


Lecture 10.2: Kruskal's Algorithm

```
Kruskal()  
{  
    T =  $\emptyset$ ;  
    for each v  $\in$  V  
        MakeSet(v);  
    sort E by increasing edge weight w  
    for each (u,v)  $\in$  E (in sorted order)  
        if FindSet(u)  $\neq$  FindSet(v)  
            T = T  {{u,v}};  
            Union(FindSet(u), FindSet(v));  
}
```

Kruskal's Algorithm

```
Kruskal()
```

```
{
```

```
    T =  $\emptyset$ ;
```

```
    for each v  $\in$  V
```

```
        MakeSet(v);
```

```
    sort E by increasing edge weight w
```

```
    for each (u,v)  $\in$  E (in sorted order)
```

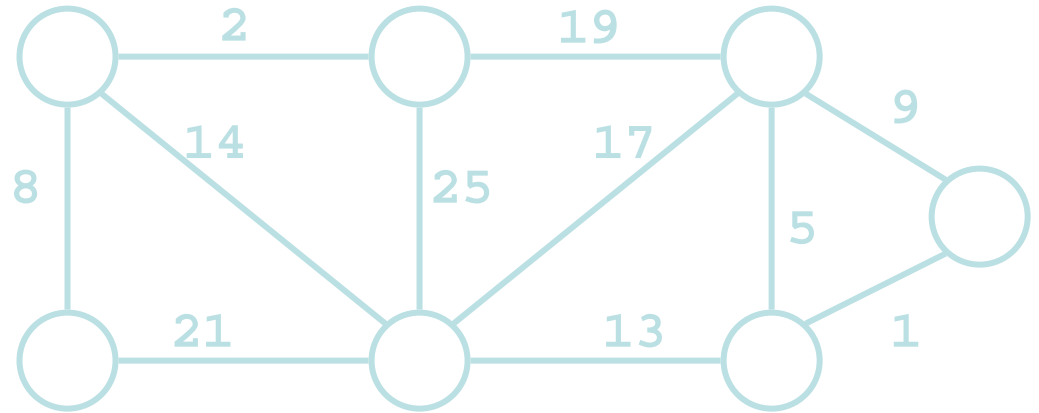
```
        if FindSet(u)  $\neq$  FindSet(v)
```

```
            T = T  $\cup$  {{u,v}};
```

```
            Union(FindSet(u), FindSet(v));
```

```
}
```

Run the algorithm:



Kruskal's Algorithm

Kruskal()

{

$T = \emptyset;$

{

for each $v \in V$

MakeSet(v);

sort E by increasing edge weight w

for each $(u,v) \in E$ (in sorted order)

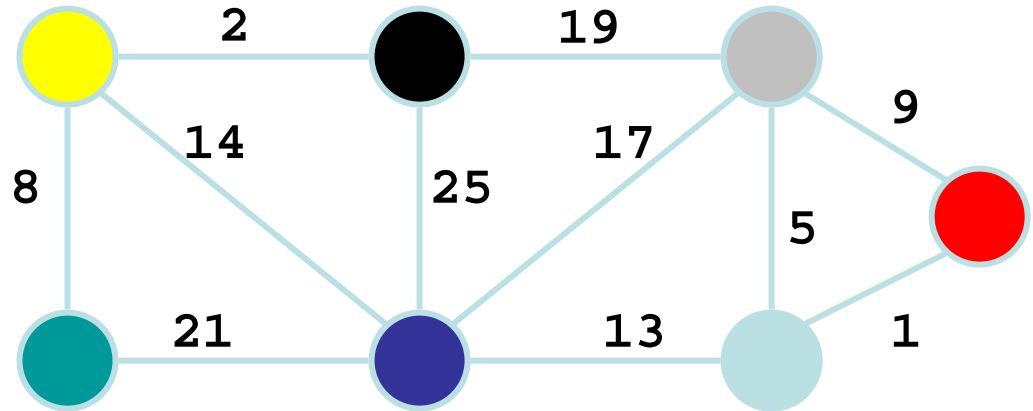
if FindSet(u) \neq FindSet(v)

$T = T \cup \{u,v\};$

Union(FindSet(u), FindSet(v));

}

Run the algorithm:

