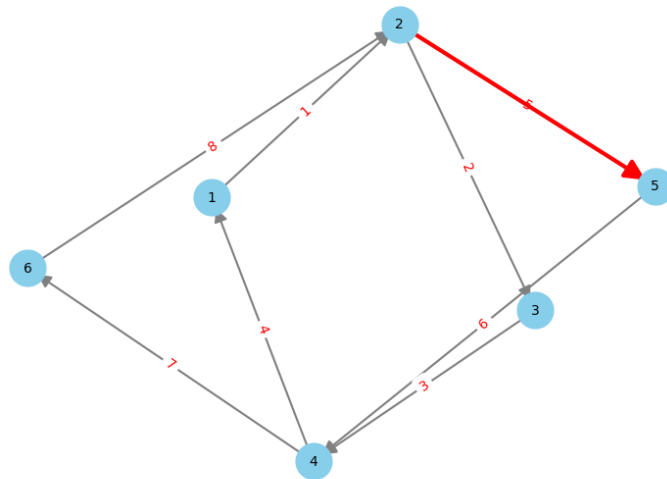
Step 4: Traversing Edge 7 (4 -> 6)



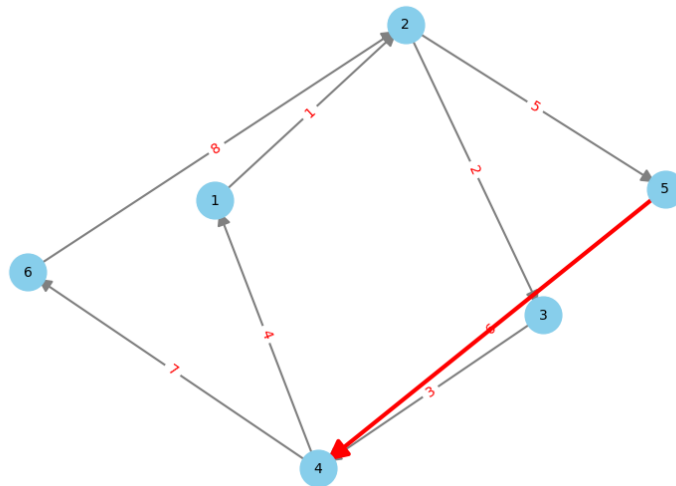
Step 5: Traversing Edge 8 (6 -> 2)



Step 6: Traversing Edge 5 (2 -> 5)



Step 7: Traversing Edge 6 (5 -> 4)



Step 8: Traversing Edge 4 (4 -> 1)

