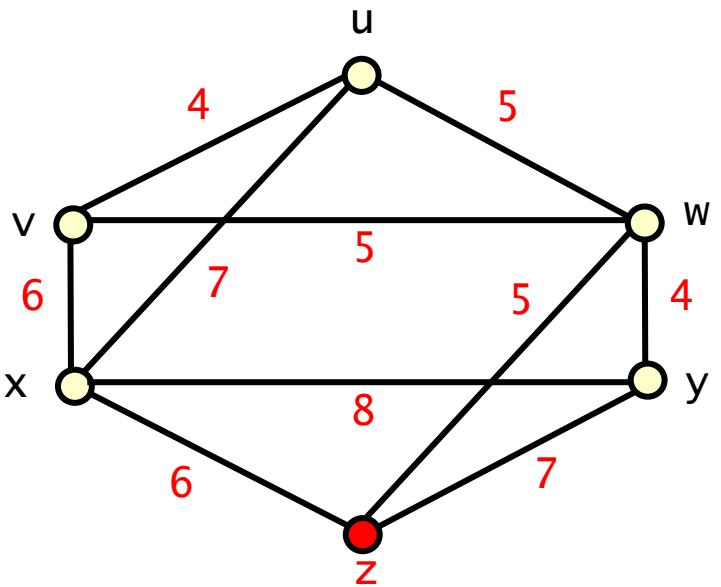
Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which
  wt({q, q.bestTV}) is minimal
  and make q a **tv**
- update s.bestTV for all **ntv** s

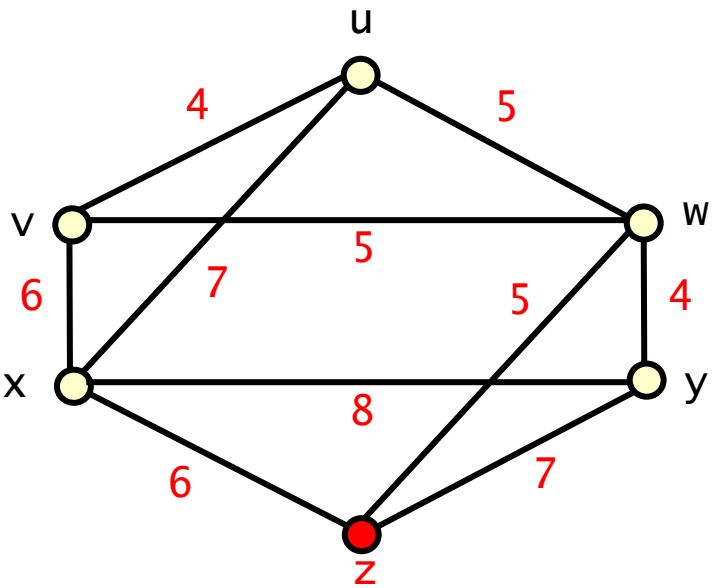


tree vertices: z
non-tree vertices: u,v,w,x,y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	z	∞
v	z	∞
w	z	5
x	z	6
y	z	7
z	-	-

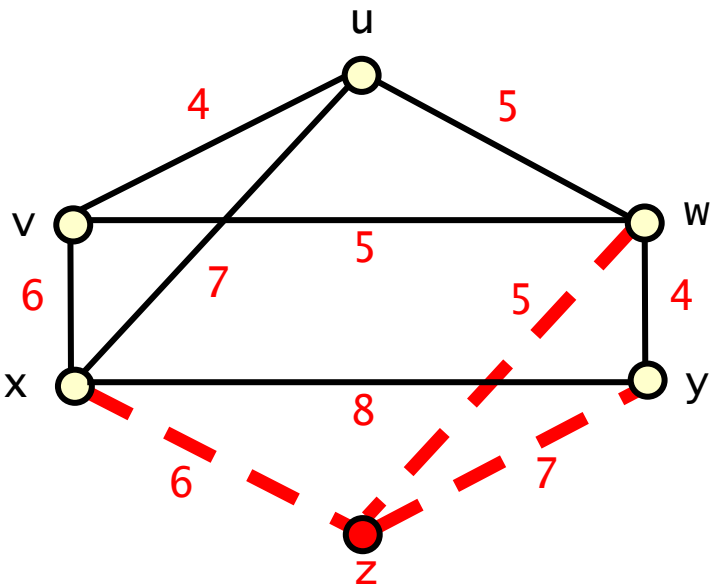
tree vertices: z
non-tree vertices: u,v,w,x,y

initialise **bestTV** to the only **tv** z

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



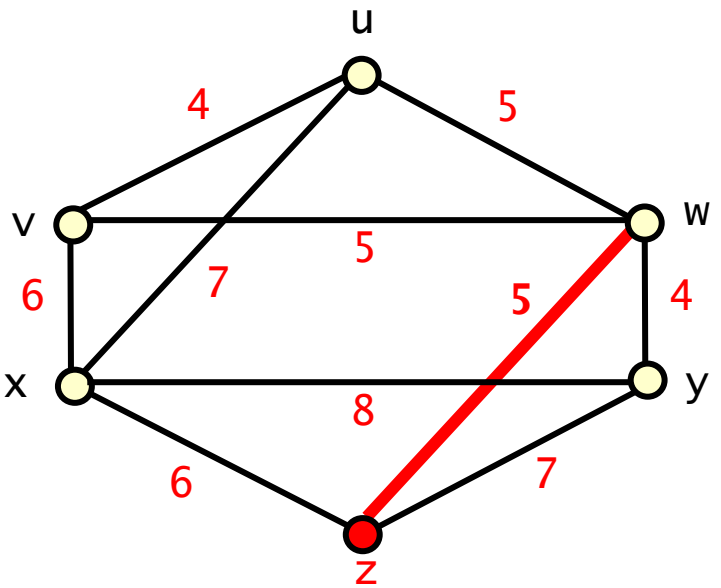
q	q.bestTV	wt({q.bestTV,q})
u	z	∞
v	z	∞
w	z	5
x	z	6
y	z	7
z	-	-

tree vertices: z
non-tree vertices: u,v,w,x,y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