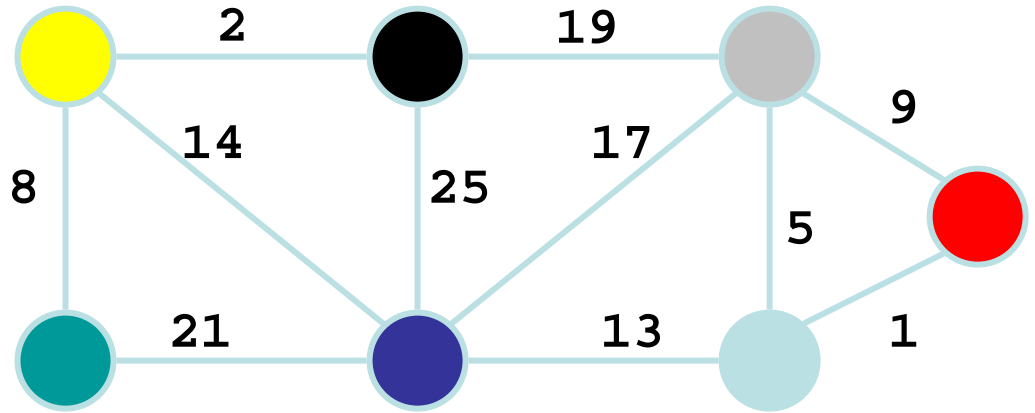


Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{  
    T =  $\emptyset$ ;  
    for each v  $\in$  V  
        MakeSet(v);  
    { sort E by increasing edge weight w  
      for each (u,v)  $\in$  E (in sorted order)  
        if FindSet(u)  $\neq$  FindSet(v)  
            T = T  $\cup$  {{u,v}};  
            Union(FindSet(u), FindSet(v));  
    }
```



Kruskal's Algorithm

Kruskal()

{

$T = \emptyset;$

for each $v \in V$

 MakeSet(v);

sort E by increasing edge weight w

for each $(u,v) \in E$ (in sorted order)

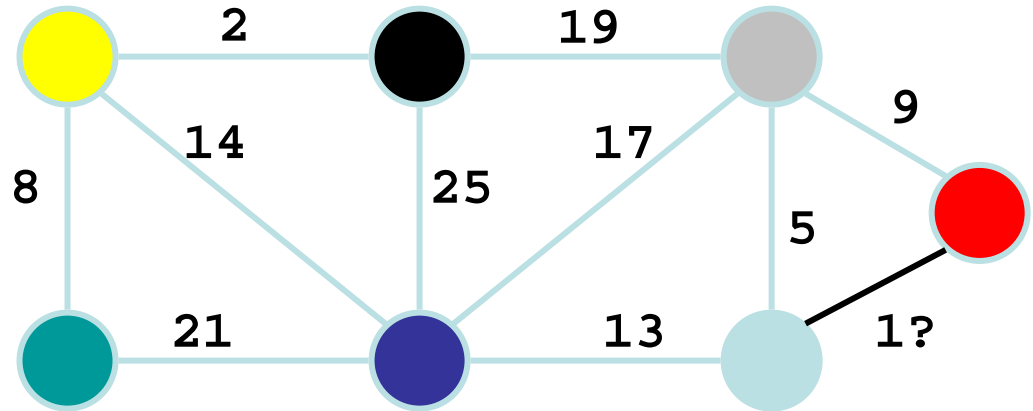
 if FindSet(u) \neq FindSet(v)

$T = T \cup \{u,v\};$

 Union(FindSet(u), FindSet(v));

}

Run the algorithm:



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
    T =  $\emptyset$ ;
```

```
    for each v  $\in$  V
```

```
        MakeSet(v);
```

```
    sort E by increasing edge weight w
```

