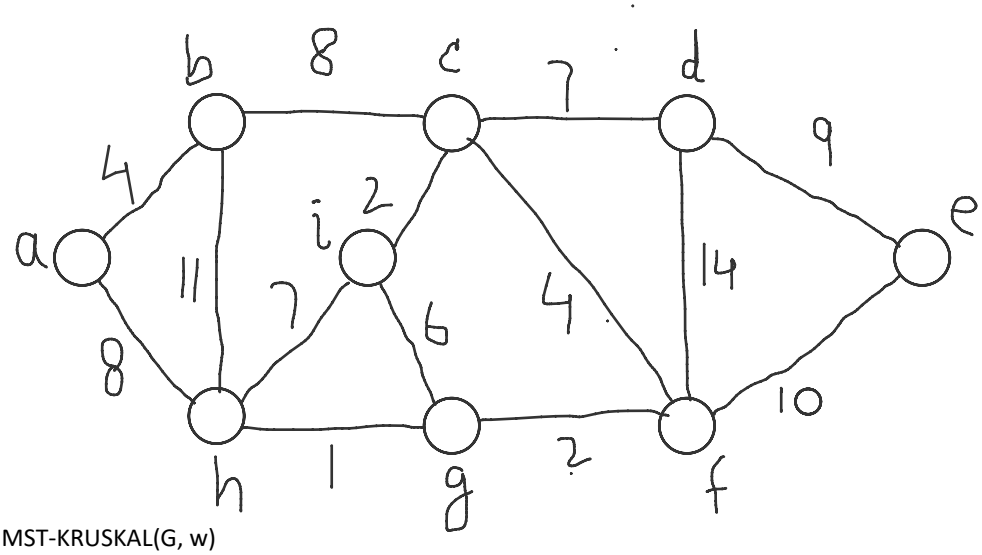
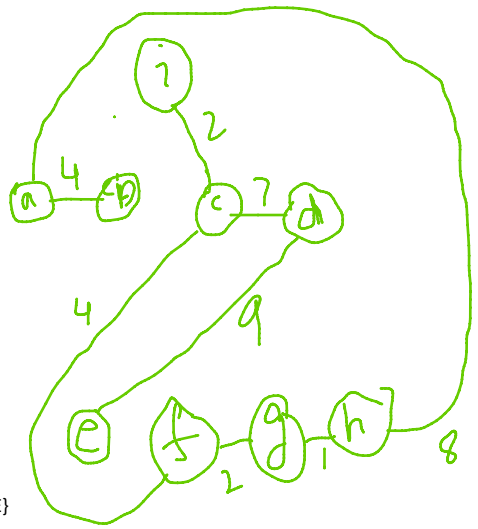
Kruskal Algorithm (MST)

Thursday, June 25, 2020 7:59 AM

4+8+2+1++2+4+7+9=37



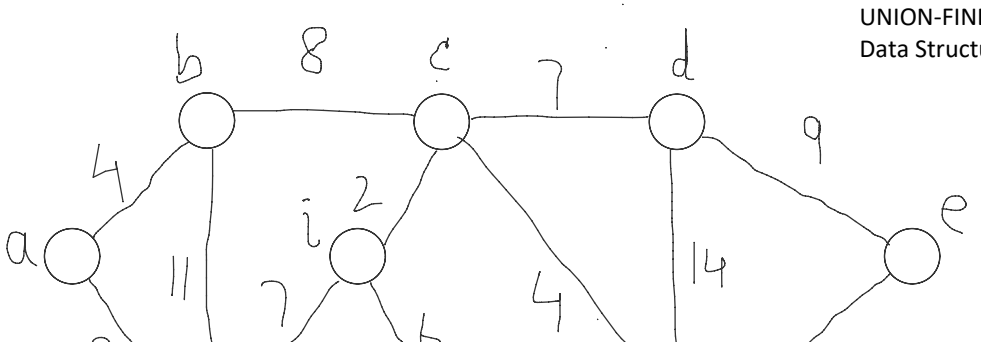


MST-KRUSKAL(G, w)

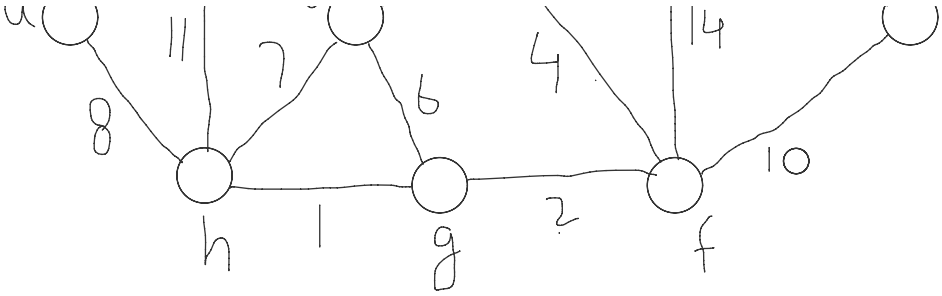
1. A = ∅;
2. for each vertex v ∈ G. V
3. MAKE-SET(v)
4. sort the edges of G. E into nondecreasing order by weight w
5. for each edge (u, v) ∈ G. E, taken in nondecreasing order by weight
6. if FIND-SET(u) ≠ FIND-SET(v)
7. A = A U {(u, v)}
8. UNION(u, v)
9. return A
10. // {g, h}, {c, i}, {f, g}, {a, b}, {c, f}, {i, g}, {c, d},{i, h}, {a, h}, {b, c}, {d, e}, {f, e}, {d, f}

{H, G, F, C, I, D, A, B, E}

A ={ {g, h}, {c, i}, {f, g}, {a, b}, {c, f}, {c, d}, {a, h}, {d, e},}



UNION-FIND Disjoint Sets Data Structure

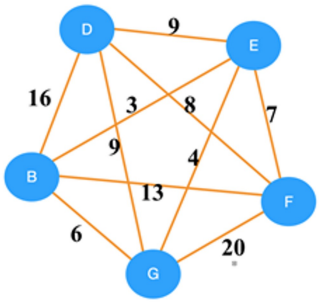


{a, b}
{d}
{e}
{h, g, f, i, c}

A= {{h, g},{i, c},{f, g},{a, b},{c, f},}

MST-KRUSKAL(G, w)

1. A = Ø;
2. for each vertex v ∈ G. V
3. MAKE-SET(v)
4. sort the edges of G. E into nondecreasing order by weight w
5. for each edge (u, v) ∈ G. E, taken in nondecreasing order by weight
6. if FIND-SET(u) ≠ FIND-SET(v)
7. A = A U {(u, v)}
8. UNION(u, v)
9. return A
10.



1. // {B,E},{E,G},{B,G},{E,F},{D,F},{D,G},{D,E},{B,F},{B,D},{F,G}
2. // 3 4 6 7 8 9 9 13 16 20