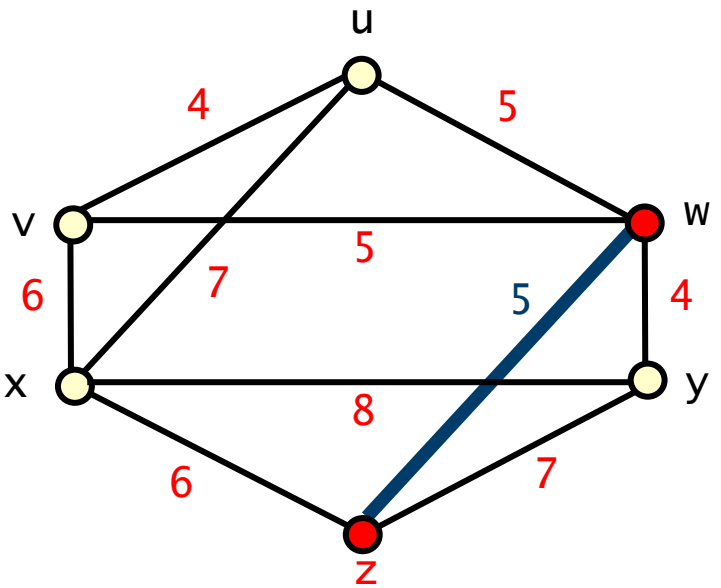
q	q.bestTV	wt({q.bestTV,q})
u	z	∞
v	z	∞
w	z	5
x	z	6
y	z	7
z	-	-

tree vertices: z
non-tree vertices: u,v,w,x,y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



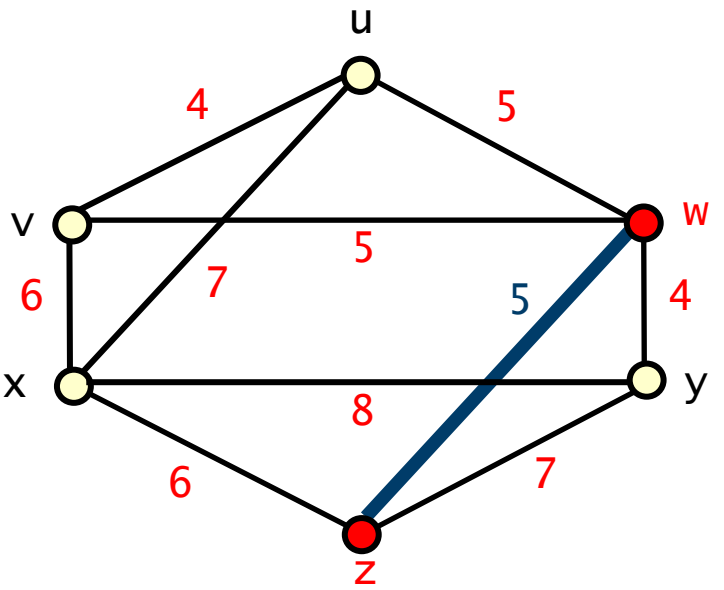
q	q.bestTV	wt({q.bestTV,q})
u	z	∞
v	z	∞
w	z	5
x	z	6
y	z	7
z	-	-

tree vertices: w, z
non-tree vertices: u, v, x, y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which `wt({q, q.bestTV})` is minimal and make q a **tv**
- update `s.bestTV` for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	z→w	∞→5
v	z→w	∞→5
w	-	-
x	z	6
y	z→w	7→4
z	-	-

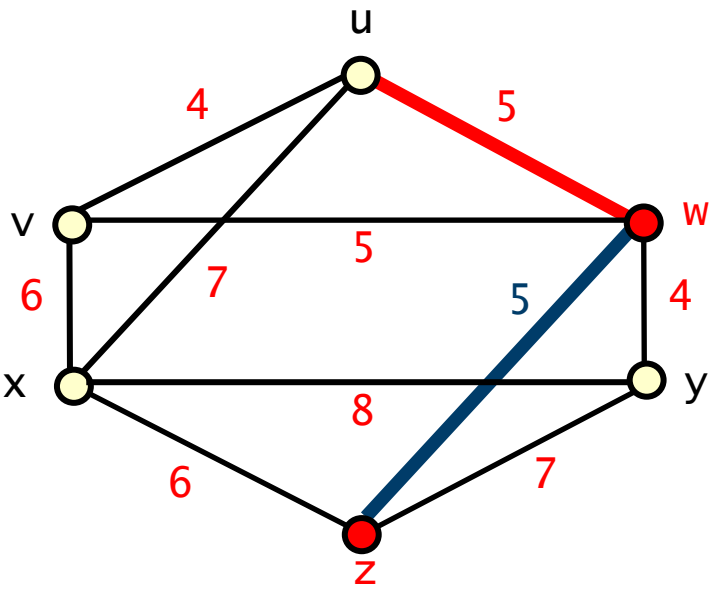
tree vertices: w, z
non-tree vertices: u, v, x, y

check bestTV against new tv vertex w

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which `wt({q, q.bestTV})` is minimal and make q a **tv**
- update `s.bestTV` for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	**z→w**	**∞→5**
v	z→w	∞→5
w	-	-
x	z	6
y	z→w	7→4
z	-	-

