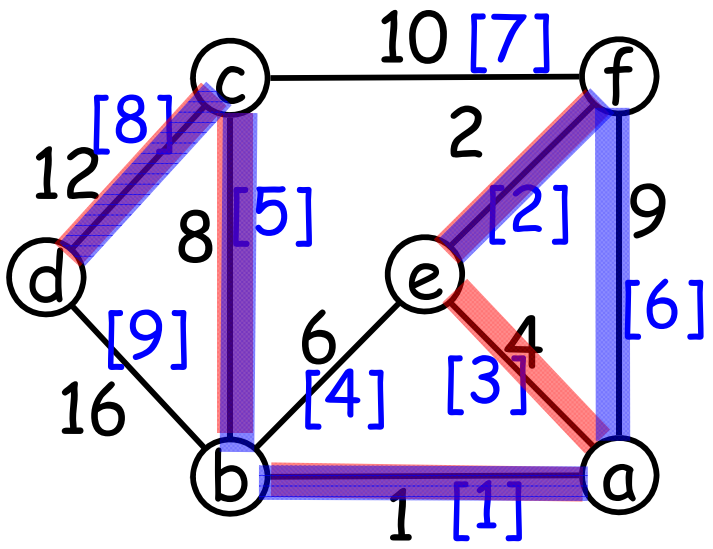
Problem: Minimum Spanning Trees

$G = (V, E)$



Cost= 1+2+4+8+12
= 27

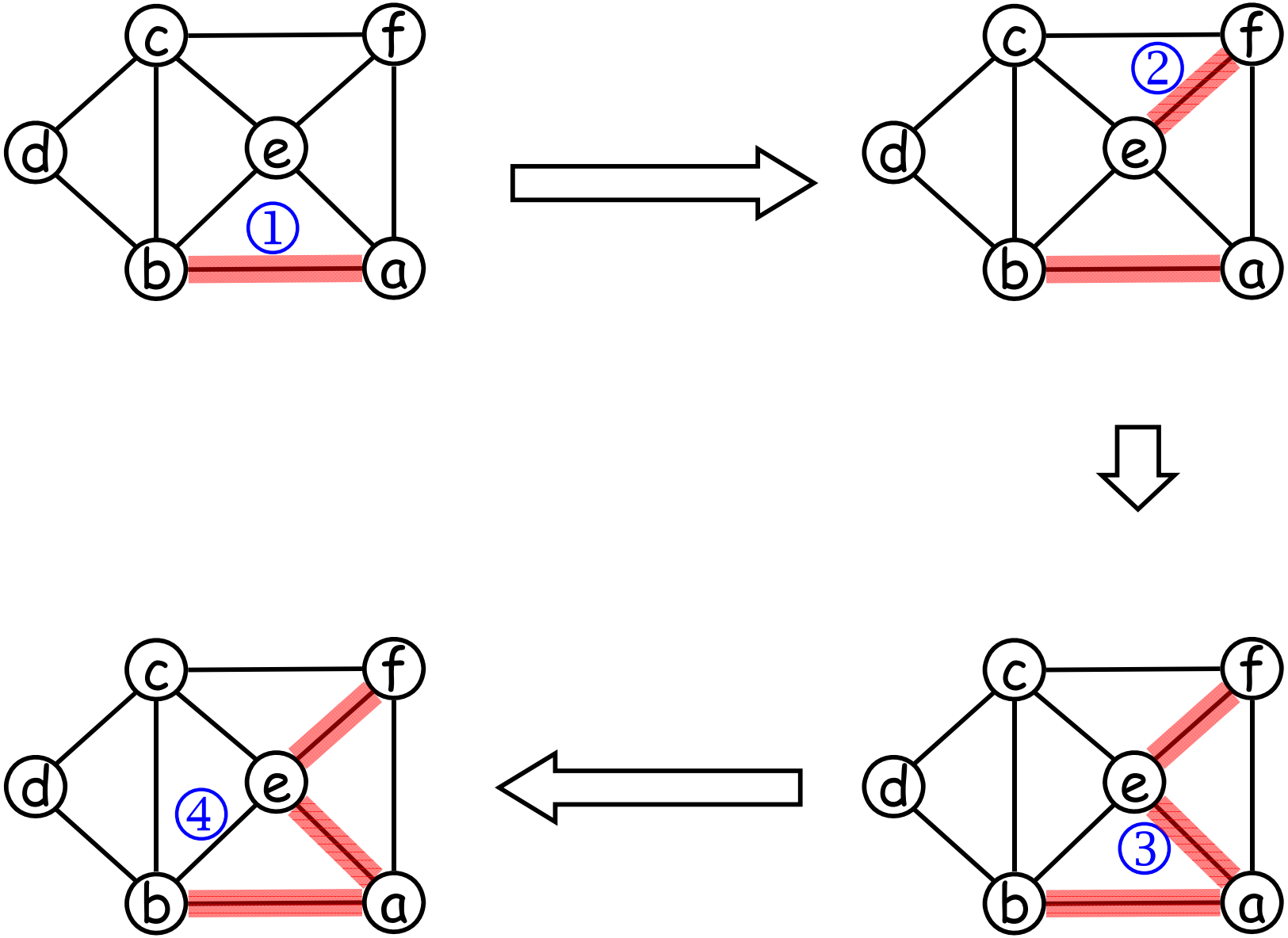
(an MST)