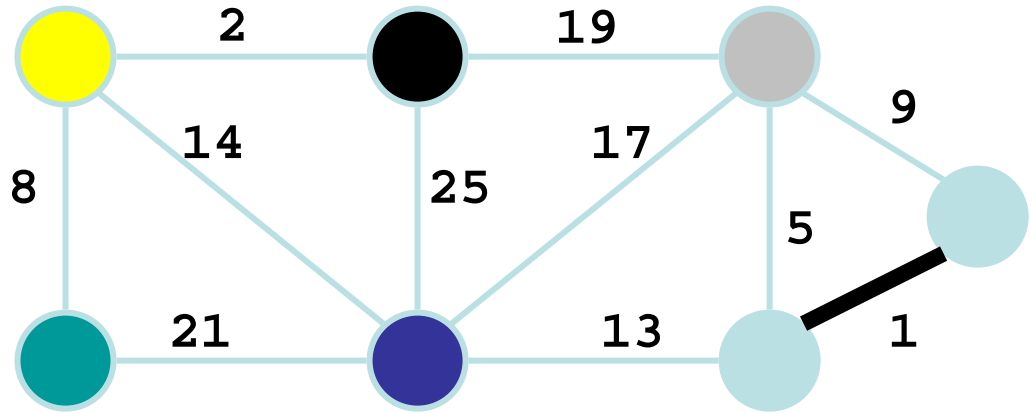
```
    { for each (u,v)  $\in$  E (in sorted order)
```

```
        if FindSet(u)  $\neq$  FindSet(v)
```

```
            T = T  $\cup$  {(u,v)};
```

```
            Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

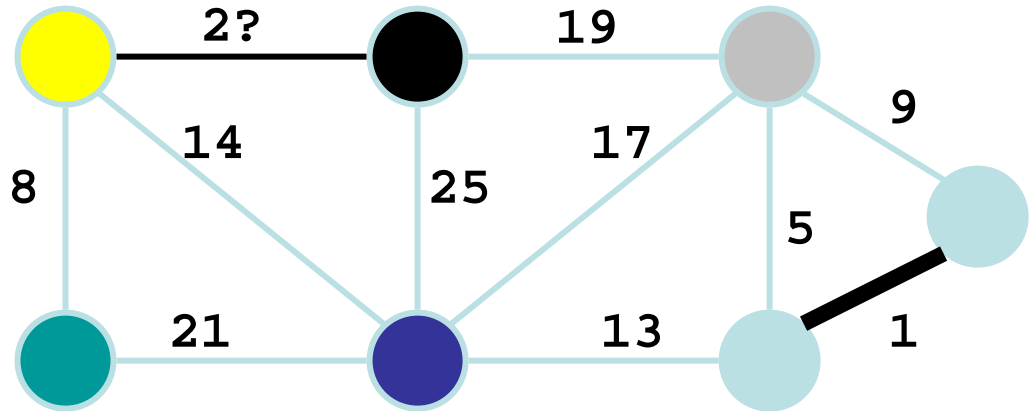
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

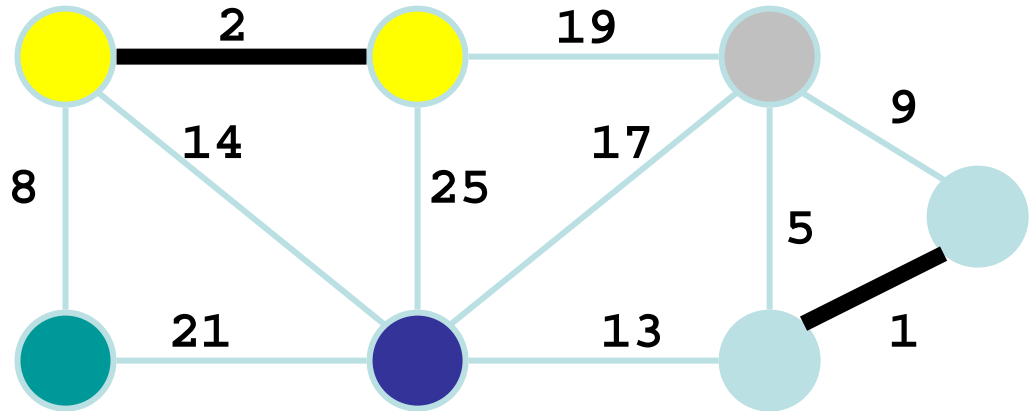
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {(u,v)};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

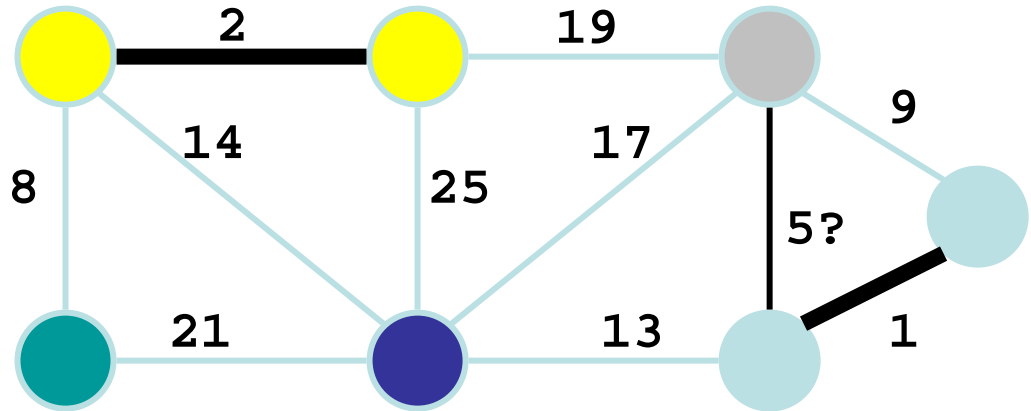
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each  $v \in V$ 
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

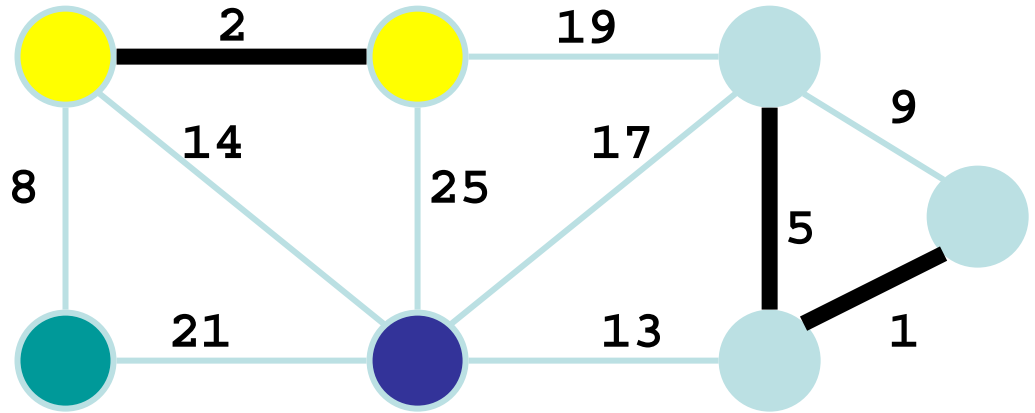
```
  { for each  $(u,v) \in E$  (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$   $\{\{u,v\}\}$ ;
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