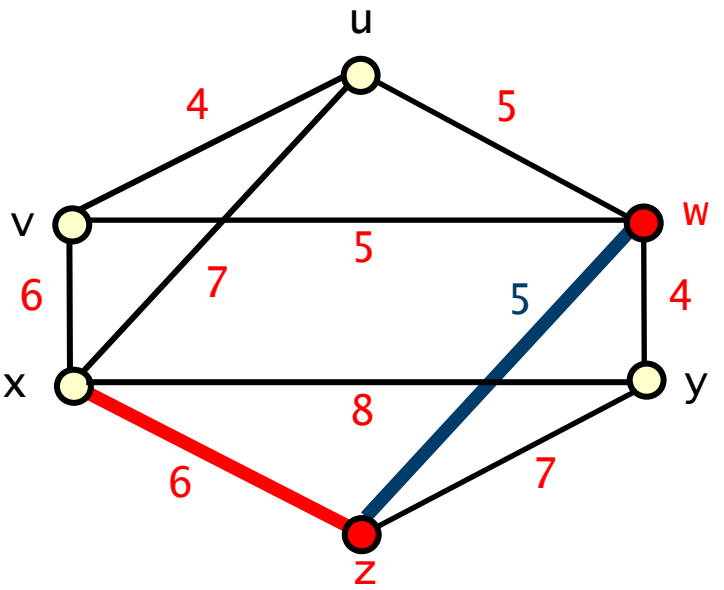
tree vertices: w, z
non-tree vertices: u, v, x, y

check bestTV against new tv vertex w

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	z→w	∞→5
v	z→w	∞→5
w	-	-
x	**z**	**6**
y	z→w	7→4
z	-	-

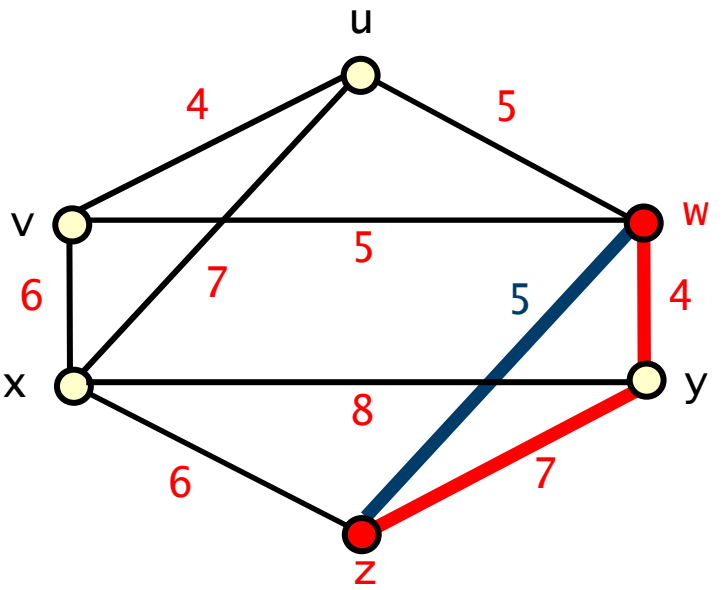
tree vertices: w, z
non-tree vertices: u, v, x, y

check bestTV against new tv vertex w

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which `wt({q, q.bestTV})` is minimal and make q a **tv**
- update `s.bestTV` for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	z→w	∞→5
v	z→w	∞→5
w	-	-
x	z	6
y	**z→w**	**7→4**
z	-	-

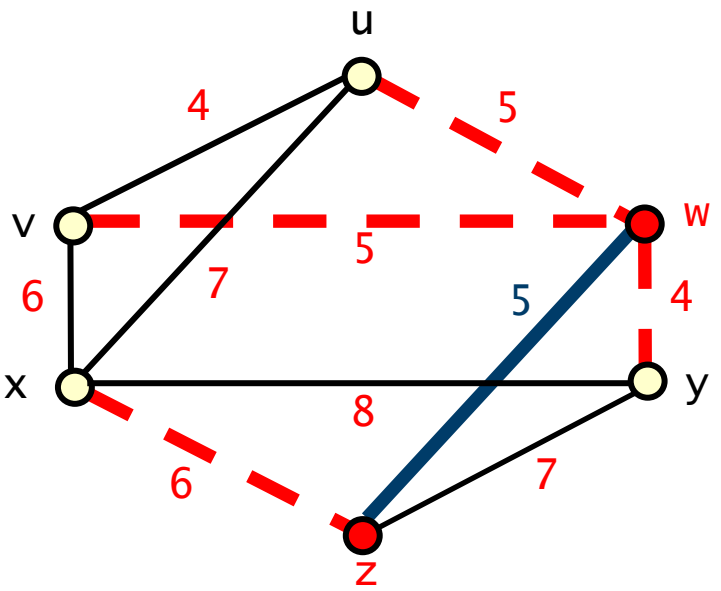
tree vertices: w, z
non-tree vertices: u, v, x, y

check bestTV against new tv vertex w

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



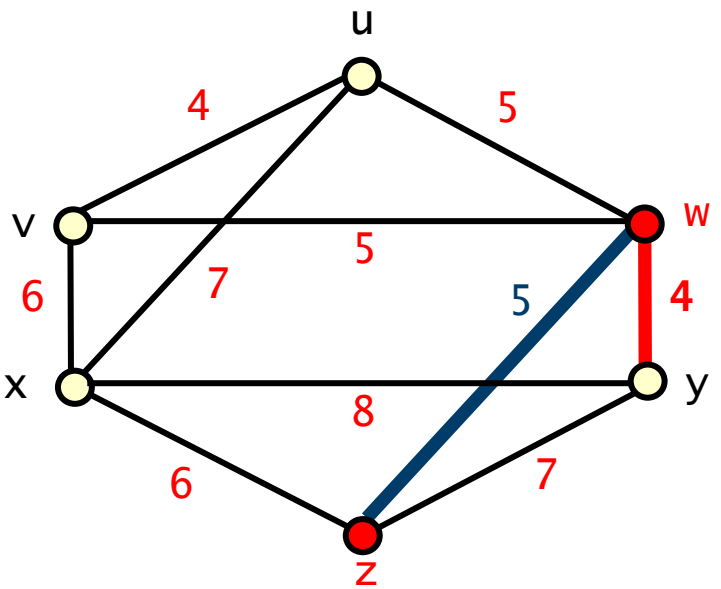
q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	w	4
z	-	-

tree vertices: w, z
non-tree vertices: u, v, x, y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



