Lecture 21

G is connected, undirected.

```
kruskal-mst (G = (V, E), w : E → R)
  T ← {}
  sort edges
  for each v in V:
    make-set(v)
  for i ← 1 to m:
    if findset(u_i) != findset(v_i):
      union(u_i, v_i)
      T ← ∪{e_i}
```

Disjoint Set ADT

`make-set(`x`)`: create a new set $\{x\}$ that contains only x and makes x the representative

`findset(`x`)`: returns the representative of the set that contains x

`union(`x, y`)`: puts two sets together, picks new representative

Hashtable approach

key vertex, value rep

f : a
a : a
b : b
c : b
d : d
e : d

a→[a]
b→[b,c]
d→[d,e,f]

Linked list approach

A
$B \to C$
$D \to E \to F$

circular linked list? → for `findset(`x`)`
→ use doubly linked list insteadt, go back:
A
$B \leftrightarrow C$
$D \leftrightarrow E \leftrightarrow F$

`union(C,E):`

doubly linked list, each item has pointer to rep
then point B to D → better `union`, but worse `findset`