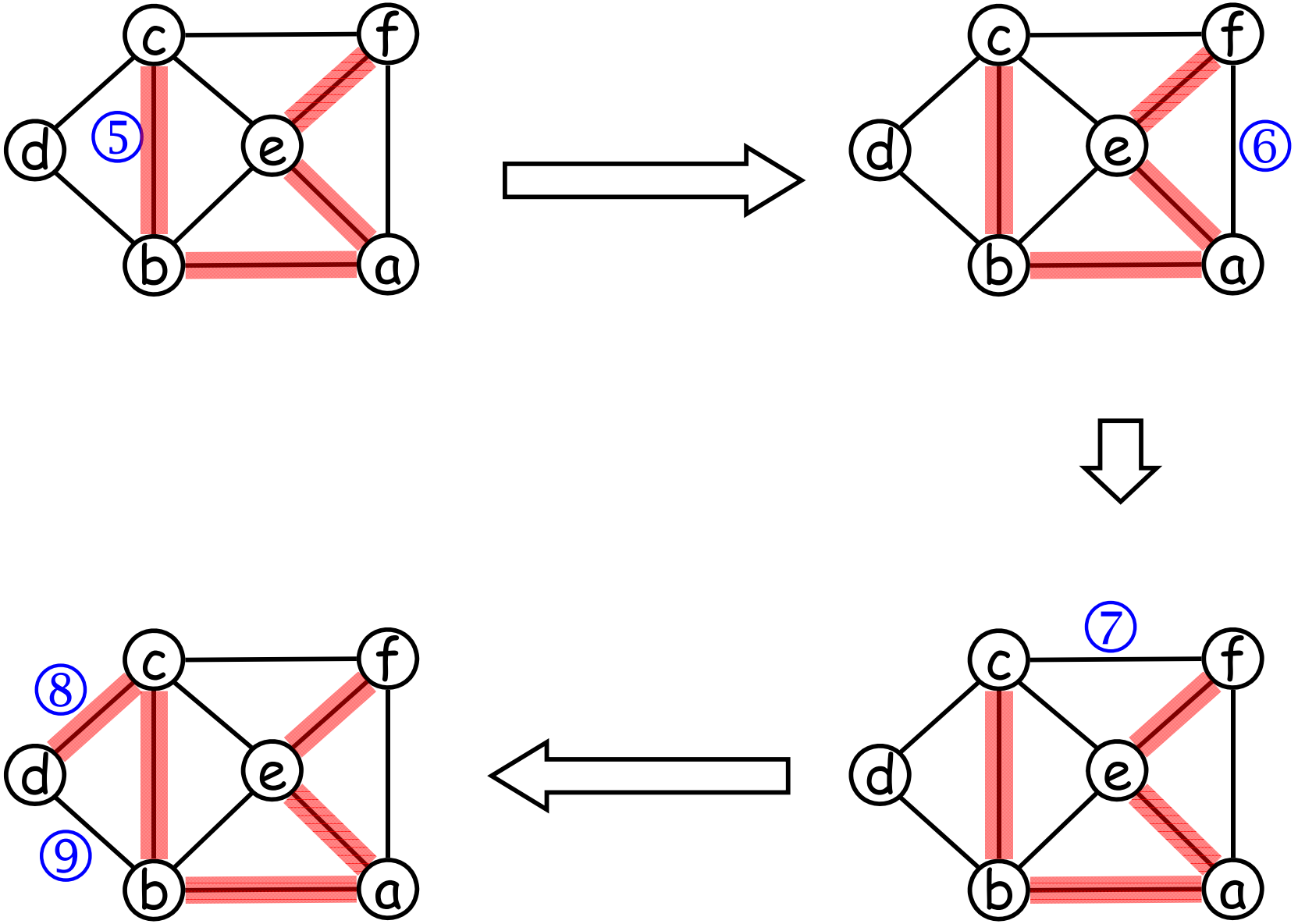
Algorithm: (Greedy Method, Kruskal)