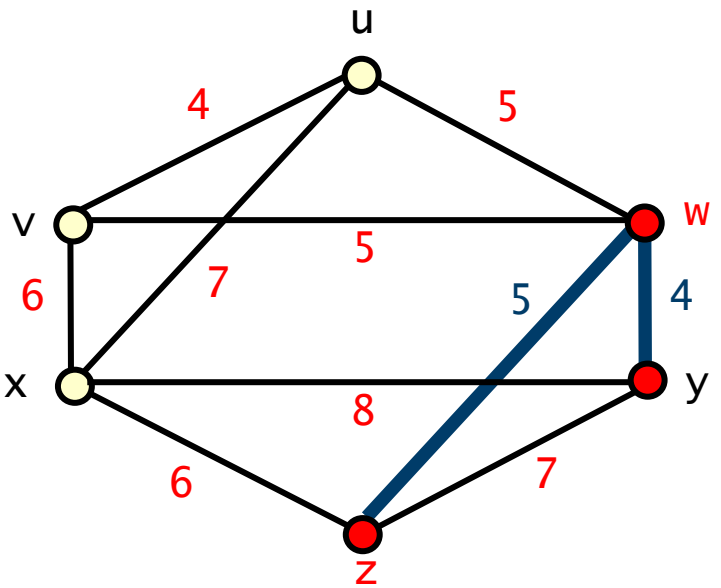
q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	**w**	**4**
z	-	-

tree vertices: w, z
non-tree vertices: u, v, x, y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



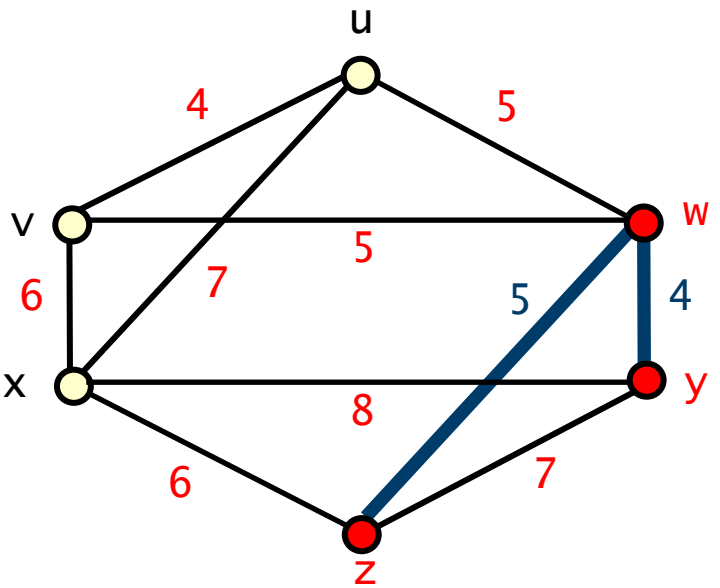
q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	w	4
z	-	-

tree vertices: w, y, z
non-tree vertices: u, v, x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	-	-
z	-	-

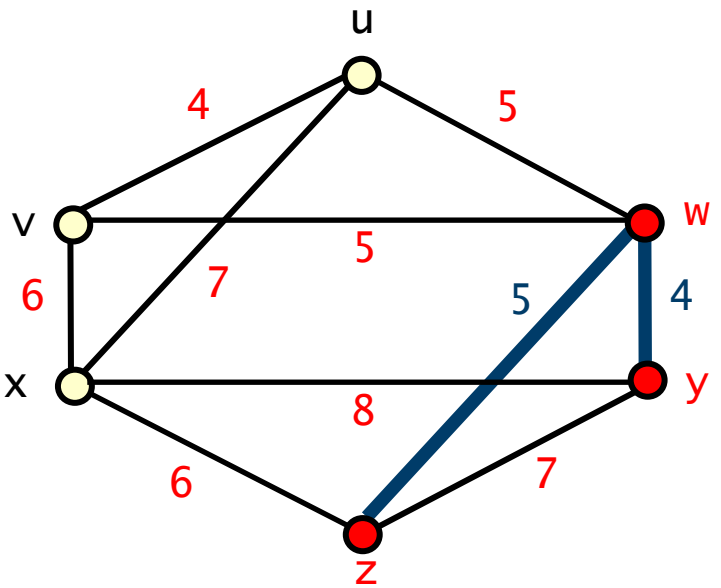
tree vertices: w, y, z
non-tree vertices: u, v, x

check bestTV against new tv vertex y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	**w**	**5**
v	**w**	**5**
w	-	-
x	z	6
y	-	-
z	-	-