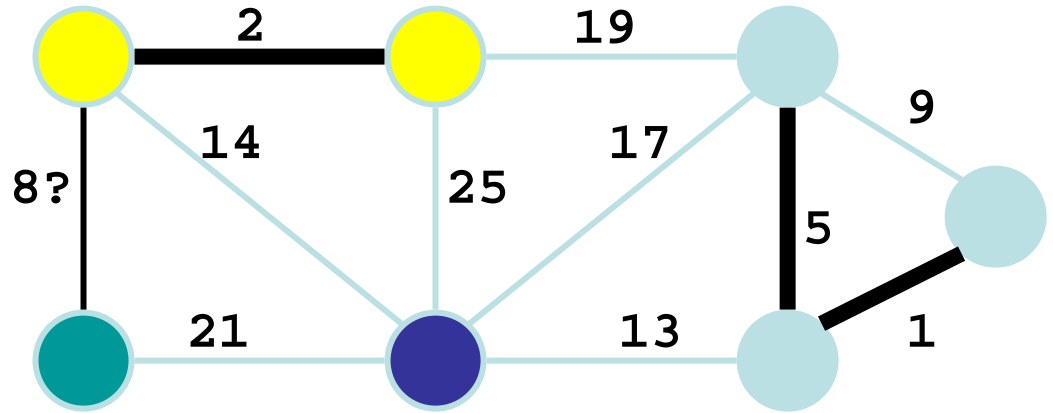
```
  for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

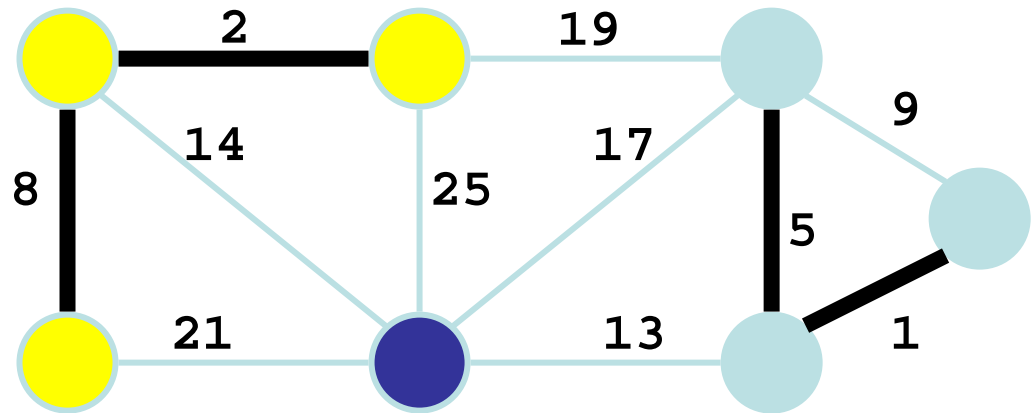
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {(u,v)};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

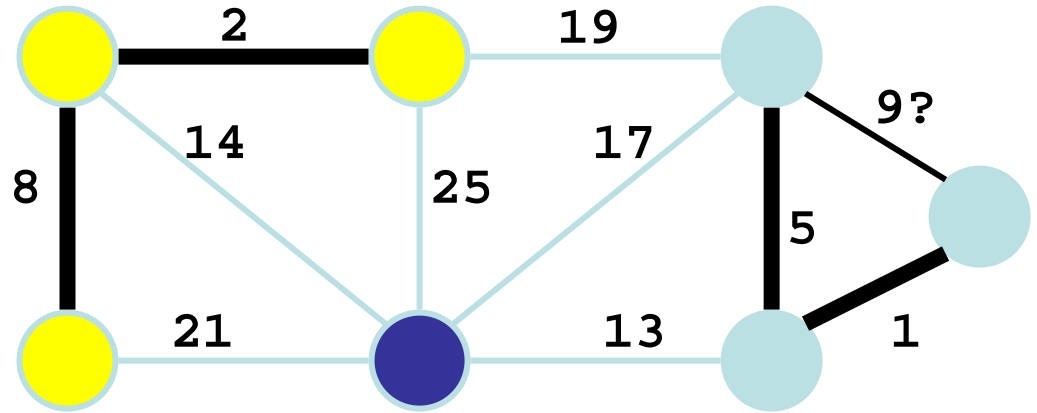
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

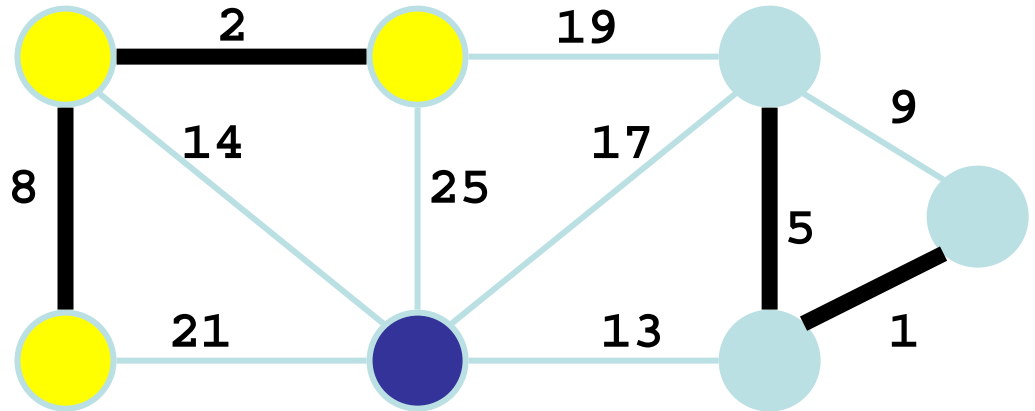
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

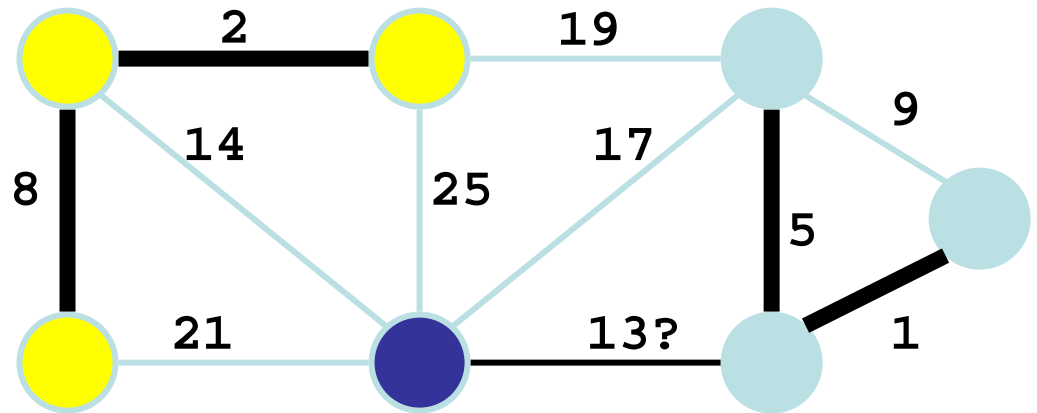
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

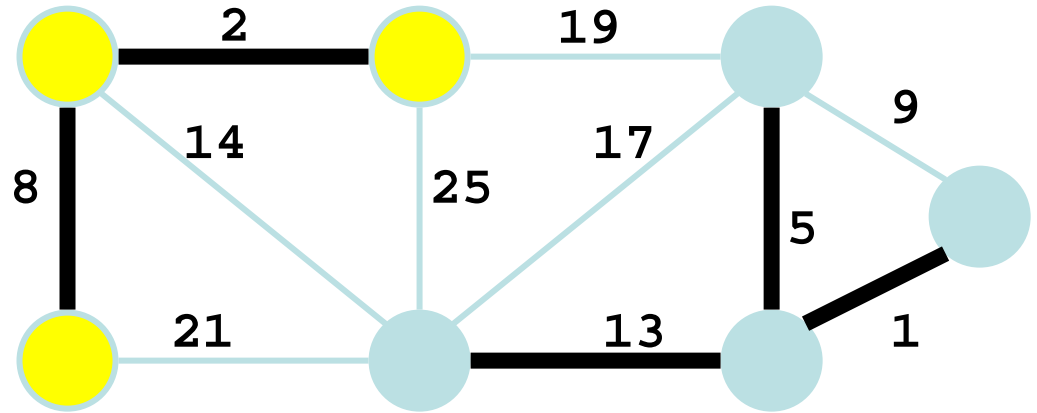
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
  T =  $\emptyset$ ;
```

```
  for each v  $\in$  V
```

```
    MakeSet(v);
```

```
  sort E by increasing edge weight w
```