Step 1: sort edges by lengths

Step 2: for i = 1 to |E|

select the i-th edge as a tree edge if it induces no cycle.

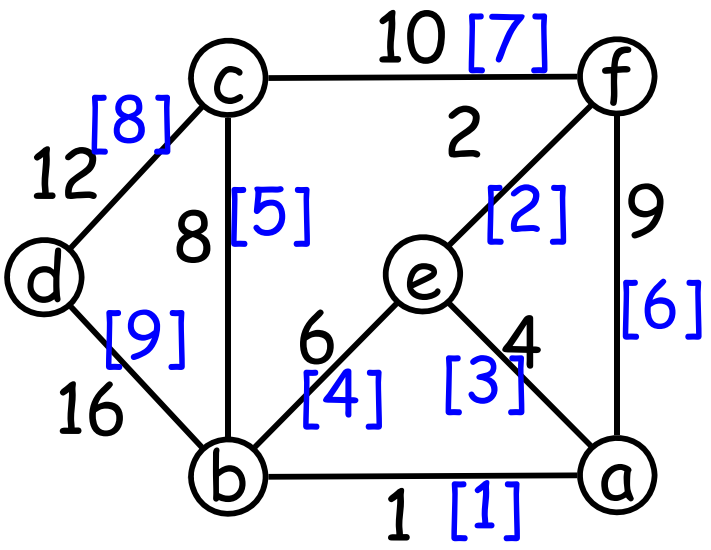

c
f
d
e
b
a
①
②
③
④



an MST

Problem: Minimum Spanning Trees

$G = (V, E)$



Algorithm: (Greedy Method, Kruskal)

Step 1: sort edges by lengths

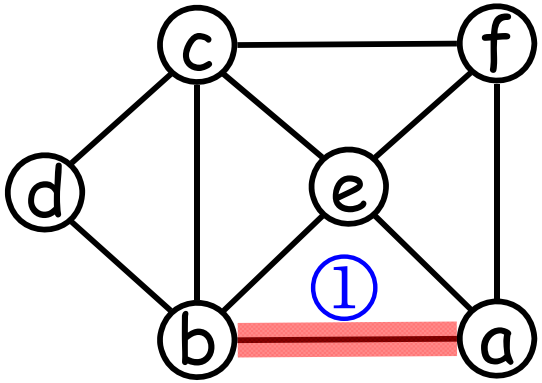
Step 2: for i = 1 to |E|

select the i-th edge as a tree edge if it induces no cycle.