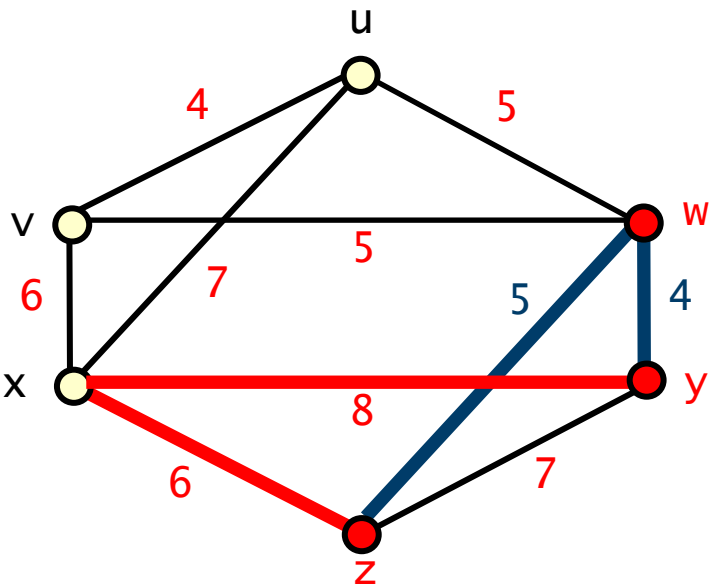
tree vertices: w, y, z
non-tree vertices: u, v, x

check bestTV against new tv vertex y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	**z**	**6**
y	-	-
z	-	-

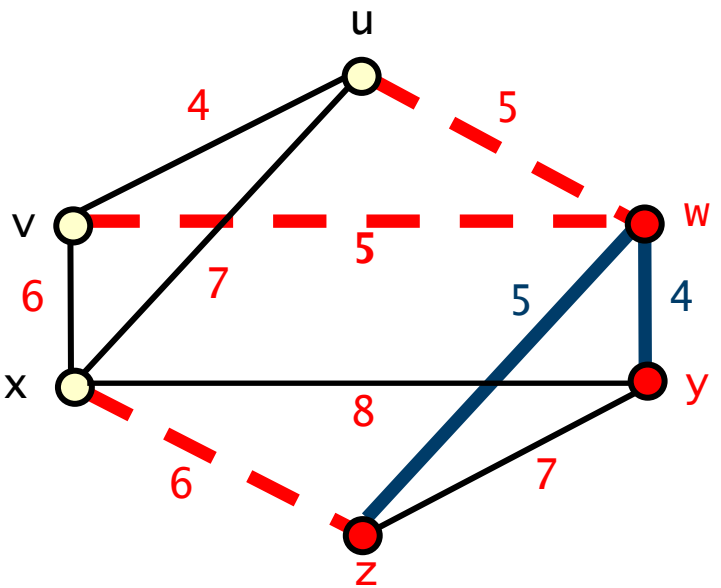
tree vertices: w, y, z
non-tree vertices: u, v, x

check bestTV against new tv vertex y

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



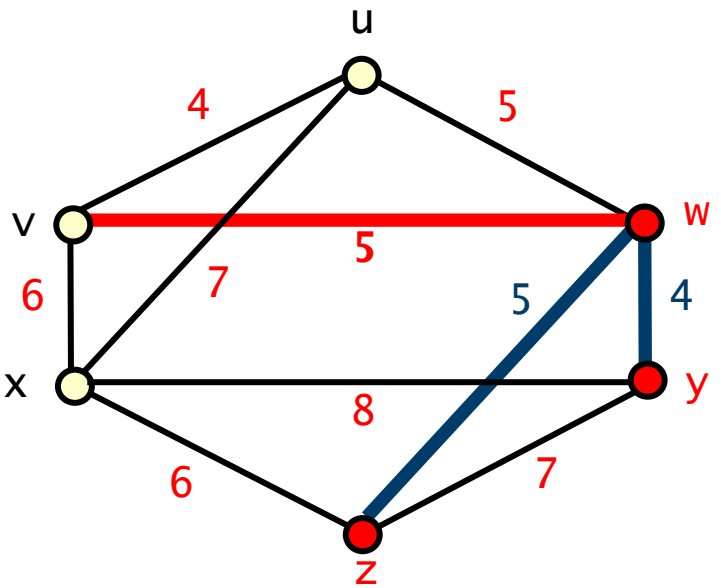
q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: w, y, z
non-tree vertices: u, v, x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



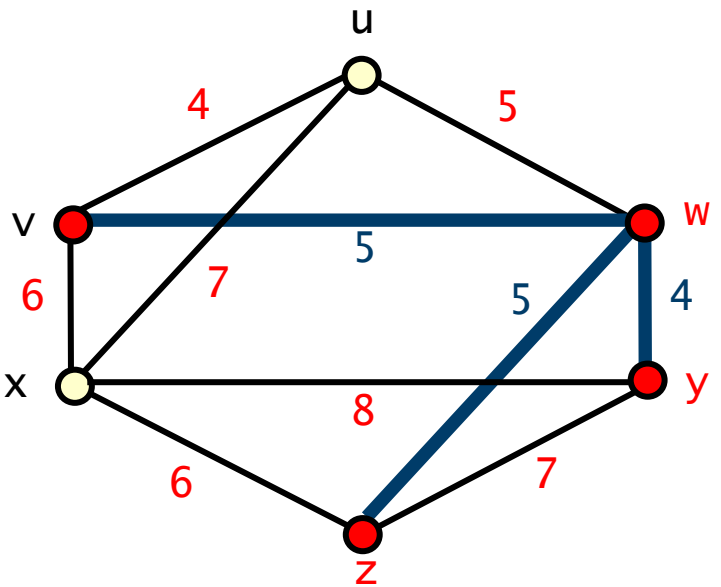
q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: w, y, z
non-tree vertices: u, v, x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