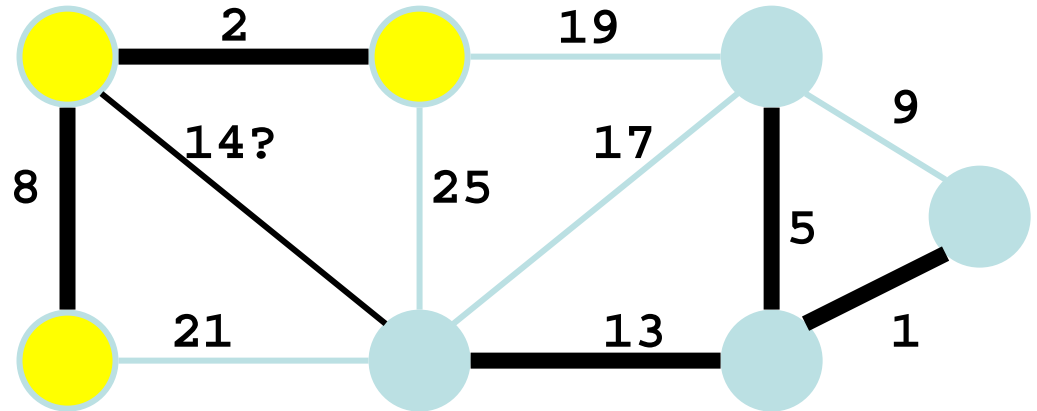
```
  { for each (u,v)  $\in$  E (in sorted order)
```

```
    if FindSet(u)  $\neq$  FindSet(v)
```

```
      T = T  $\cup$  {{u,v}};
```

```
      Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

Run the algorithm:

```
Kruskal()
```

```
{
```

```
    T =  $\emptyset$ ;
```

```
    for each v  $\in$  V
```

```
        MakeSet(v);
```

```
    sort E by increasing edge weight w
```

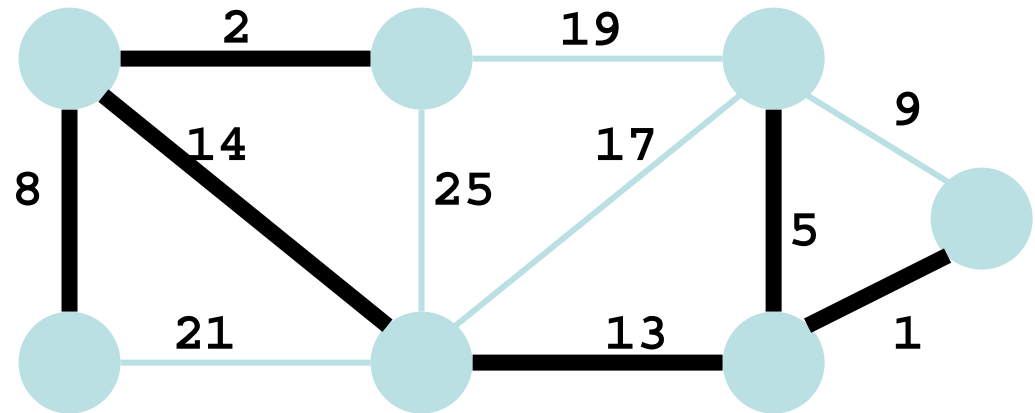
```
    for each (u,v)  $\in$  E (in sorted order)
```