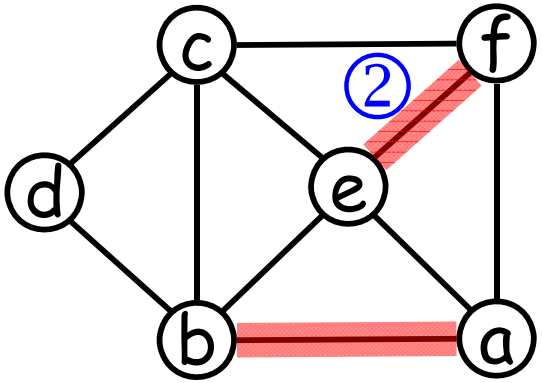
Data structure:

① graph G: matrix, lists

② sets:



{a} ∪ {b}



{a}, {b}, {c}, {d}, {e}, {f}

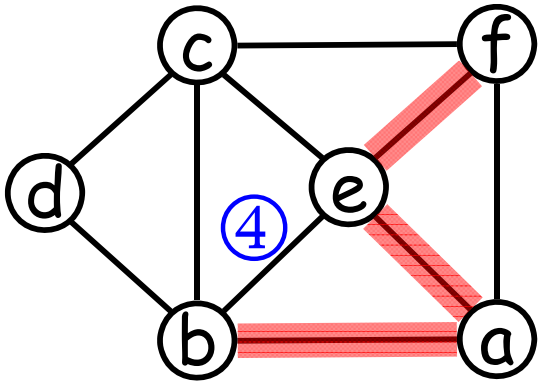
{a, b}, {c}, {d}, {e}, {f}

{e} ∪ {f}

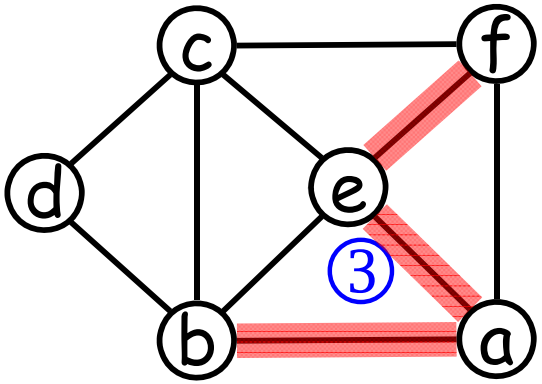
unchanged for ④

{a, b, e, f}, {c}, {d}

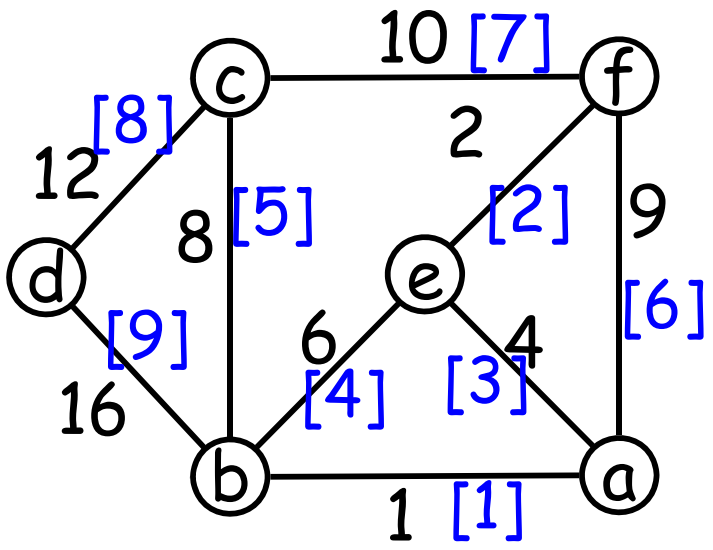
{a, b}, {c}, {d}, {e, f}



{a, b} ∪ {e, f}



Problem: Minimum Spanning Trees



Algorithm: (Greedy Method, Kruskal)

Step 1: sort edges by lengths

Step 2: for i = 1 to |E|

select the i-th edge as a tree edge
if it induces no cycle.
(two ends are in different sets)

Data structure:

① graph G: matrix, lists

② sets: array, list, tree, …

Analysis: ①list ②tree ➔ O(|E| lg |E|)

- Algorithm
  - Design
    - Method (Algorithm)
      - Techniques: Divide-and-Conquer, Dynamic Programming, Greedy Method, Branch-and-Bound, Approximation, Amortized Analysis, ...
      - Fundamental Problems: Sorting, Shortest Path, Minimum Spanning Tree, Maximum Flow, Convex hull, ...
    - Data Structure: Array, List, Stack, Binary tree, Heap, Disjoint Sets, B-tree, ...
  - Analysis (高 中 數 學)