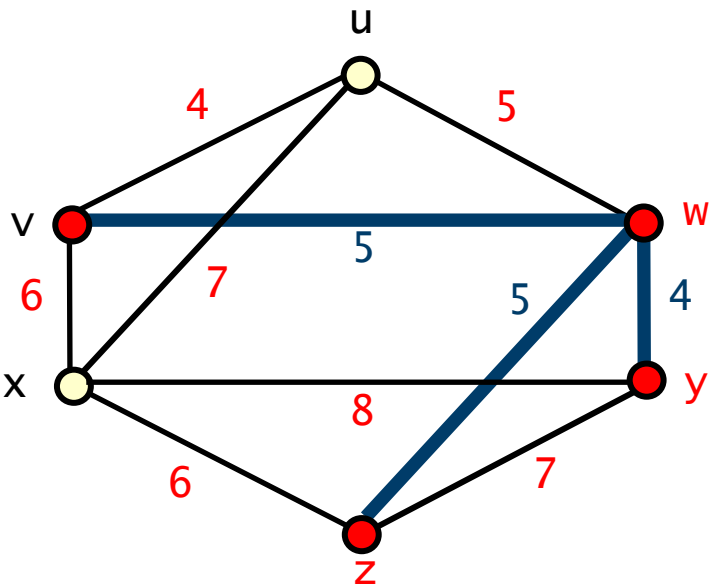
q	q.bestTV	wt({q.bestTV,q})
u	w	5
v	w	5
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: v,w,y,z
non-tree vertices: u,x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	w→v	5→4
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

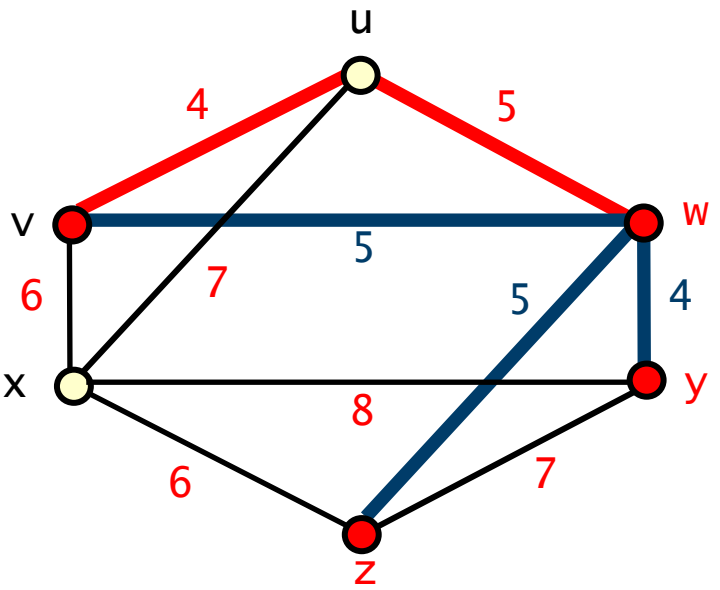
tree vertices: v,w,y,z
non-tree vertices: u,x

check bestTV against new tv vertex v

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	**w→v**	**5→4**
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

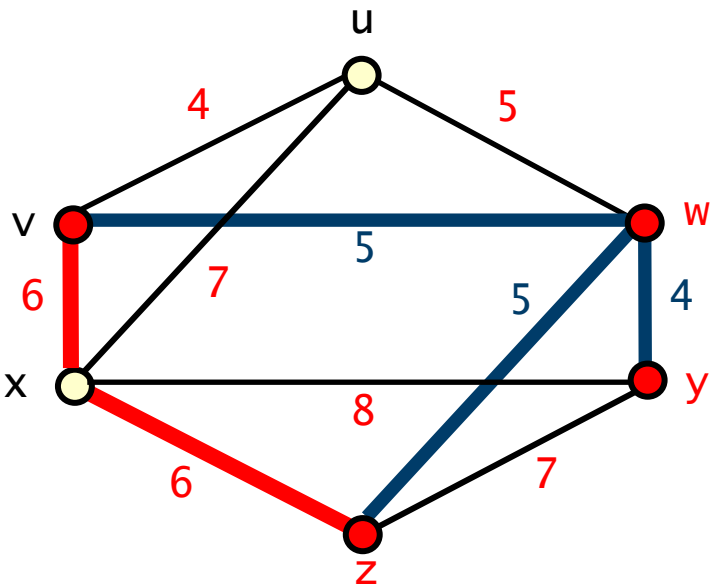
tree vertices: v,w,y,z
non-tree vertices: u,x

check bestTV against new tv vertex v

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	w→v	5→4
v	-	-
w	-	-
x	**z**	**6**
y	-	-
z	-	-

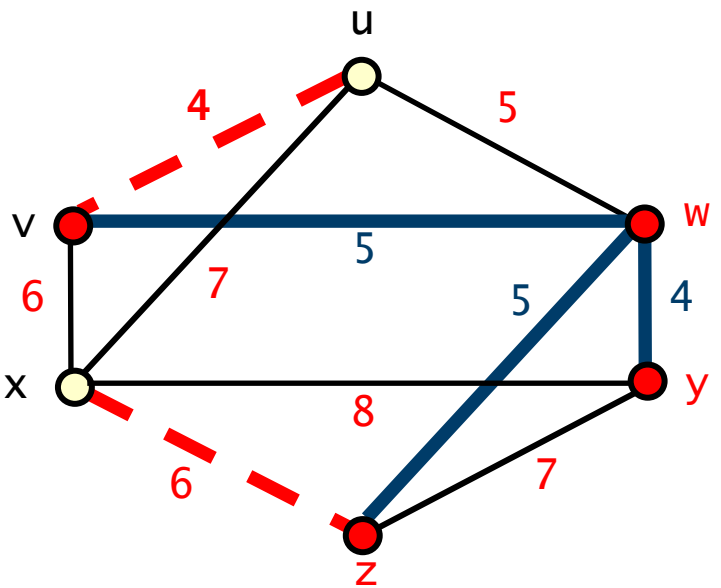
tree vertices: v,w,y,z
non-tree vertices: u,x