```
        if FindSet(u)  $\neq$  FindSet(v)
```

```
            T = T  $\cup$  {{u,v}};
```

```
            Union(FindSet(u), FindSet(v));
```

```
}
```



Kruskal's Algorithm

```
Kruskal()
```

```
{
```

```
    T =  $\emptyset$ ;
```

```
    for each v  $\in$  V
```

```
        MakeSet(v);
```

```
    sort E by increasing edge weight w
```

```
    { for each (u,v)  $\in$  E (in sorted order)
```

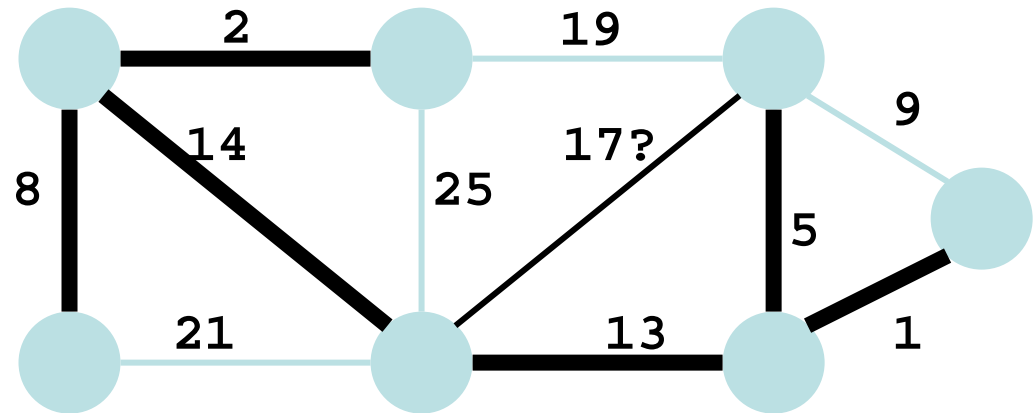
```
        if FindSet(u)  $\neq$  FindSet(v)
```