check bestTV against new tv vertex v

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



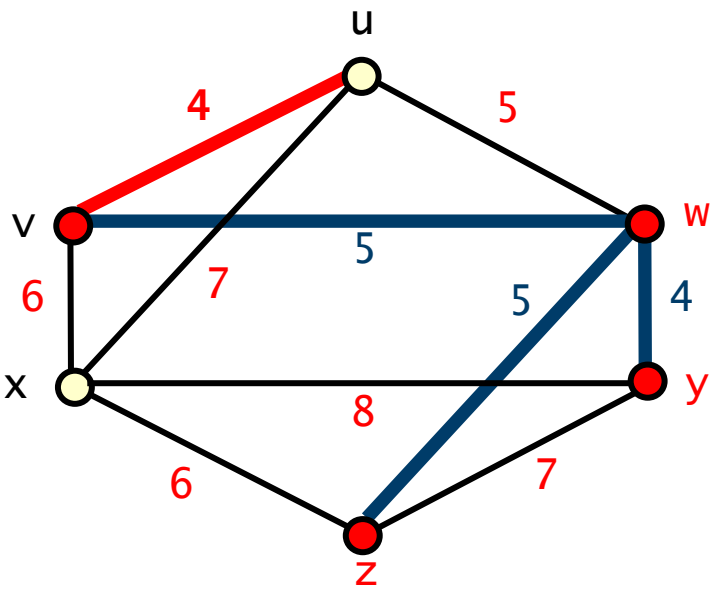
q	q.bestTV	wt({q.bestTV,q})
u	v	4
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: v,w,y,z
non-tree vertices: u,x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



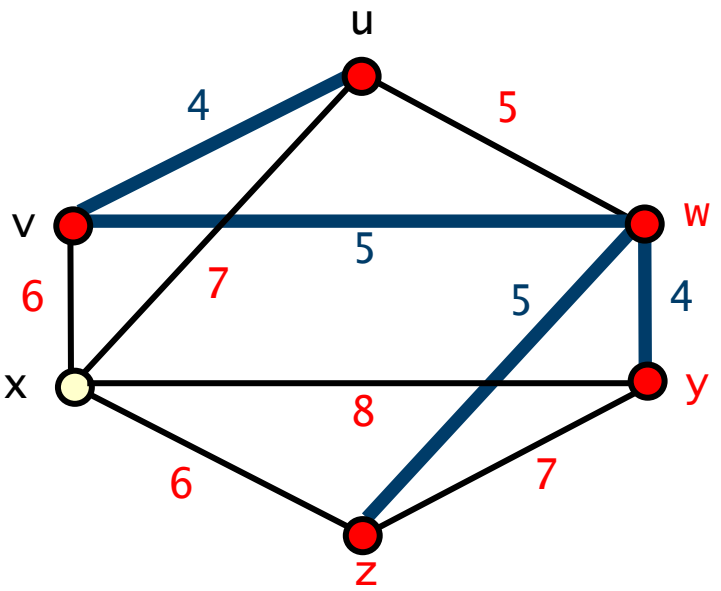
q	q.bestTV	wt({q.bestTV,q})
u	v	4
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: v,w,y,z
non-tree vertices: u,x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



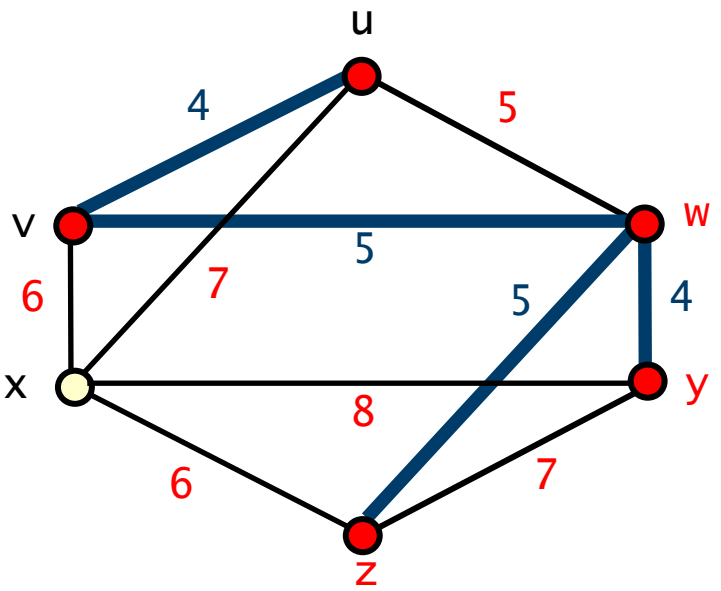
q	q.bestTV	wt({q.bestTV,q})
u	v	4
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: u,v,w,y,z
non-tree vertices: x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



q	q.bestTV	wt({q.bestTV,q})
u	-	-
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

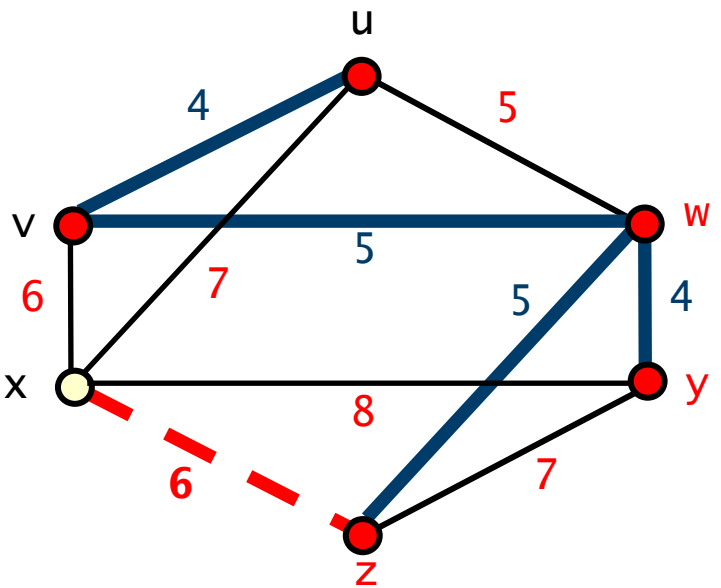
tree vertices: u,v,w,y,z
non-tree vertices: x

check bestTV against new tv vertex u

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



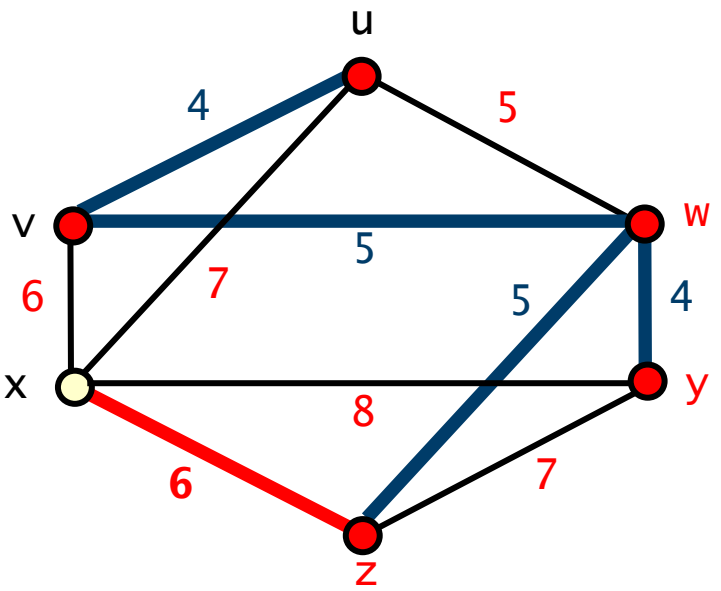
q	q.bestTV	wt({q.bestTV,q})
u	-	-
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: u, v, w, y, z
non-tree vertices: x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s



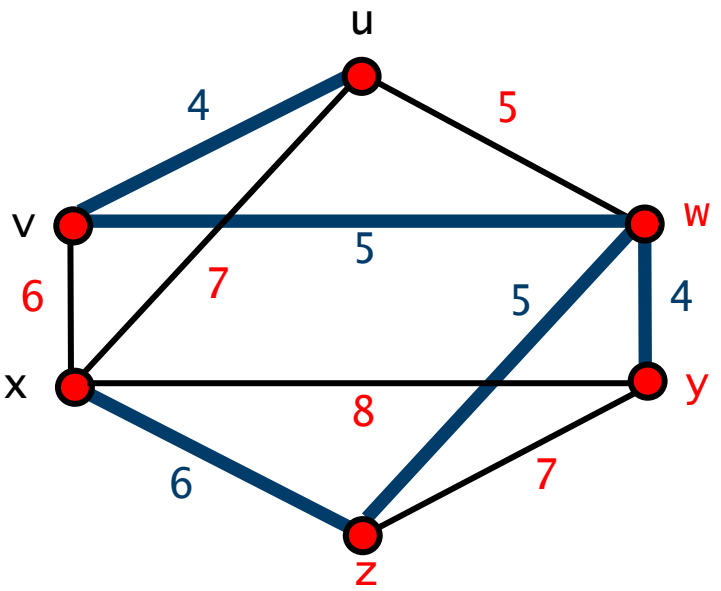
q	q.bestTV	wt({q.bestTV,q})
u	-	-
v	-	-
w	-	-
x	z	6
y	-	-
z	-	-

tree vertices: u, v, w, y, z
non-tree vertices: x

Dijkstra's refinement – Example

Weighted graph G

- choose **ntv** q for which **wt({q, q.bestTV})** is minimal and make q a **tv**
- update s.bestTV for all **ntv** s

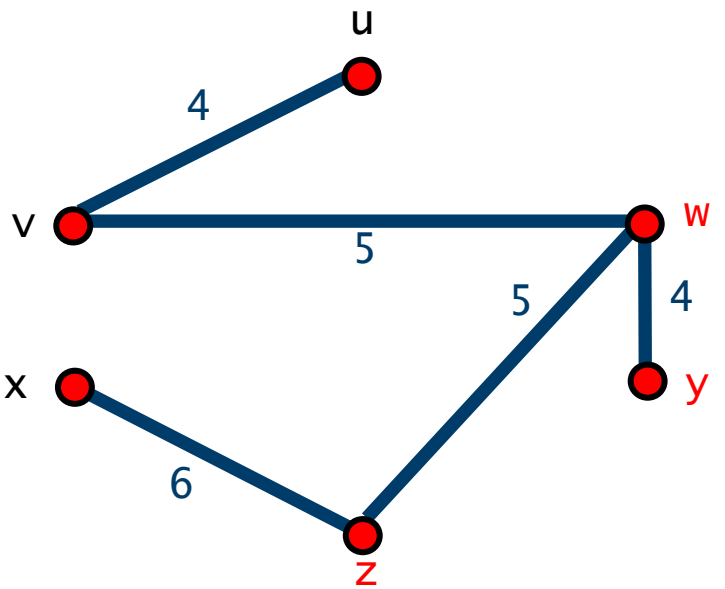


q	q.bestTV	wt({q.bestTV,q})
u	-	-
v	-	-
w	-	-
x	-	-
y	-	-
z	-	-

tree vertices: u,v,w,x,y,z
non-tree vertices:

Dijkstra's refinement – Example

Weighted graph G



Minimum spanning tree for G

- weight 24

q	q.bestTV	wt({q.bestTV,q})
u	-	-
v	-	-
w	-	-
x	-	-
y	-	-
z	-	-