```
            T = T  $\cup$  {{u,v}};
```

```
            Union(FindSet(u), FindSet(v));
```

```
}
```

Run the algorithm:



Kruskal's Algorithm

Kruskal()

{

$T = \emptyset;$

for each $v \in V$

 MakeSet(v);

sort E by increasing edge weight w

for each $(u,v) \in E$ (in sorted order)

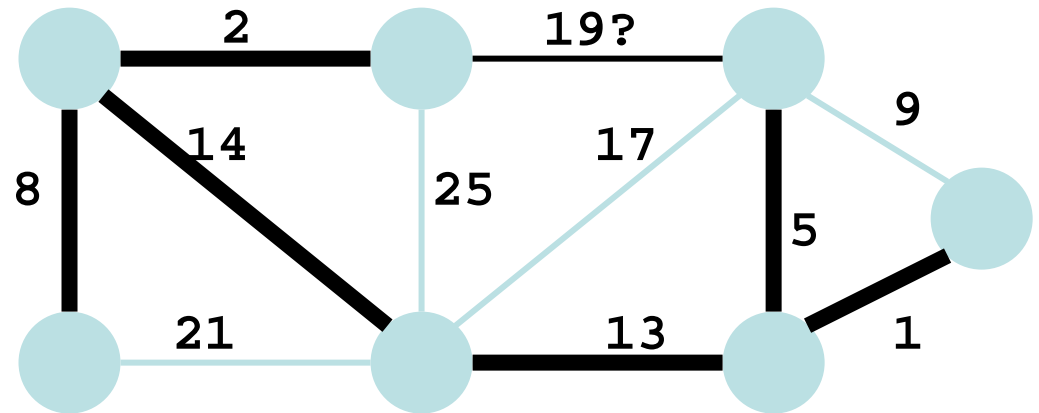
 if FindSet(u) \neq FindSet(v)

$T = T \cup \{(u,v)\};$

 Union(FindSet(u), FindSet(v));

}

Run the algorithm:



Kruskal's Algorithm

Kruskal()

{

$T = \emptyset;$

for each $v \in V$

 MakeSet(v);

sort E by increasing edge weight w

for each $(u,v) \in E$ (in sorted order)

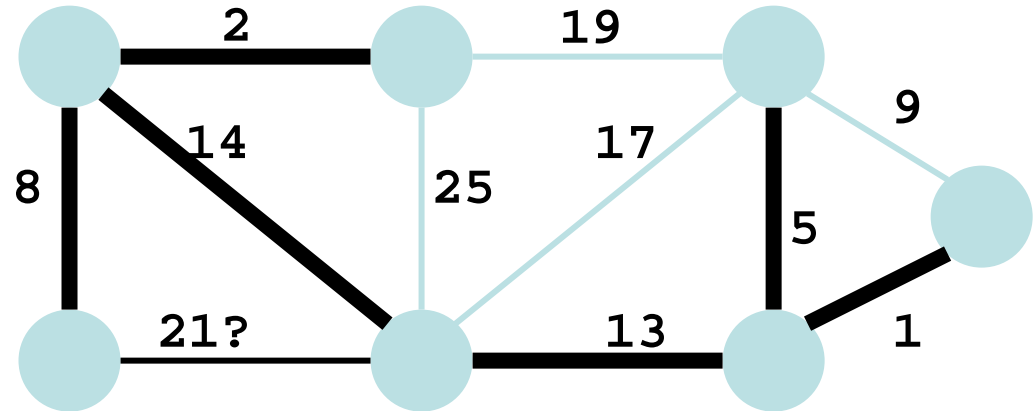
 if FindSet(u) \neq FindSet(v)

$T = T \cup \{(u,v)\};$

 Union(FindSet(u), FindSet(v));

}

Run the algorithm:



Kruskal's Algorithm

Kruskal()

{

$T = \emptyset;$

for each $v \in V$

 MakeSet(v);

sort E by increasing edge weight w

for each $(u,v) \in E$ (in sorted order)

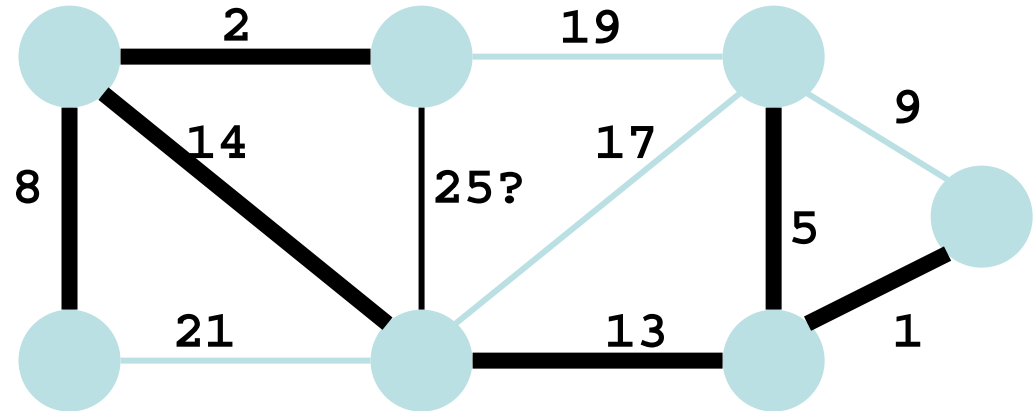
 if FindSet(u) \neq FindSet(v)

$T = T \cup \{(u,v)\};$

 Union(FindSet(u), FindSet(v));

}

Run the algorithm:



Kruskal's Algorithm

Kruskal()

{

$T = \emptyset;$

for each $v \in V$

 MakeSet(v);

sort E by increasing edge weight w

for each $(u,v) \in E$ (in sorted order)

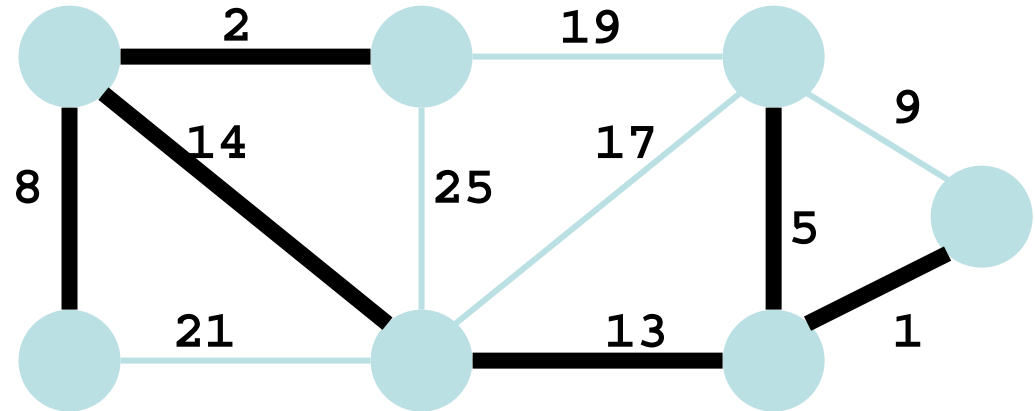
 if FindSet(u) \neq FindSet(v)

$T = T \cup \{(u,v)\};$

 Union(FindSet(u), FindSet(v));

}

Run the algorithm:



Kruskal's Algorithm

```
Kruskal()
```

```
{
```

```
    T =  $\emptyset$ ;
```

```
    for each v  $\in$  V
```

```
        MakeSet(v);
```

```
    sort E by increasing edge weight w
```

```
    for each (u,v)  $\in$  E (in sorted order)
```

```
        if FindSet(u)  $\neq$  FindSet(v)
```

```
            T = T  $\cup$  {{u,v}};
```

```
            Union(FindSet(u), FindSet(v));
```

```
}
```

Run the algorithm:

