Forelesning 12

Bonusmateriale

Ting som ikke ble med i forelesningen, men som kanskje kan være av interesse

Dette er ment å gi en mer konkret og detaljert forståelse av hvordan Edmonds-Karp fungerer, og hvordan den kan implementeres, men dette er kun et supplement for økt forståelse. Du trenger ikke «pugge» denne detaljerte varianten, eller huske akkurat hvordan den fungerer, så lenge du har skjønt og husker det som står i pensum.

Mer detaljert Edmonds–Karp

Dere trenger ikke kunne denne i detalj!

$v.f$

Potensial (potensiell flyt)

Hvor mye mer flyt får vi til å sende fra s til v?

mlh@ntnu.no

$$v.f$$

Potensial (potensiell flyt)

Bare et potensial, siden vi kanskje ikke klarer sende så mye videre

$v.f$

Potensial (potensiell flyt)

Gitt av flaskehalsen $c_f(s \rightsquigarrow v)$ for en eller annen sti $s \rightsquigarrow v$

$$v.f$$

Potensial (potensiell flyt)

Etter traversering er $t.f = c_f(p)$ for en forøkende sti p (eller 0)

$$v.f$$

Potensial (potensiell flyt)

Implementasjonsdetalj fra original-algoritmen; diskuteres ikke i boka

$v.f$

Potensial (potensiell flyt)

Ford og Fulkerson sin «merkelapp»: $\langle v.\pi, \pm, v.f \rangle$

mlh@ntnu.no

$$v.f$$

Potensial (potensiell flyt)

Vi lagrer ikke fortegn/retning ($\pm$) eksplisitt; bruker bare $\langle v.\pi, v.f \rangle$

FORD-FULKERSON(G, s, t)

```
1  for each edge (u, v) ∈ G.E
2      (u, v).f = 0
3  while there is a path p from s to t in G_f
4      c_f(p) = min {c_f(u, v) : (u, v) is in p}
5      for each edge (u, v) in p
6          if (u, v) ∈ E
7              (u, v).f = (u, v).f + c_f(p)
8          else (v, u).f = (v, u).f − c_f(p)
```

FORD-FULKERSON′(G, s, t)

```
1  for each edge (u, v) ∈ G.E
2      (u, v).f = 0
3  while LABELING(G, s, t)
4      c_f(p) = t.f
5      for u, v = t.π, t; u ≠ NIL; u, v = u.π, u
6          if (u, v) ∈ G.E
7              (u, v).f = (u, v).f + c_f(p)
8          else (v, u).f = (v, u).f − c_f(p)
```

```
FORD-FULKERSON'(G, s, t)
1  for each edge (u, v) ∈ G.E
2      (u, v).f = 0
3  while LABELING(G, s, t)
4      c_f(p) = t.f
5      for u, v = t.π, t; u ≠ NIL; u, v = u.π, u
6          if (u, v) ∈ G.E
7              (u, v).f = (u, v).f + c_f(p)
8          else (v, u).f = (v, u).f − c_f(p)
```

```
FORD-FULKERSON''(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

FORD-FULKERSON$''$(G, s, t)

```
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

EDMONDS-KARP(G, s, t)

```
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

EDMONDS-KARP(G, s, t)

G	flytnett
s	kilde
t	sluk

Finn forøkende stier med BFS

mlh@ntnu.no

EDMONDS-KARP(G, s, t)

1 **for** each edge $(u, v) \in$ G.E

G	flytnett
s	kilde
t	sluk
u	node
v	node

Initialiser flyten

mlh@ntnu.no

EDMONDS-KARP(G, s, t)

```
1  for each edge (u, v) ∈ G.E
2        (u, v).f = 0
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt

Initialiser flyten

mlh@ntnu.no

EDMONDS-KARP(G, s, t)

1 **for** each edge $(u, v) \in$ G.E
2 $\quad (u, v).f = 0$
3 **while** BFS-LABELING(G, s, t)

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt

Gjenta så lenge vi finner en forøkende sti . . .

mlh@ntnu.no

EDMONDS-KARP(G, s, t)

```
1  for each edge (u, v) ∈ G.E
2        (u, v).f = 0
3  while BFS-LABELING(G, s, t)
4        c_f(p) = t.f
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals

Finn forøkende sti, med flaskehals-kapasitet $c_f(p)$

mlh@ntnu.no

EDMONDS-KARP(G, s, t)

```
1  for each edge (u, v) ∈ G.E
2        (u, v).f = 0
3  while BFS-LABELING(G, s, t)
4        c_f(p) = t.f
5        u, v = t.π, t
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals
$v.\pi$	forgjenger

Starter i t og ser på kanten $(t.\pi, t)$

mlh@ntnu.no

EDMONDS-KARP(G, s, t)

1 **for** each edge $(u, v) \in$ G.E
2 $\quad (u, v).f = 0$
3 **while** BFS-LABELING(G, s, t)
4 $\quad c_f(p) = t.f$
5 $\quad u, v = t.\pi, t$
6 $\quad$ **while** $u \neq$ NIL

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals
$v.\pi$	forgjenger

Så lenge vi ikke har kommet til kilden . . .

mlh@ntnu.no

```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2        (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4        c_f(p) = t.f
 5        u, v = t.π, t
 6        while u ≠ NIL
 7              if (u, v) ∈ G.E
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals
$v.\pi$	forgjenger

Er det en fremoverkant?

mlh@ntnu.no

```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2        (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4        c_f(p) = t.f
 5        u, v = t.π, t
 6        while u ≠ NIL
 7              if (u, v) ∈ G.E
 8                    (u, v).f = (u, v).f + c_f(p)
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals
$v.\pi$	forgjenger

I så fall, vanlig flytøkning

mlh@ntnu.no

```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals
$v.\pi$	forgjenger

Bakoverkant: Flytoppheving og omdirigering

mlh@ntnu.no

```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

G	flytnett
s	kilde
t	sluk
u	node
v	node
$e.f$	flyt
$c_f(p)$	flaskehals
$v.\pi$	forgjenger

Flytt et hakk bakover langs den forøkende stien

mlh@ntnu.no

BFS-Labeling(G, s, t)

G	flytnett
s	kilde
t	sluk

Spre potensial med BFS

mlh@ntnu.no

BFS-Labeling(G, s, t)

1 **for** each vertex $u \in$ G.V

G	flytnett
s	kilde
t	sluk
u	node

Initialiser noder

mlh@ntnu.no

BFS-Labeling(G, s, t)

1 **for** each vertex $u \in$ G.V
2 $\quad u.f = 0$

G	flytnett
s	kilde
t	sluk
u	node
$v.f$	potensial

Ingen flyt har nådd frem til noen noder ennå

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2        u.f = 0
3        u.π = NIL
```

G	flytnett
s	kilde
t	sluk
u	node
$v.f$	potensial
$v.\pi$	forgjenger

Ingen noder har noen BFS-forgjenger ennå

mlh@ntnu.no

BFS-Labeling(G, s, t)

1 **for** each vertex $u \in$ G.V
2 $\quad u.f = 0$
3 $\quad u.\pi =$ NIL
4 $s.f = \infty$

G	flytnett
s	kilde
t	sluk
u	node
$v.f$	potensial
$v.\pi$	forgjenger

Vi får uendelig flyt frem til startnoden

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2        u.f = 0
3        u.π = NIL
4  s.f = ∞
5  Q = ∅
```

G	flytnett
s	kilde
t	sluk
u	node
$v.f$	potensial
$v.\pi$	forgjenger
Q	kø

Starter med tom kø

mlh@ntnu.no

```
BFS-Labeling(G, s, t)
1  for each vertex u ∈ G.V
2        u.f = 0
3        u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
```

G	flytnett
s	kilde
t	sluk
u	node
$v.f$	potensial
$v.\pi$	forgjenger
Q	kø

Legg kilden inn i køen

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2        u.f = 0
3        u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  ...
```

G	flytnett
s	kilde
t	sluk
u	node
$v.f$	potensial
$v.\pi$	forgjenger
Q	kø

mlh@ntnu.no

BFS-Labeling(G, s, t)

6 ...

7 **while** Q $\neq \emptyset$ and $t.f == 0$

G flytnett
s kilde
t sluk
Q kø

$v.f$ potensial

Til alle er traverserte eller vi når sluket ...

mlh@ntnu.no

BFS-Labeling(G, s, t)

6 ...
7 **while** Q $\neq \emptyset$ and $t.f == 0$
8 u = Dequeue(Q)

G flytnett
s kilde
t sluk
Q kø
u node

$v.f$ potensial

... velg neste node (FIFO)

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
6  ...
7  while Q ≠ ∅ and t.f == 0
8        u = Dequeue(Q)
9        for all edges (u, v), (v, u) ∈ G.E
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
$v.f$	potensial

Se på alle naboer langs både ut- og inn-kanter

mlh@ntnu.no

```
BFS-Labeling(G, s, t)
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
$v.f$	potensial

Fremoverkant?

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial

Restkapasitet

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial

Bakoverkant: Flytoppheving

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial

Ledig kapasitet og ubesøkt node?

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial

Send mest mulig av u sitt potensial videre til v

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial
$v.\pi$	forgjenger

Noter hvor potensialet kom fra

mlh@ntnu.no

BFS-Labeling(G, s, t)

```
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
16                    Enqueue(Q, v)
```

G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial
$v.\pi$	forgjenger

Husk å besøke v senere

mlh@ntnu.no

```
BFS-Labeling(G, s, t)
 6  ...
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
16                    Enqueue(Q, v)
17  return t.f ≠ 0
```

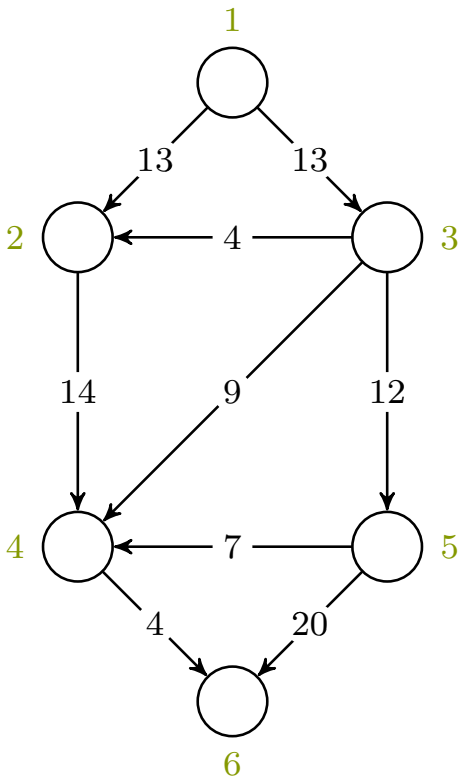
G	flytnett
s	kilde
t	sluk
Q	kø
u	node
v	nabo
c	kapasitet
c_f	restkapasitet
$e.f$	flyt
$v.f$	potensial
$v.\pi$	forgjenger

Fant vi noen forøkende sti? Dvs., fikk vi noe flyt frem til t?

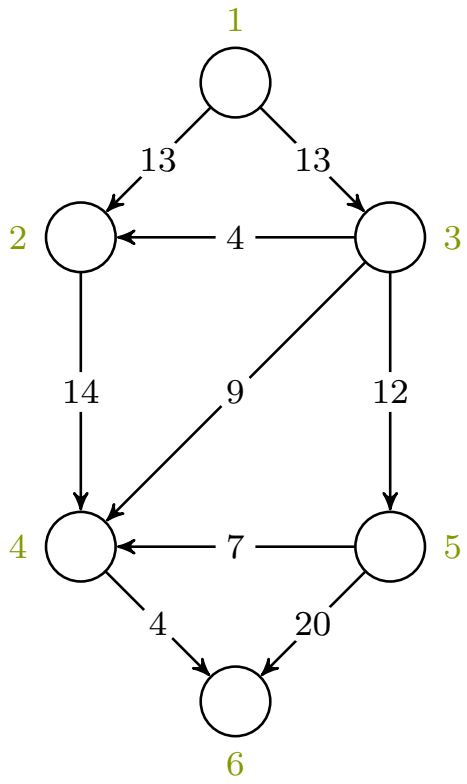
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$

Flyt



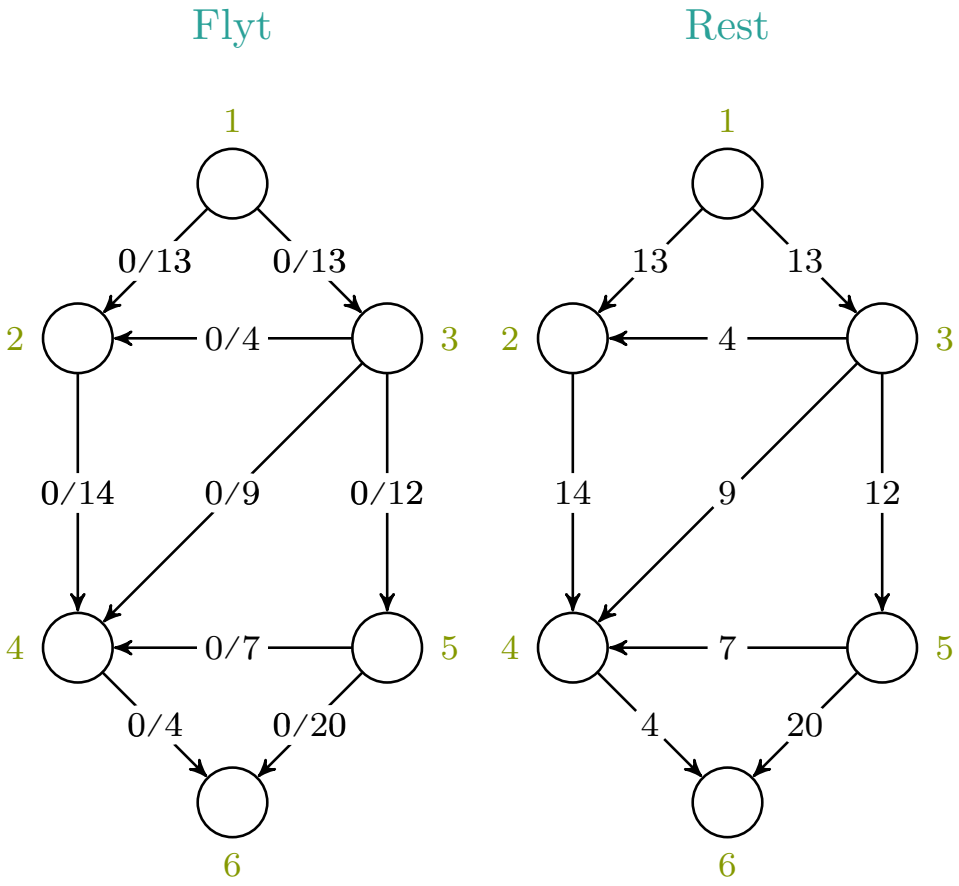
Rest



Node 1 er kilde, node 6 er sluk.

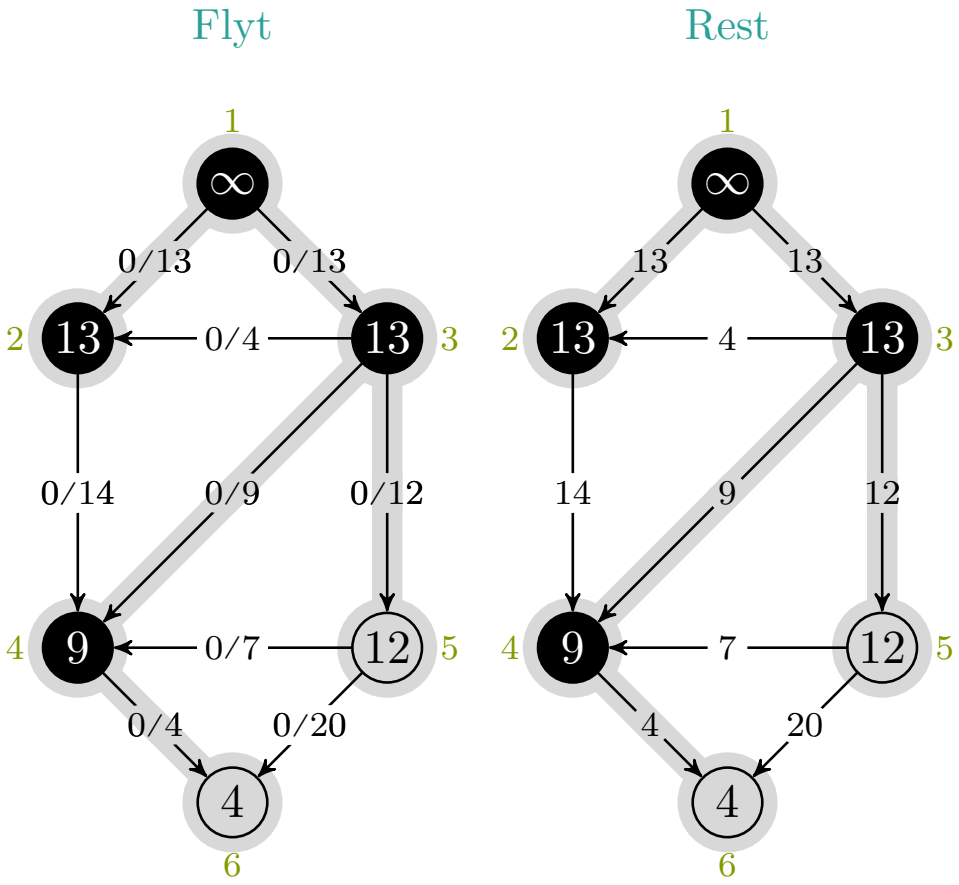
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$



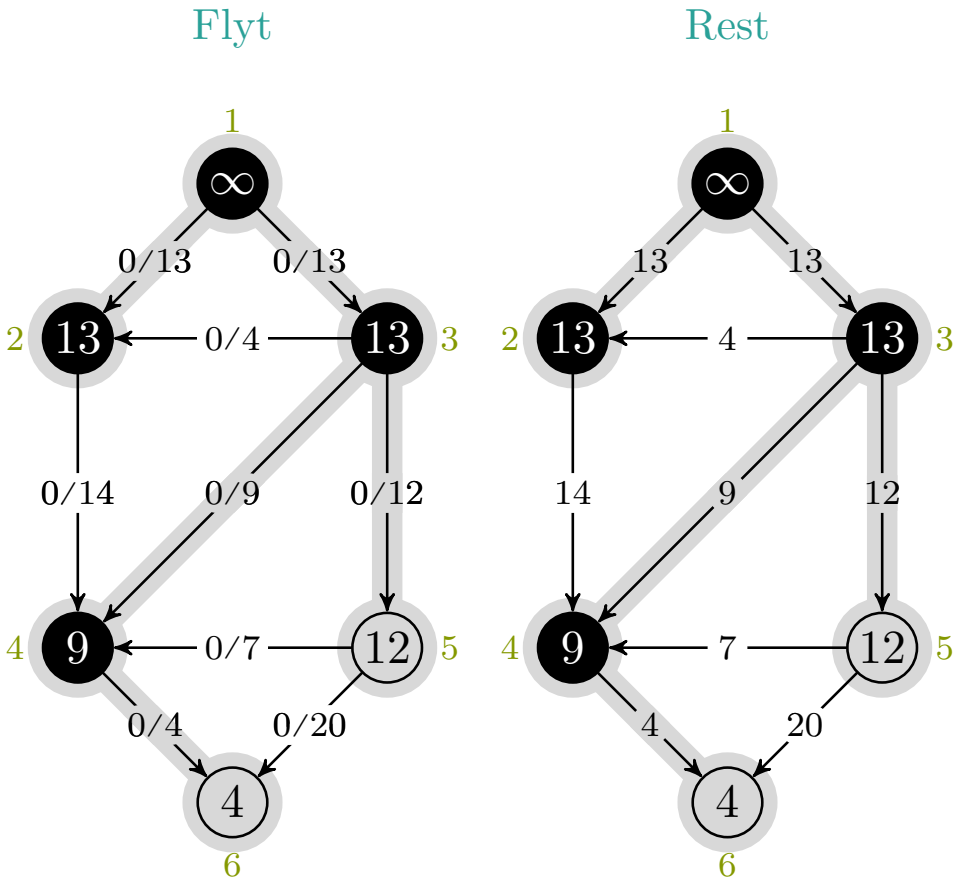
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$



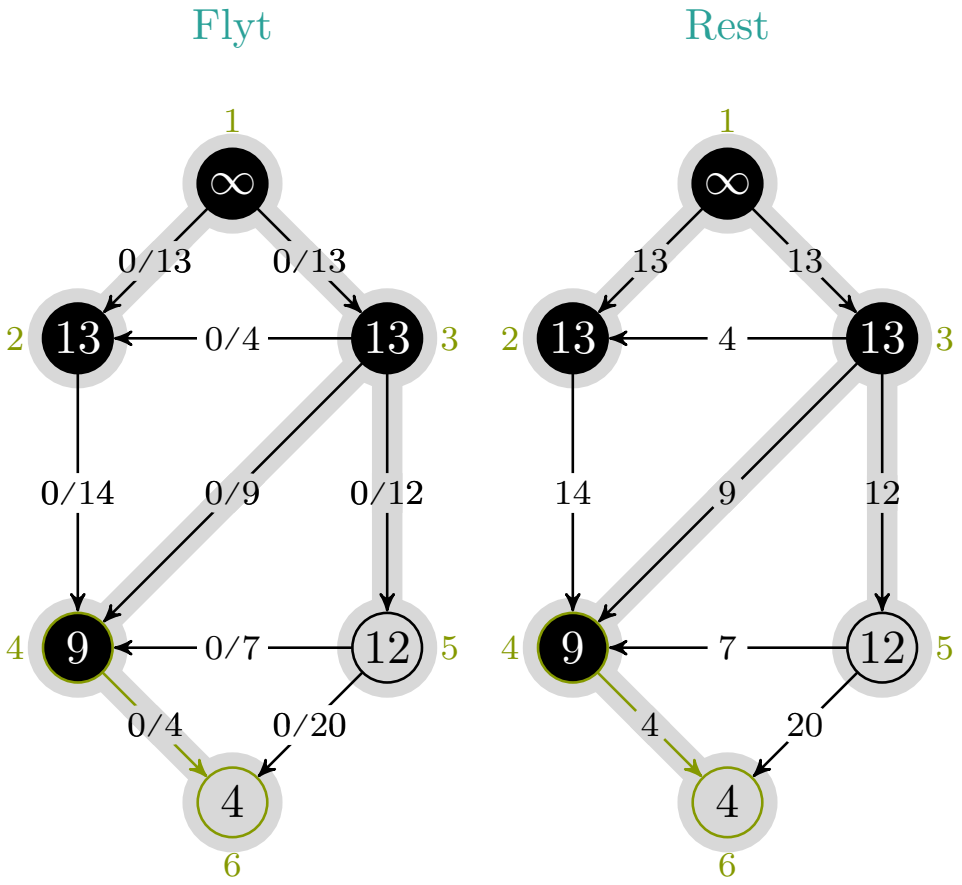
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, -, -$



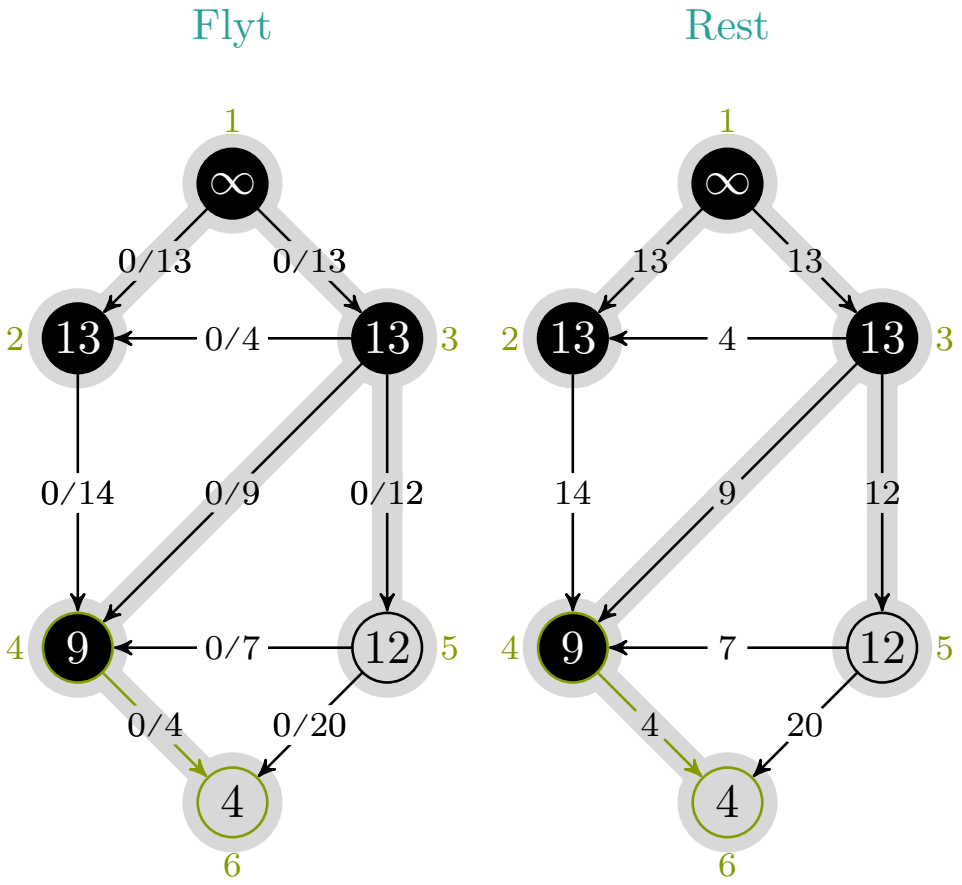
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 4, 6$



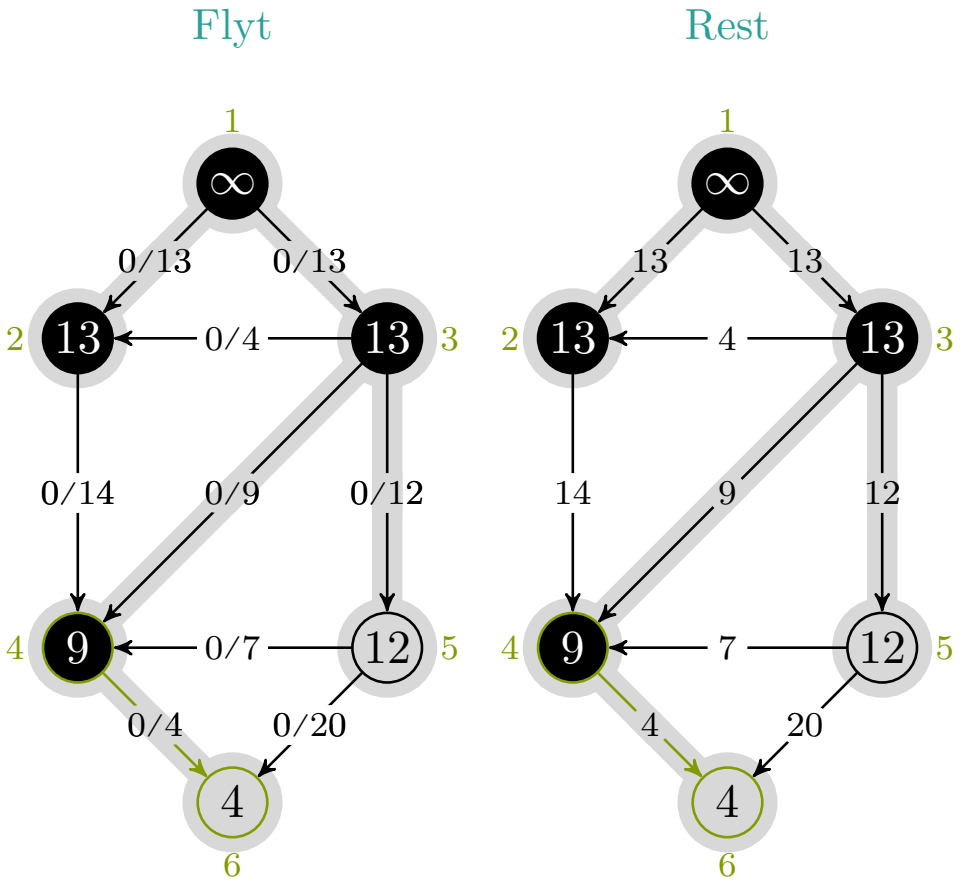
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 4, 6$



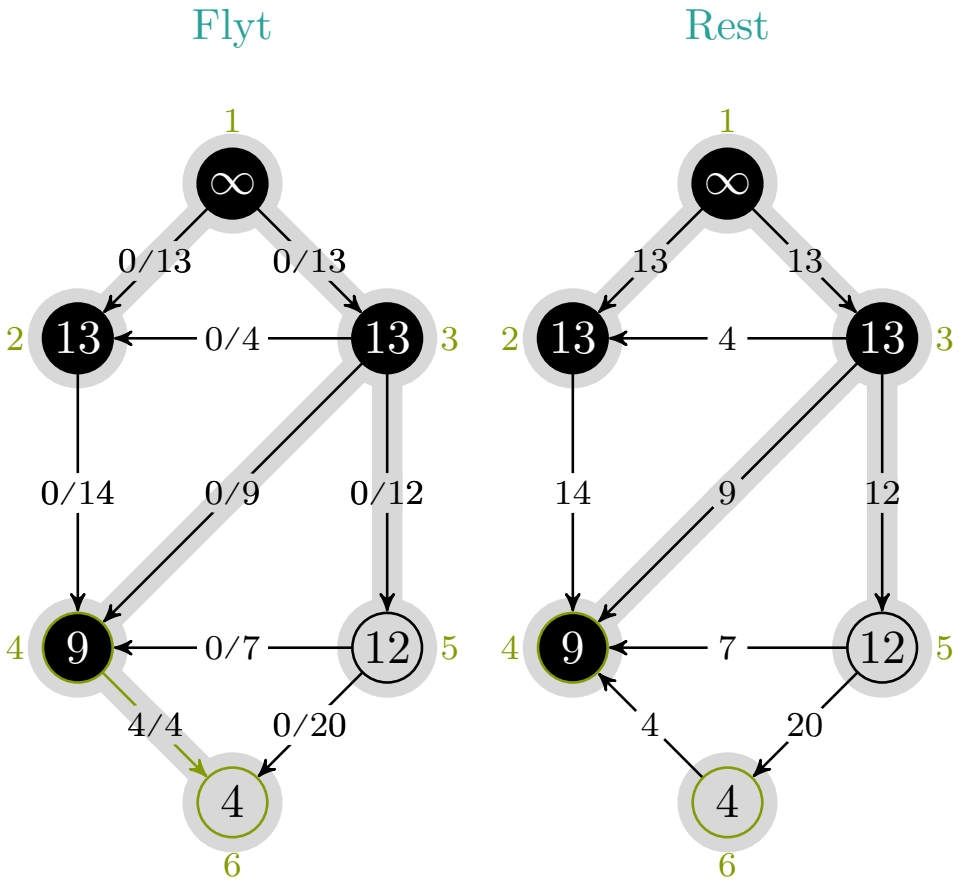
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 4, 6$



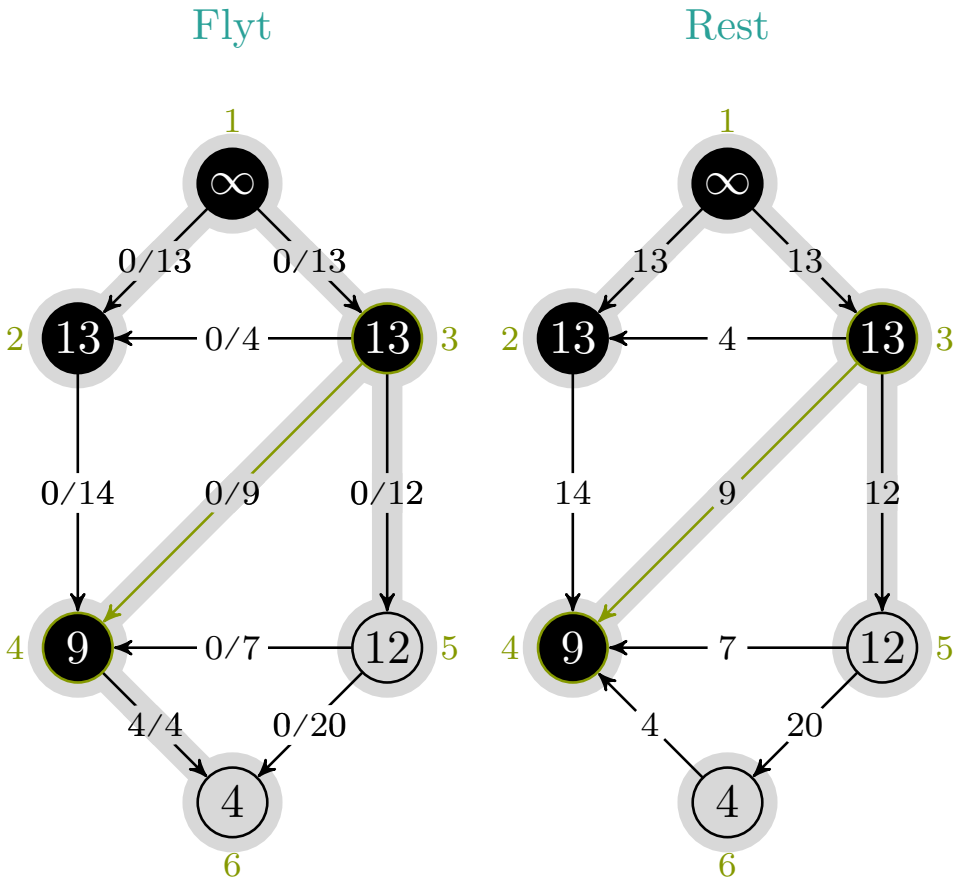
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 4, 6$



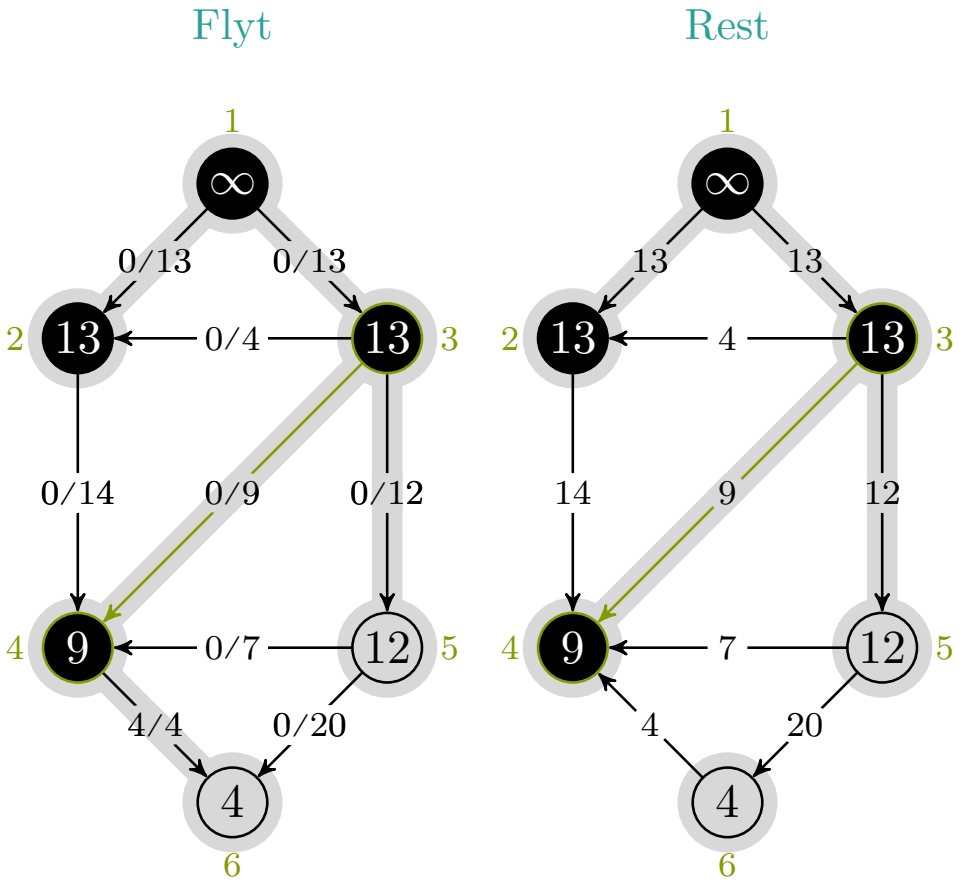
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 3, 4$



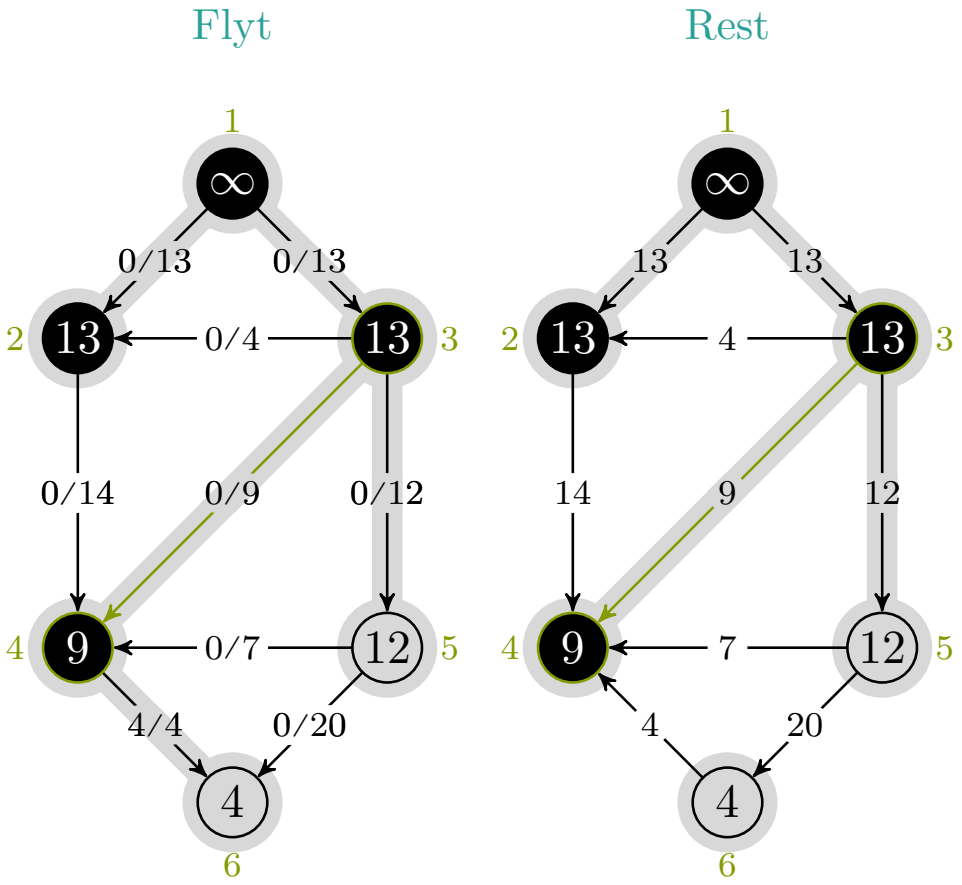
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 3, 4$



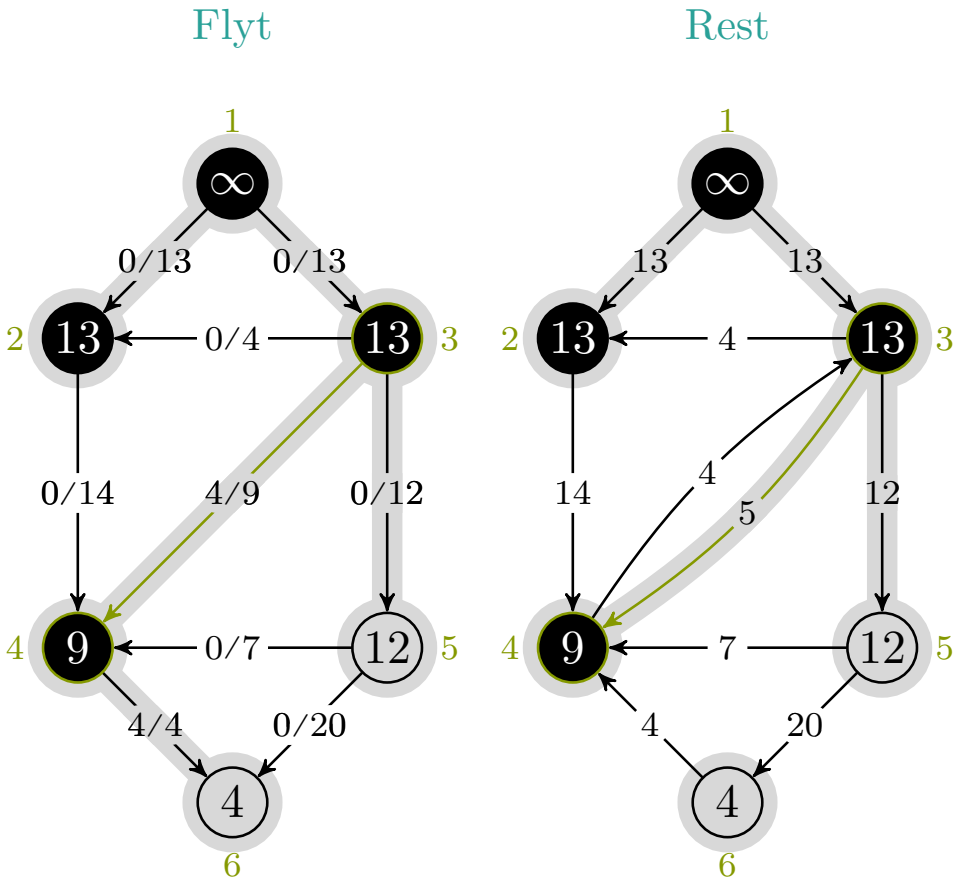
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 3, 4$



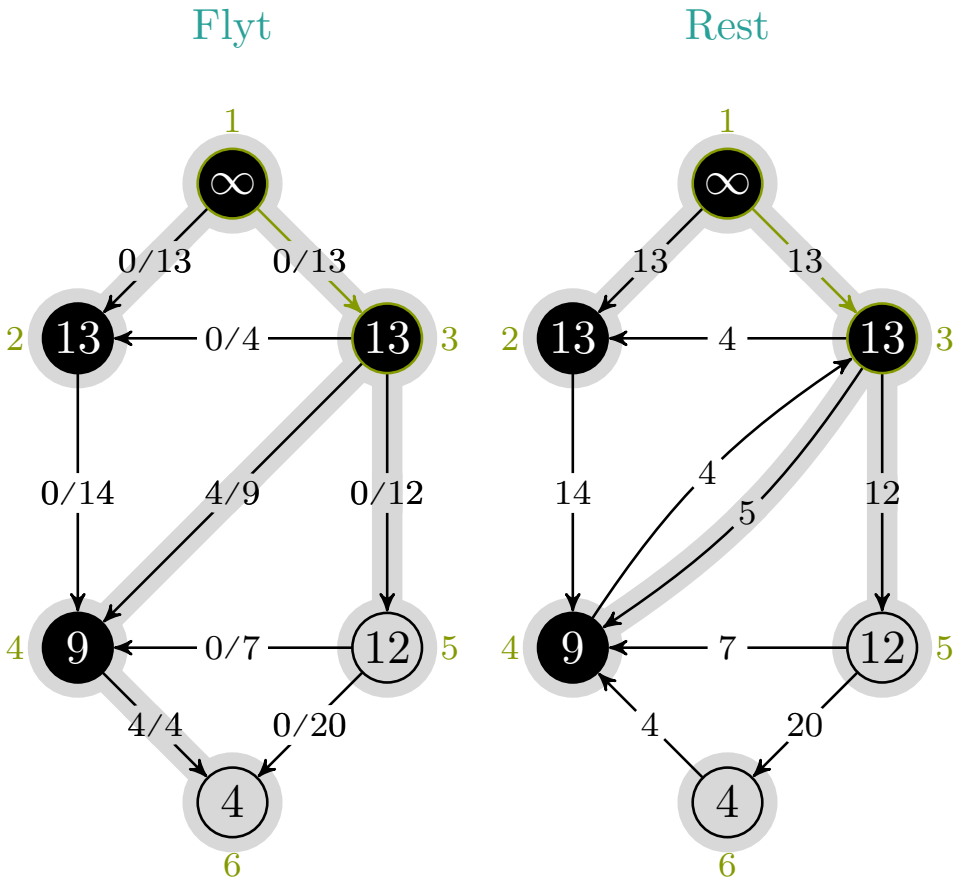
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 3, 4$



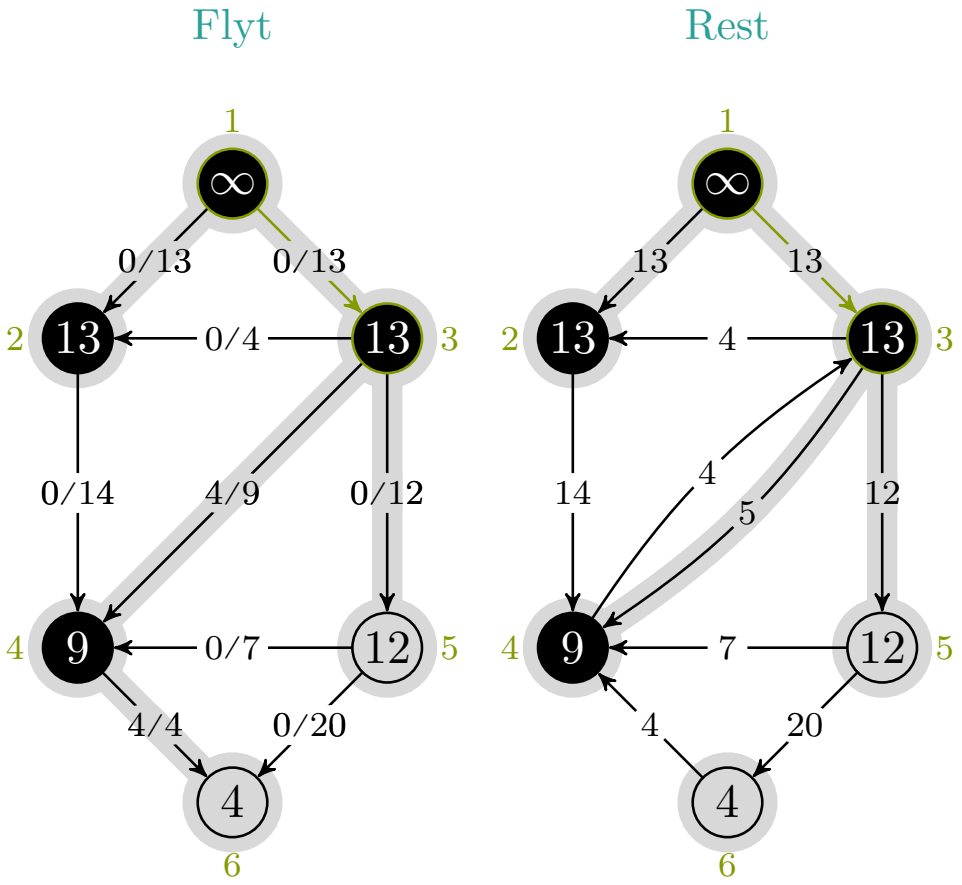
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 1, 3$



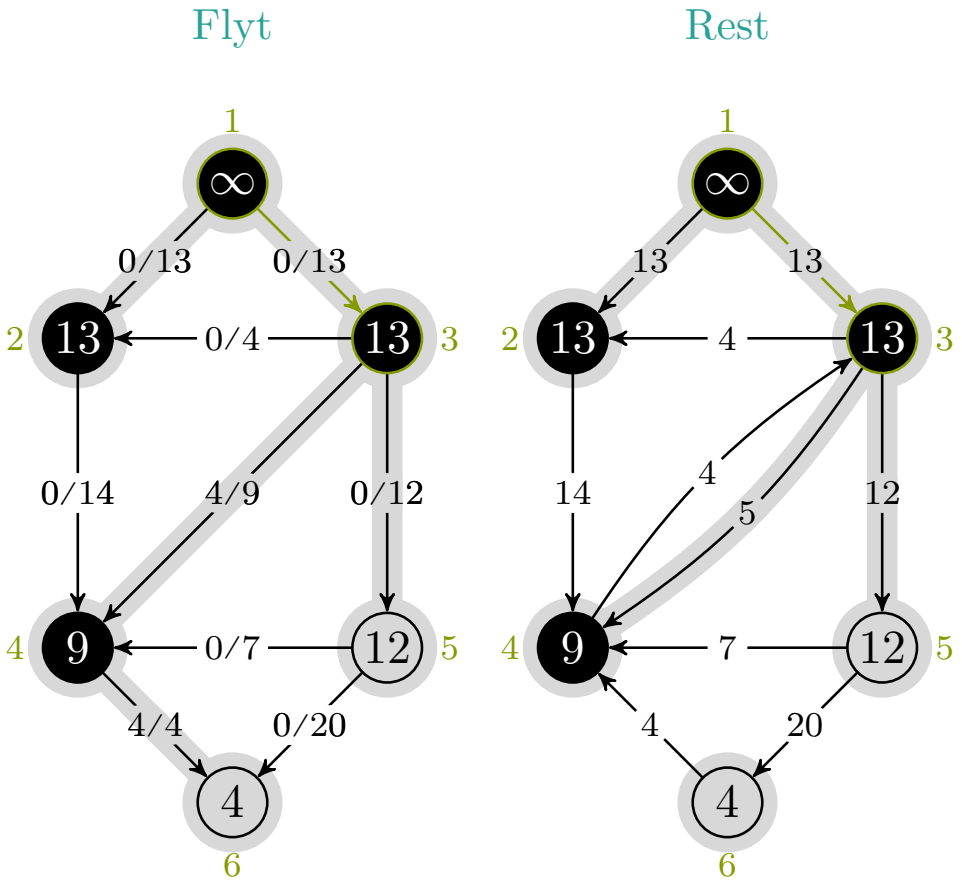
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 1, 3$



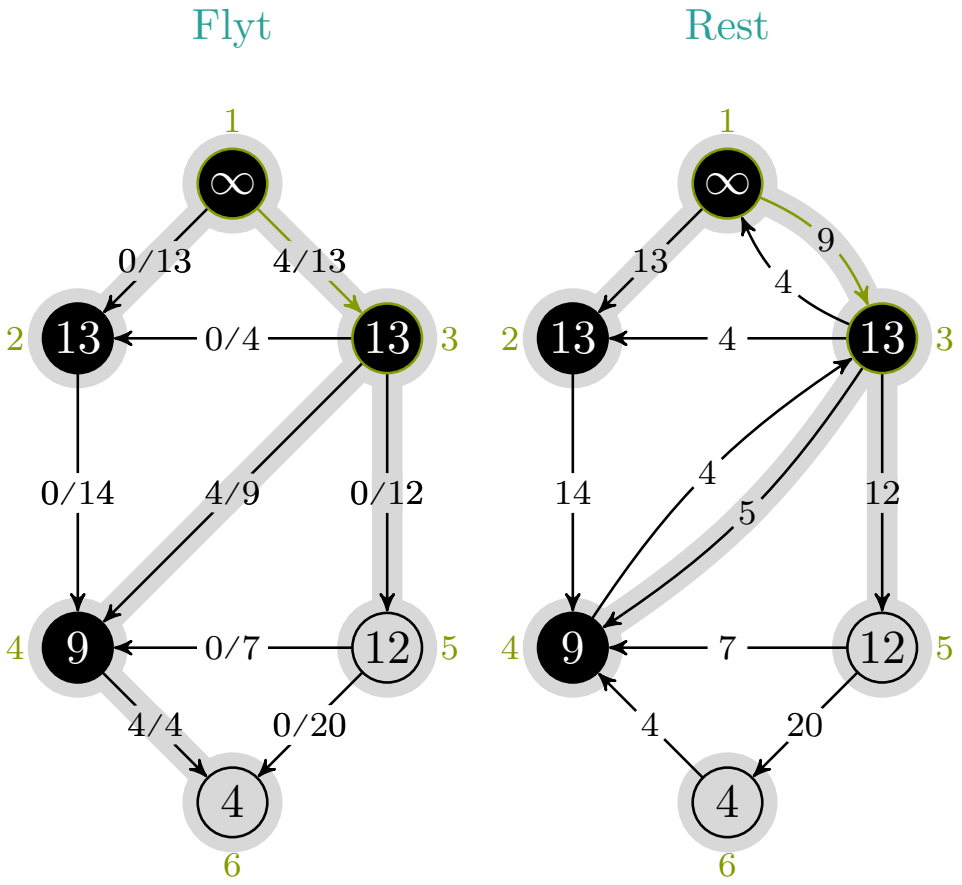
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 1, 3$



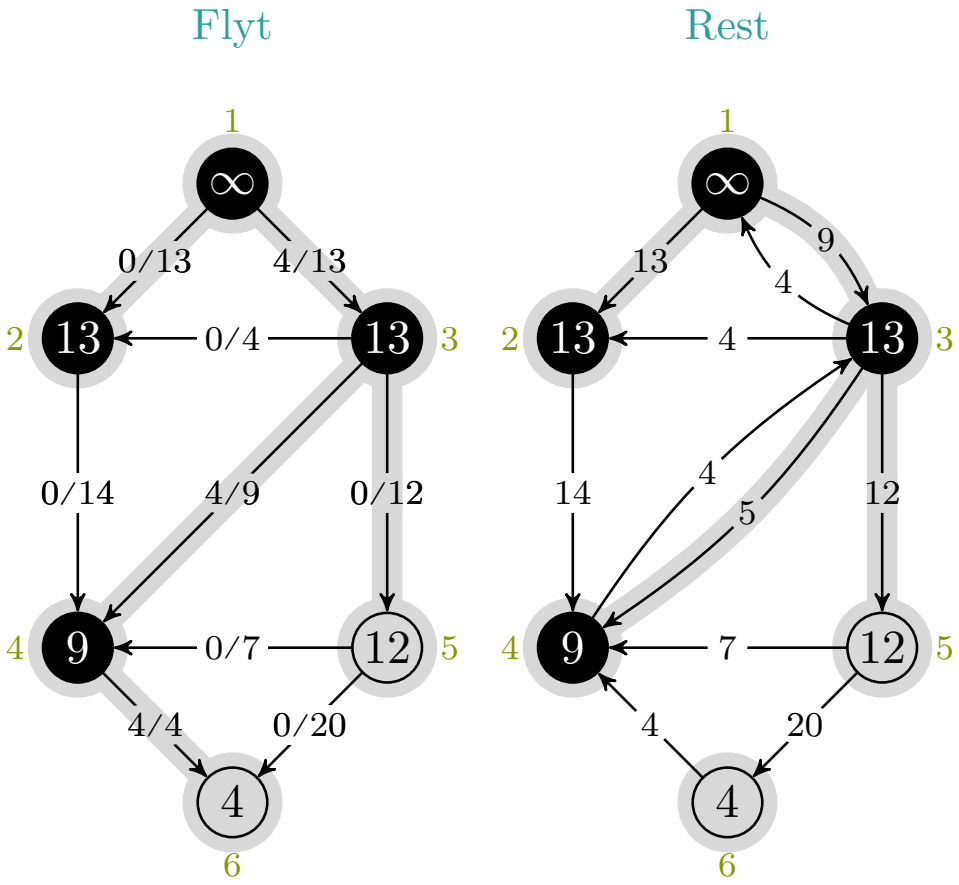
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, 1, 3$



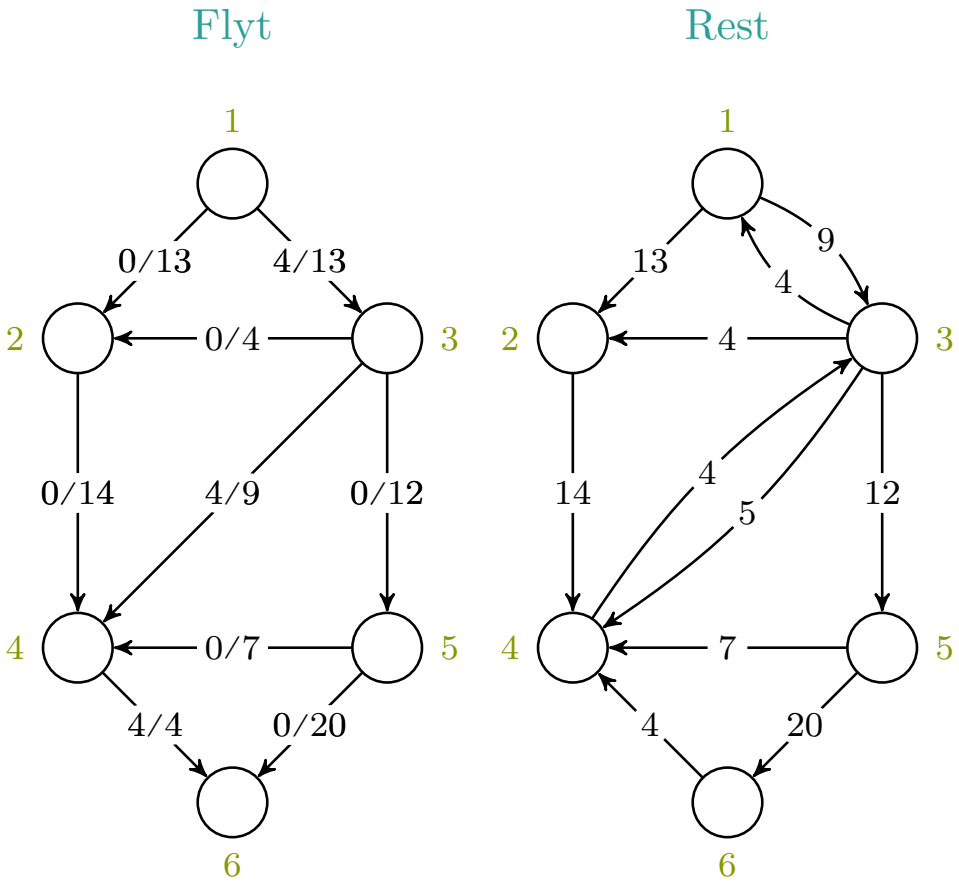
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 4, \text{NIL}, 1$



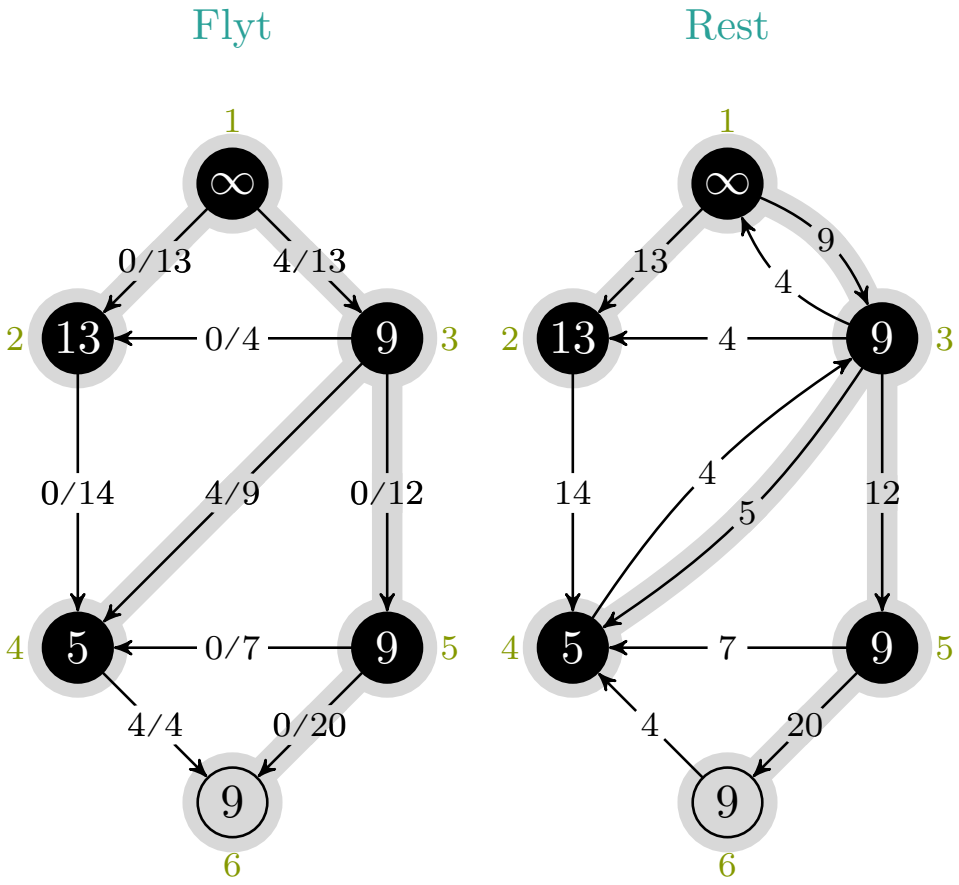
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$



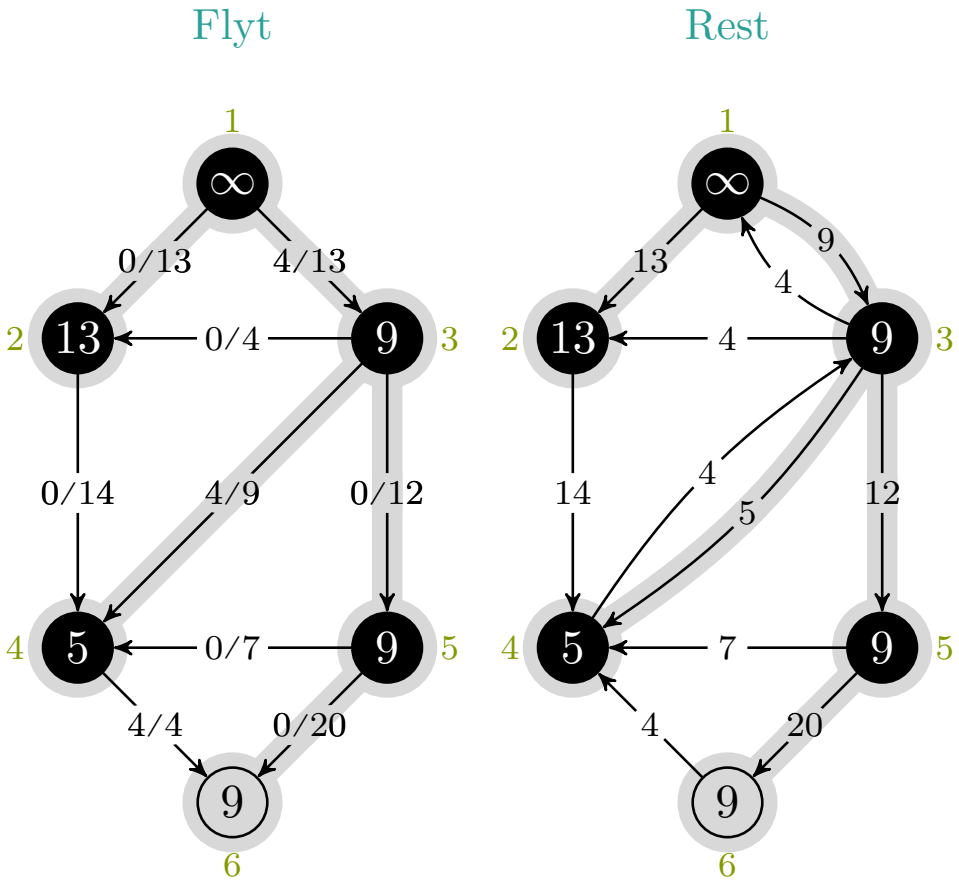
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$



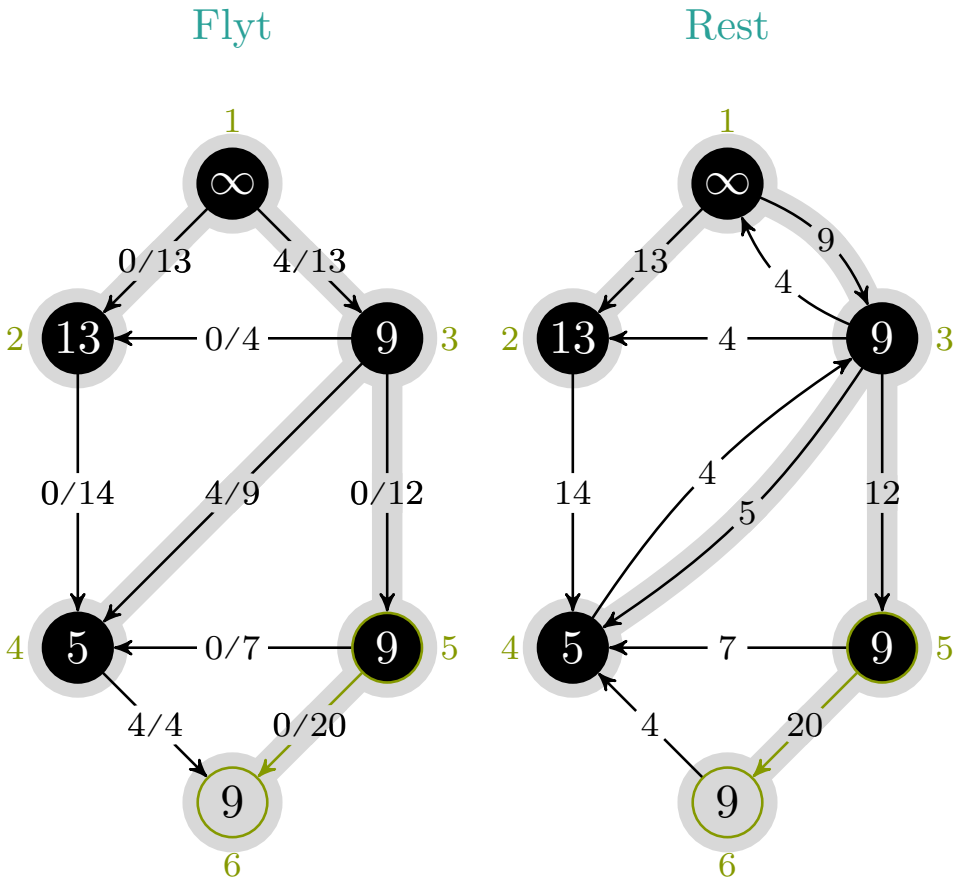
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, -, -$



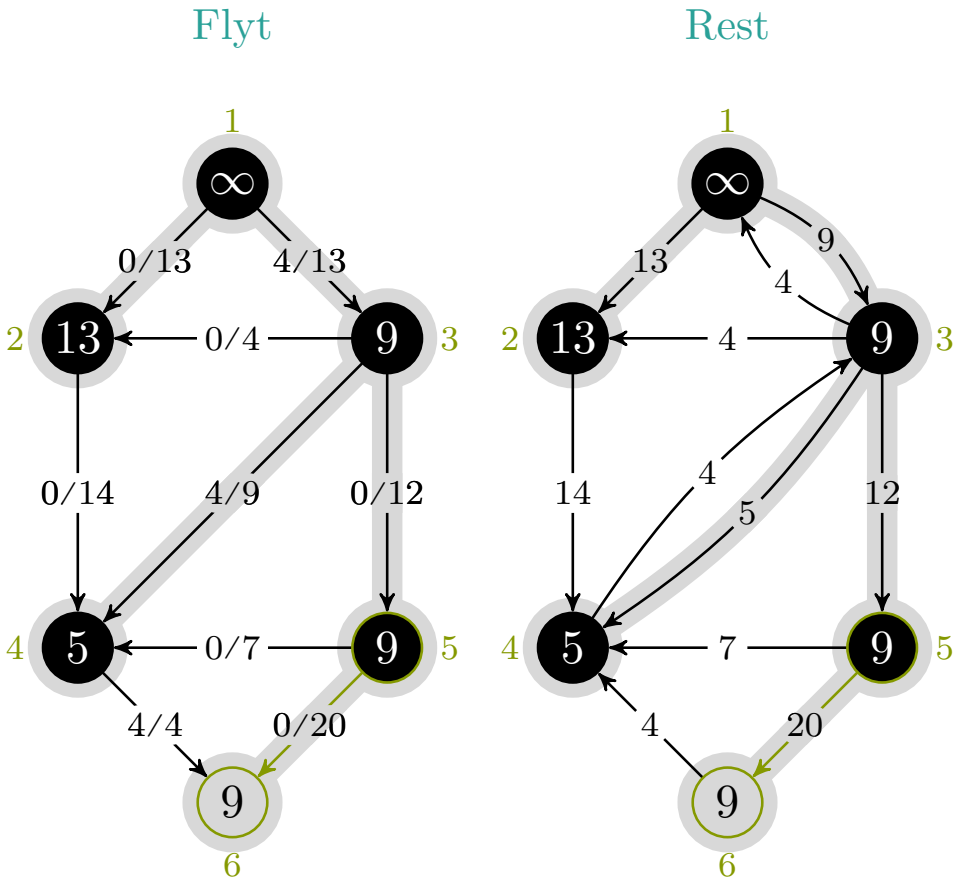
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 5, 6$



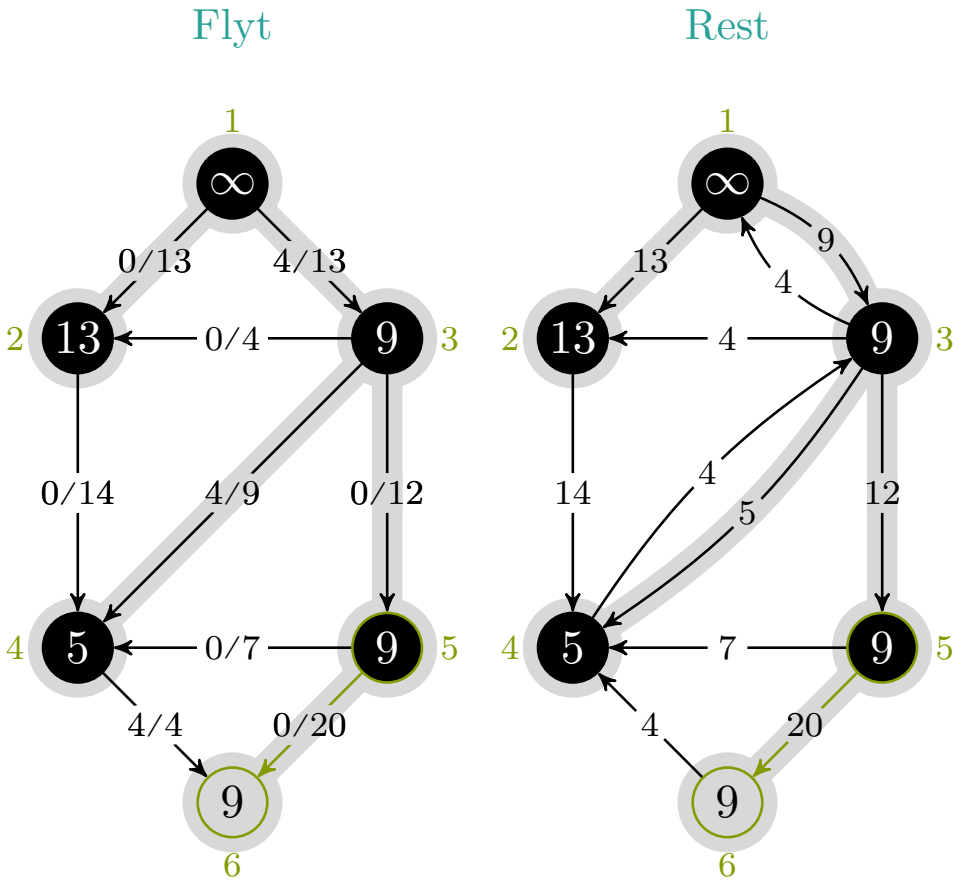
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 5, 6$



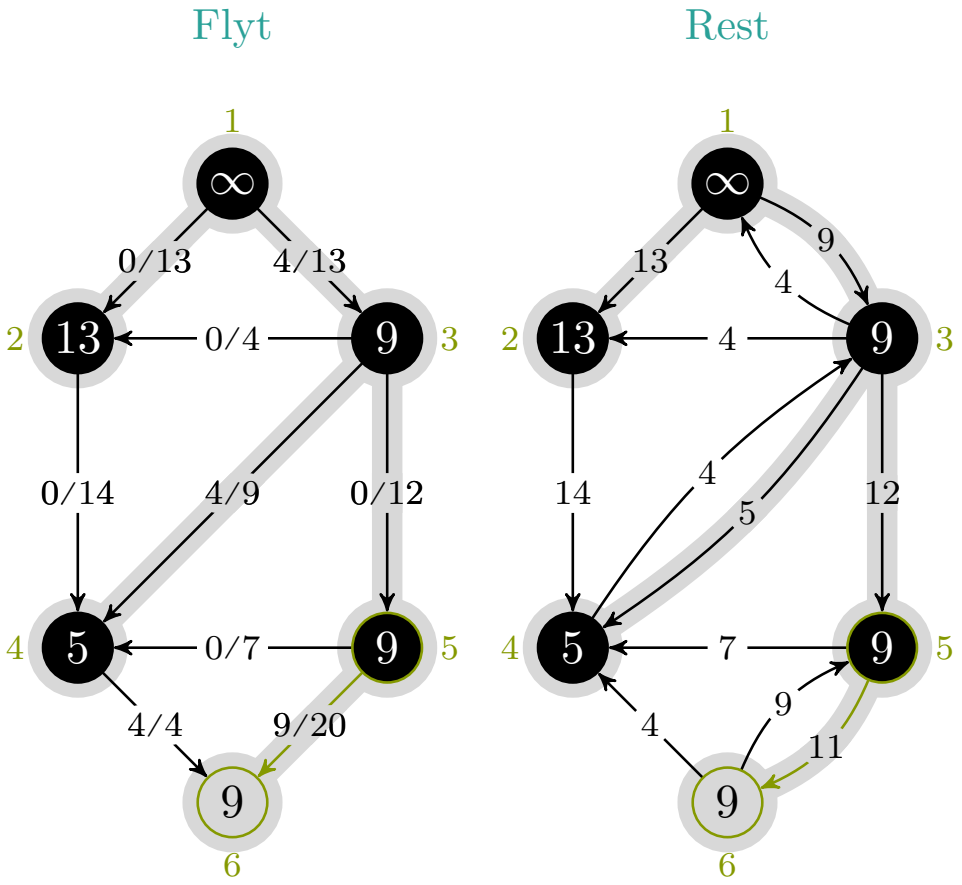
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 5, 6$



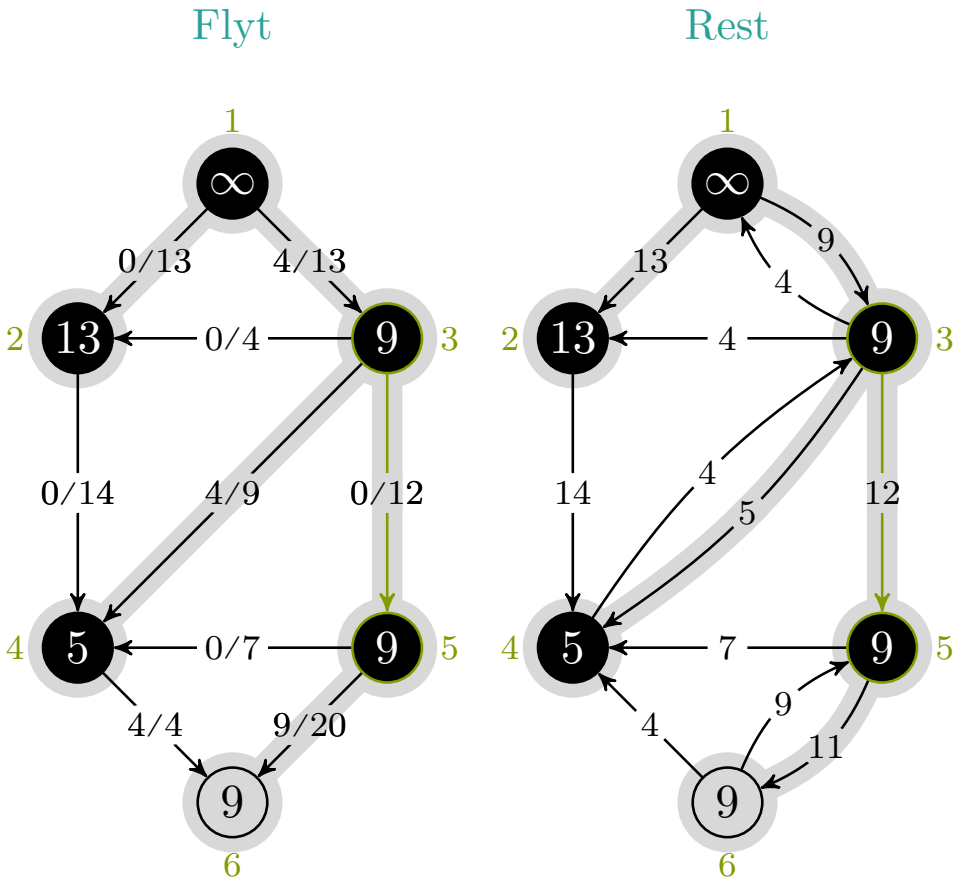
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 5, 6$



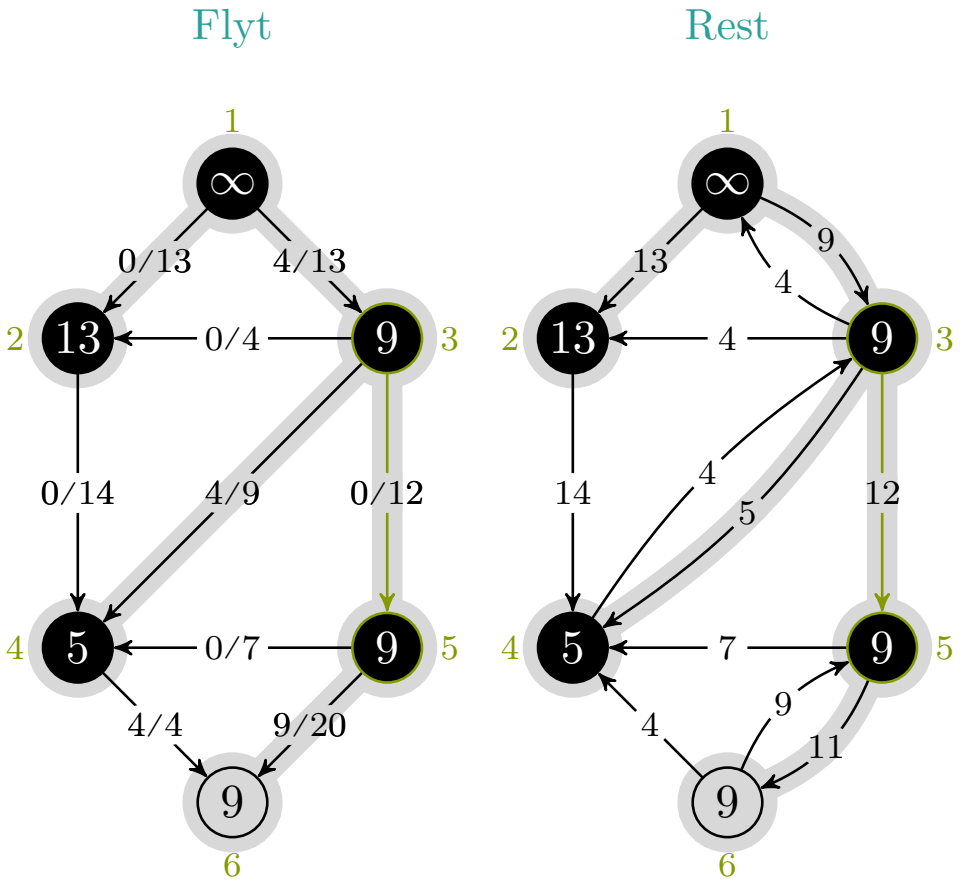
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 3, 5$



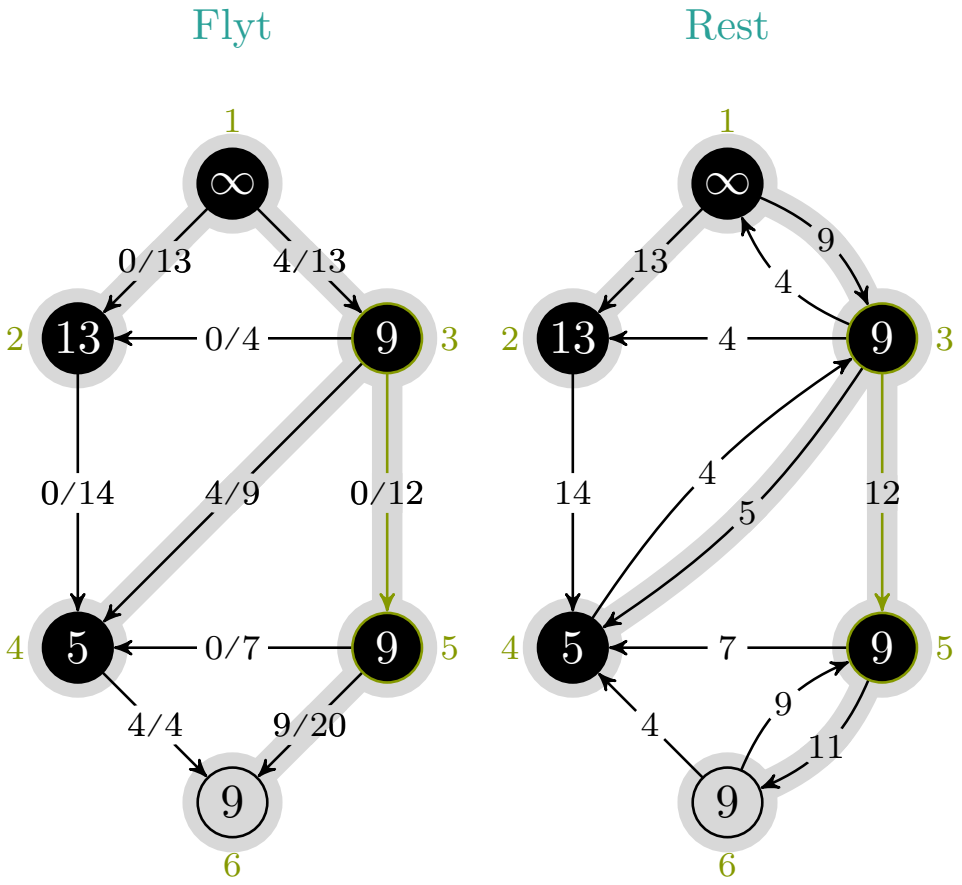
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 3, 5$



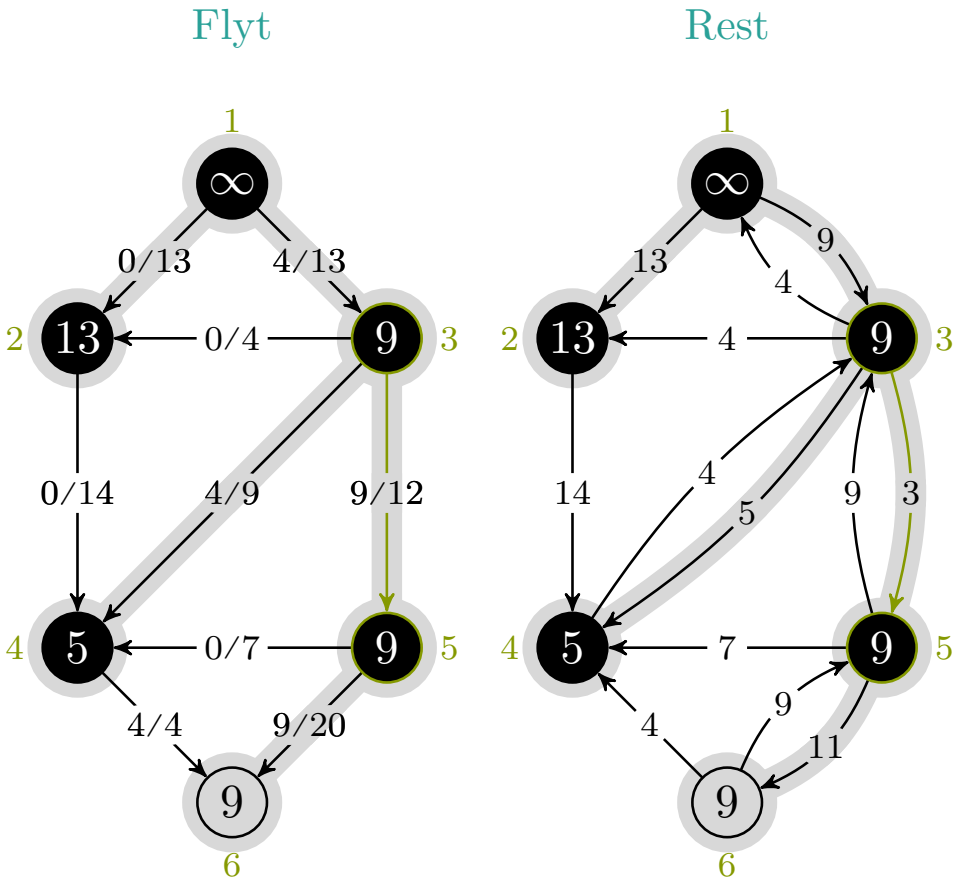
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 3, 5$



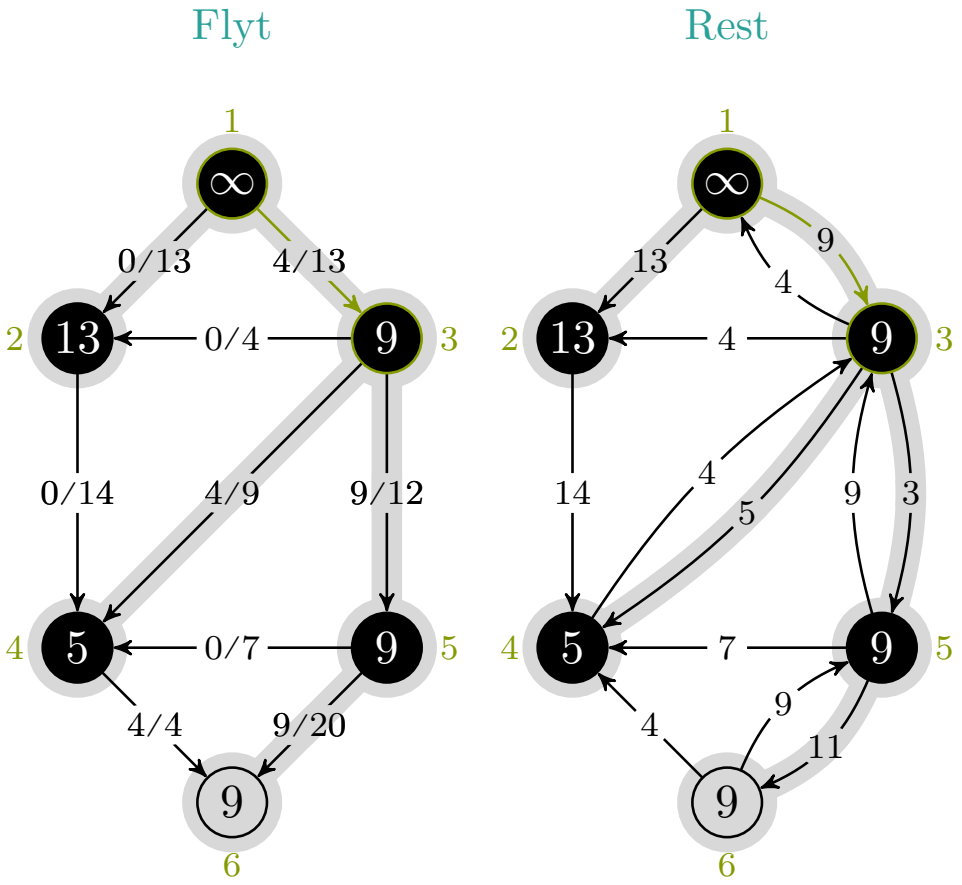
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 3, 5$



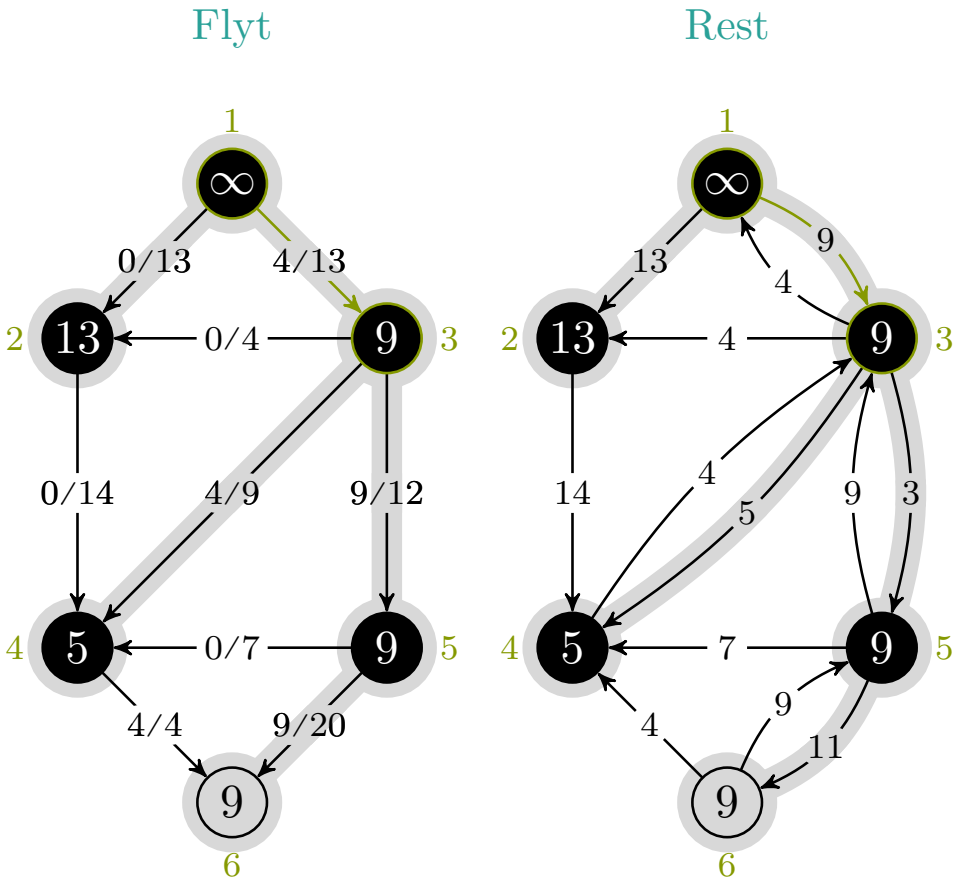
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 1, 3$



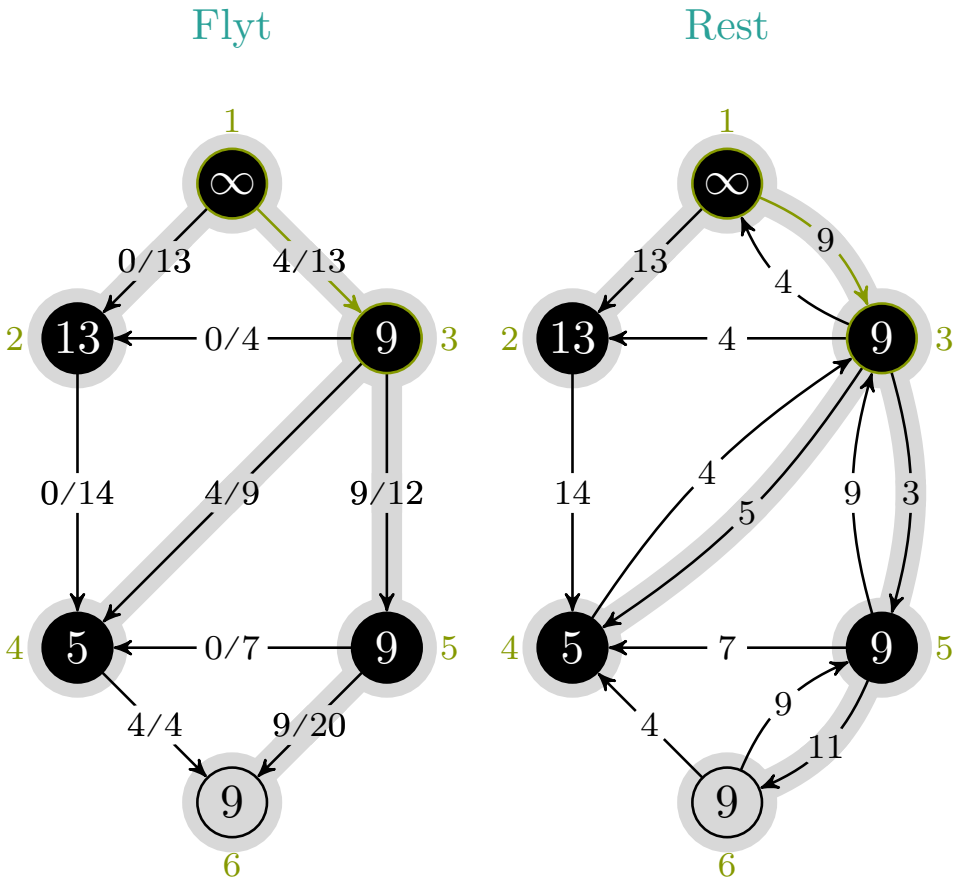
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 1, 3$



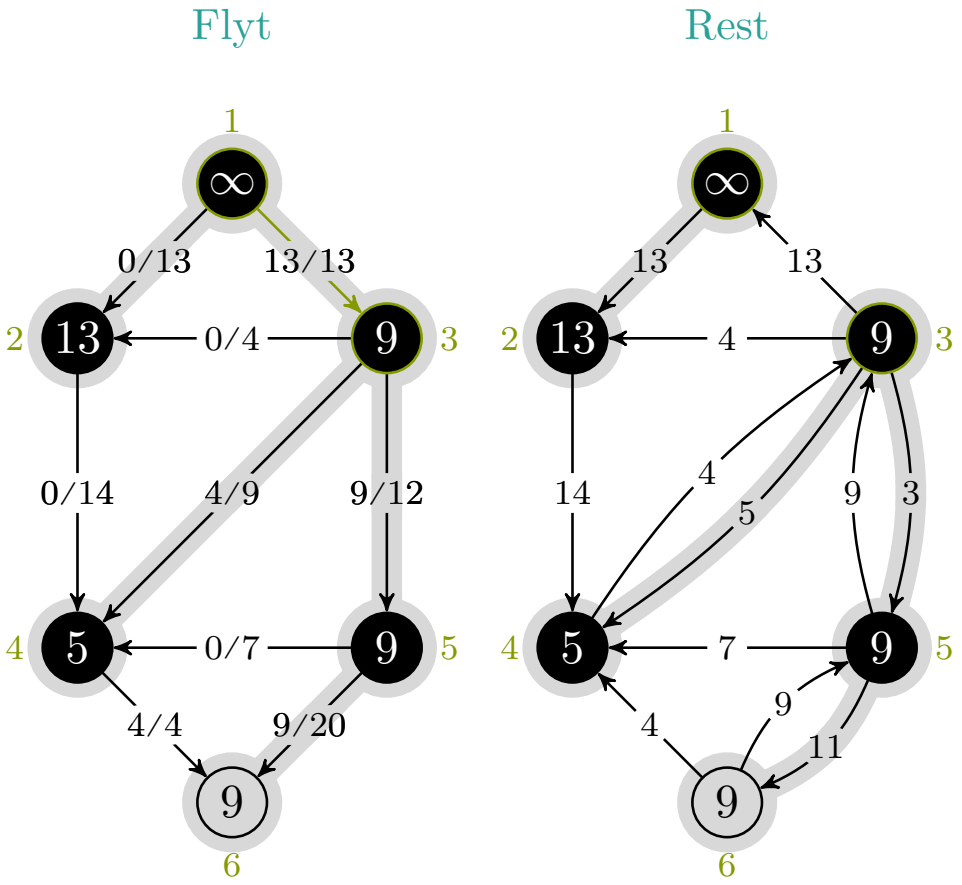
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 1, 3$



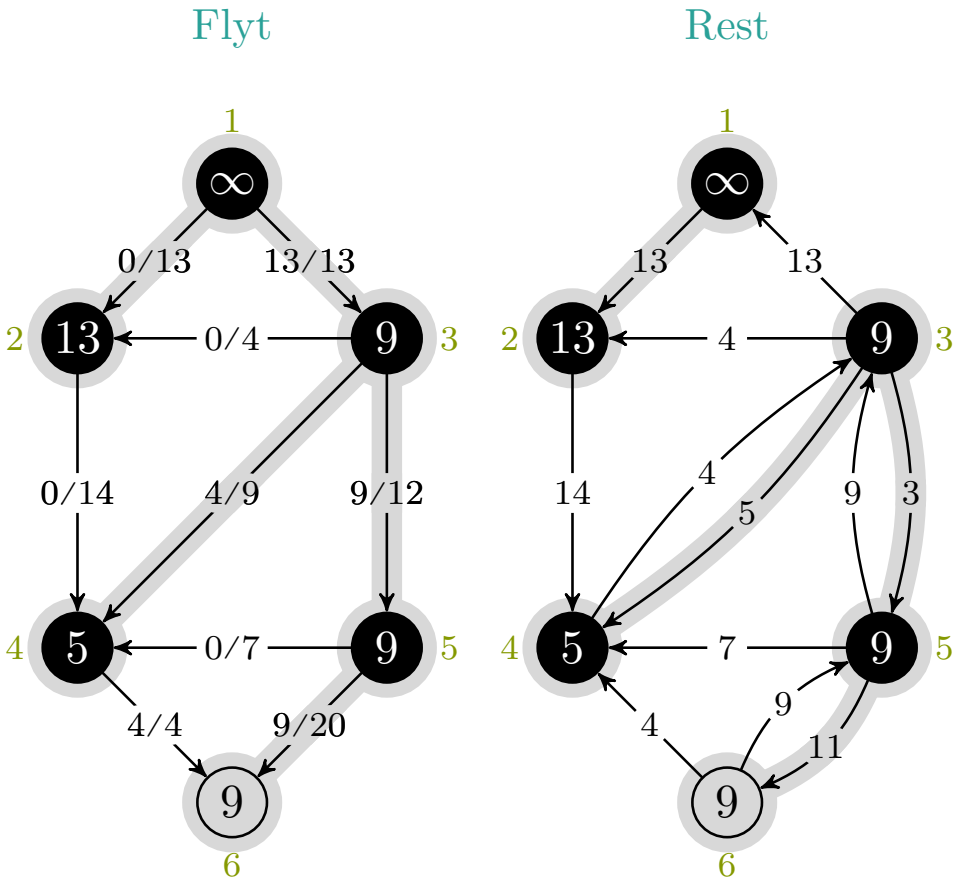
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, 1, 3$



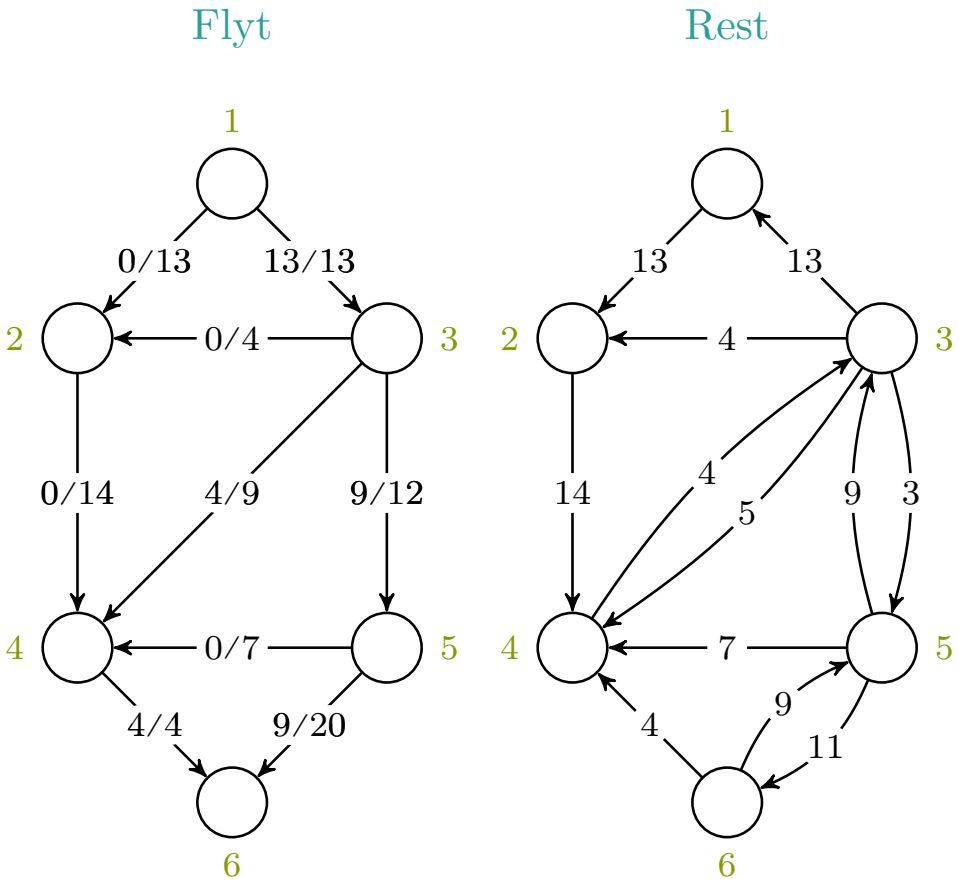
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 9, \text{NIL}, 1$



```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$

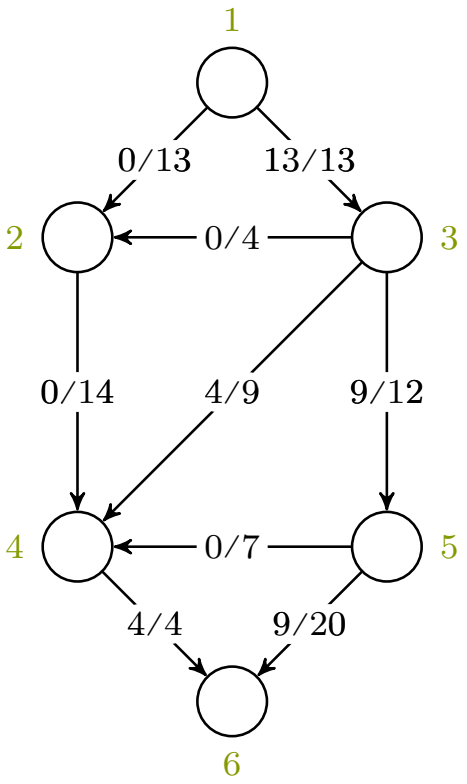


BFS-Labeling(G, s, t)

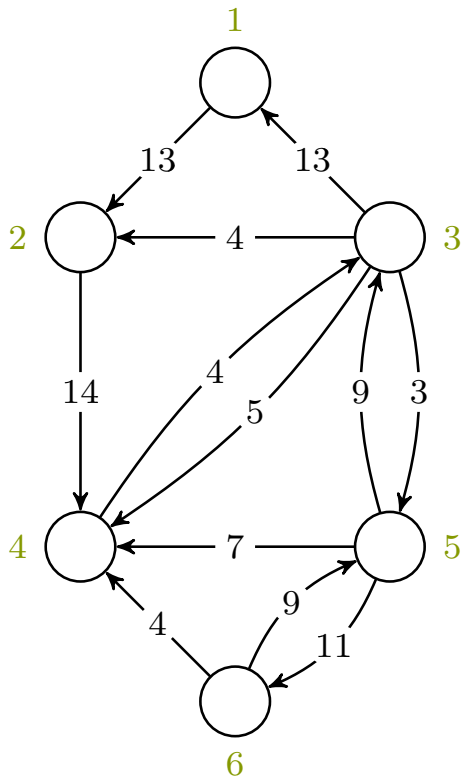
```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, -, -$

Flyt



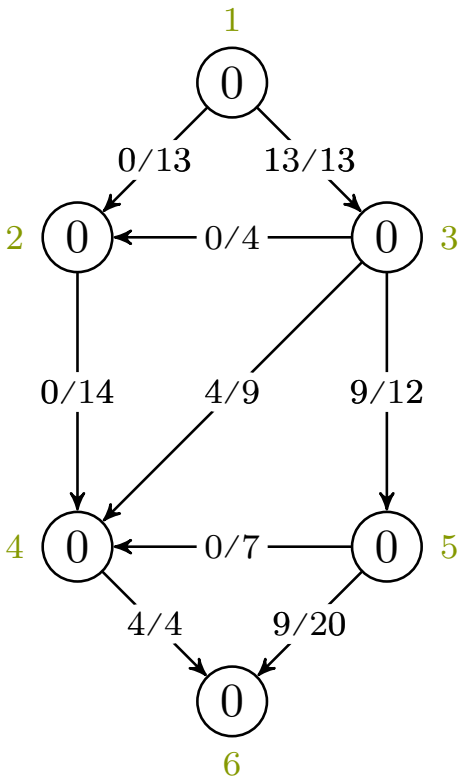
Rest



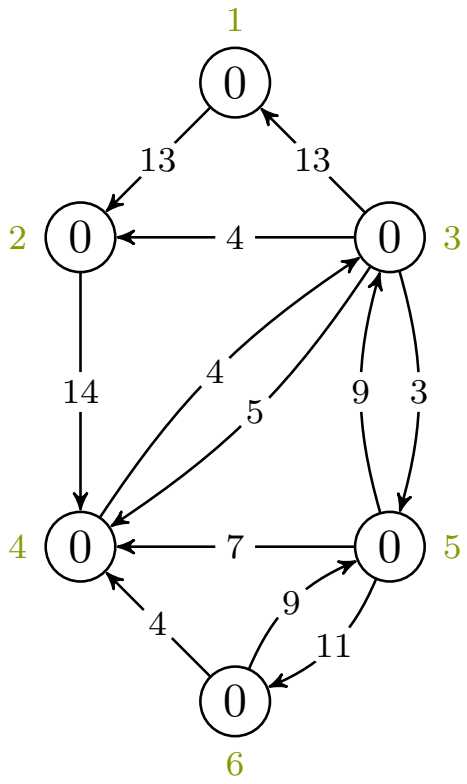
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -,-,- \;\rightarrow\; -,-,-$

Flyt



Rest

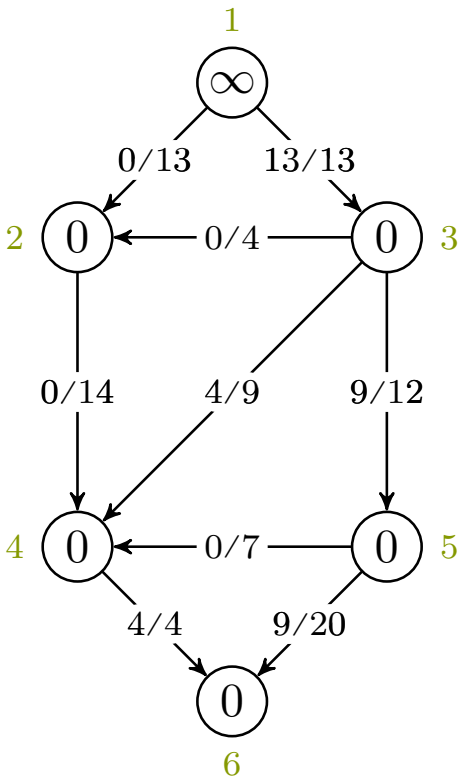


BFS-Labeling(G, s, t)

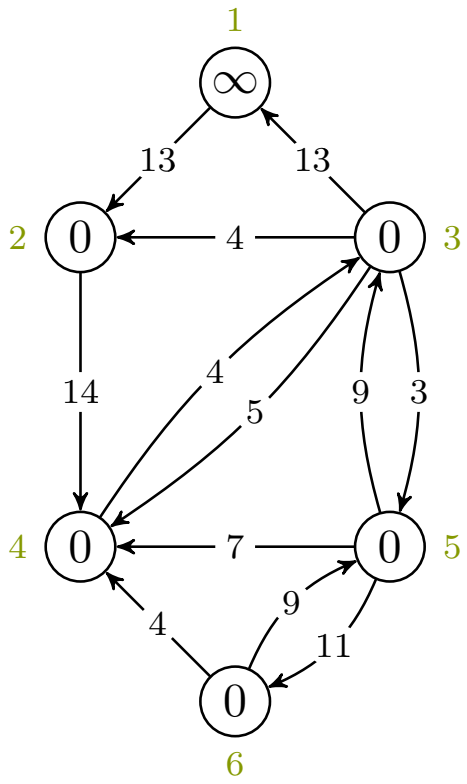
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \,\rangle\, -, -, -$

Flyt



Rest

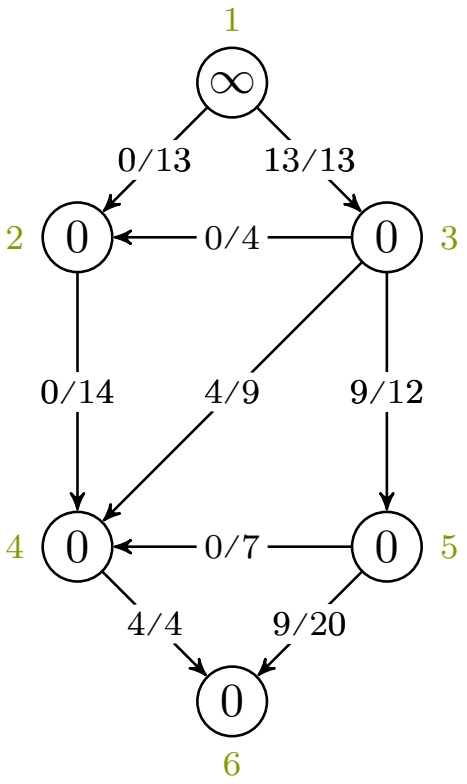


BFS-Labeling(G, s, t)

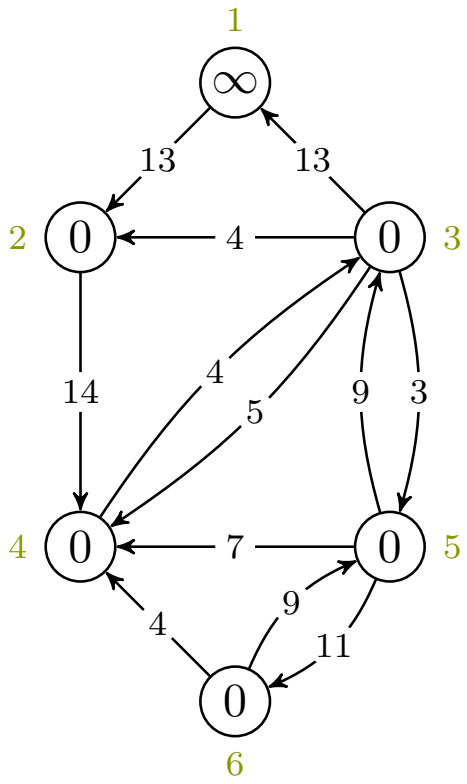
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

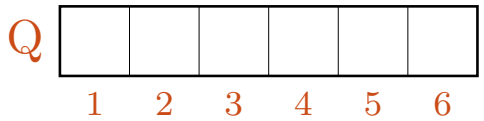
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, -, -$

Flyt



Rest

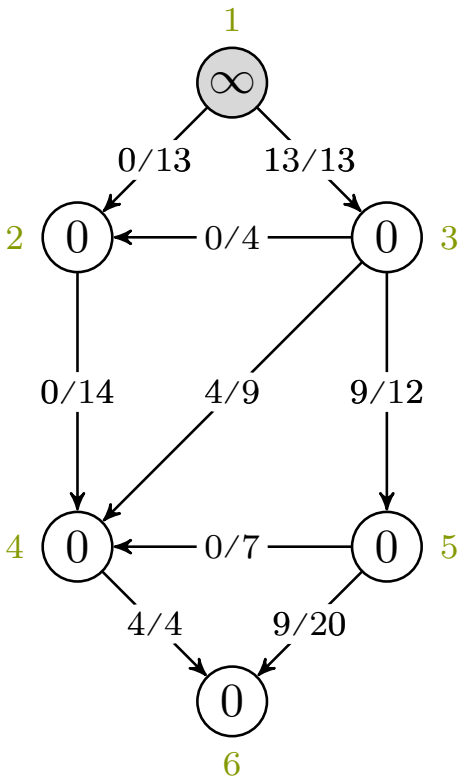




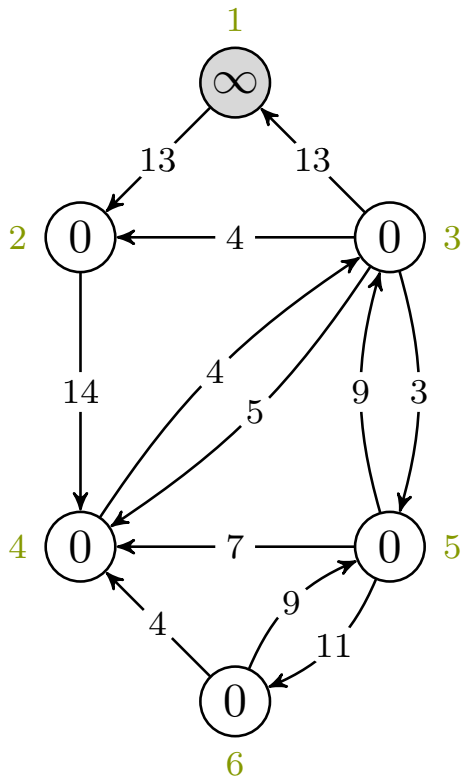
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

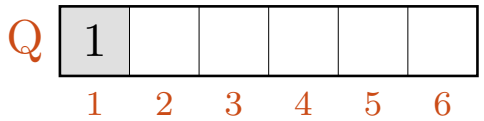
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, -, -$

Flyt



Rest

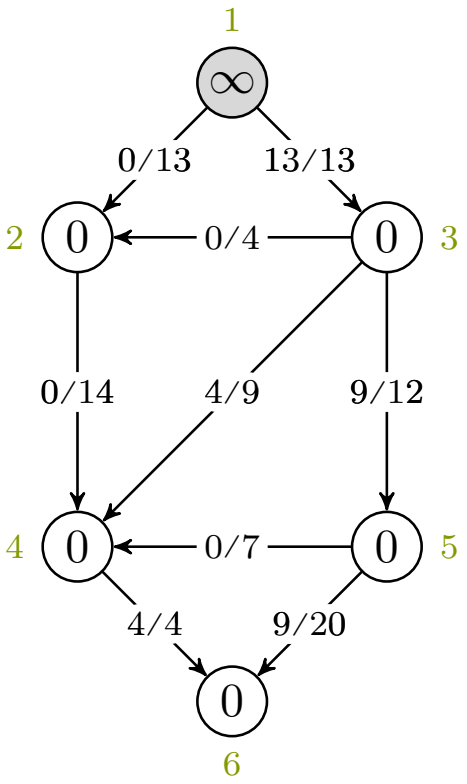




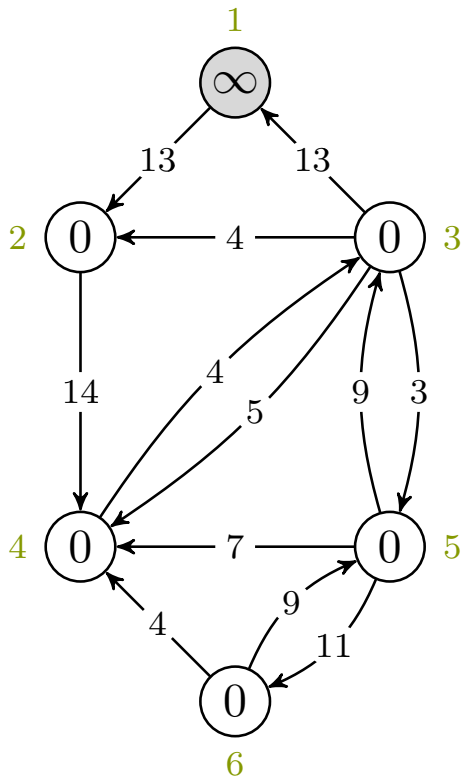
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

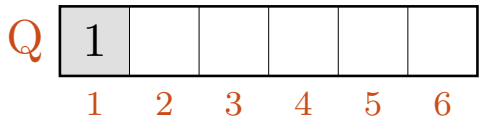
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, -, -$

Flyt



Rest



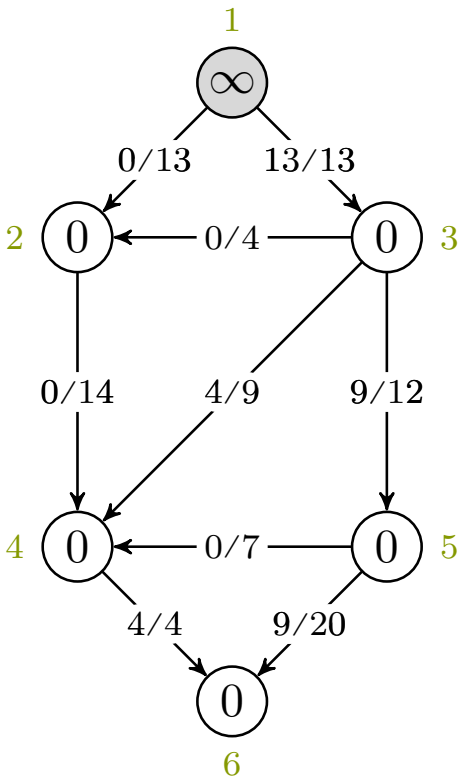


BFS-Labeling(G, s, t)

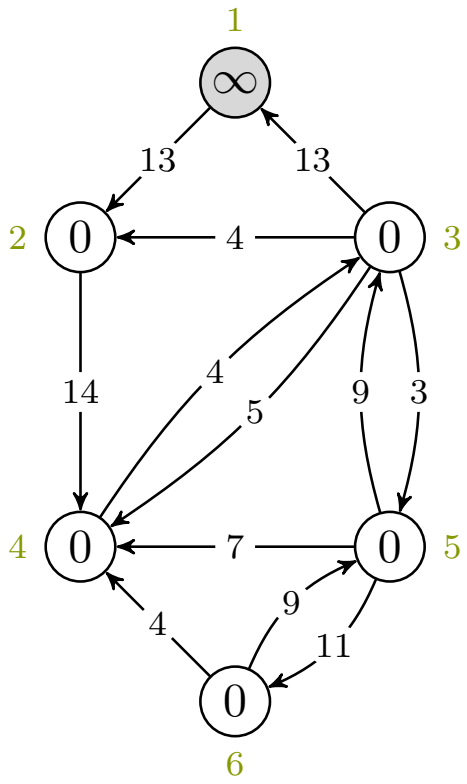
```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

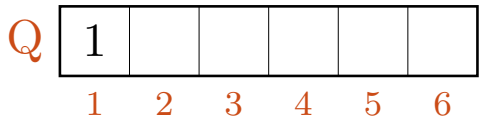
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, -$

Flyt



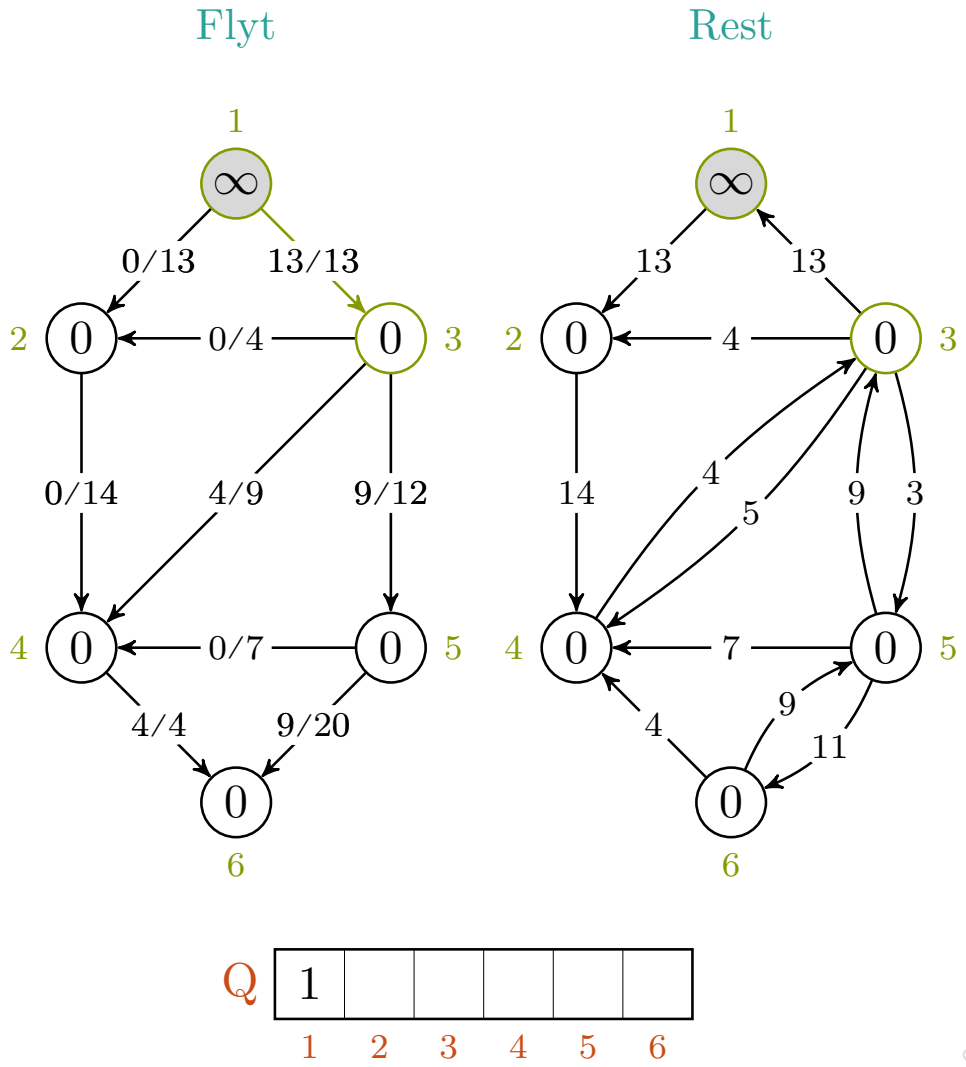
Rest





```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 3$



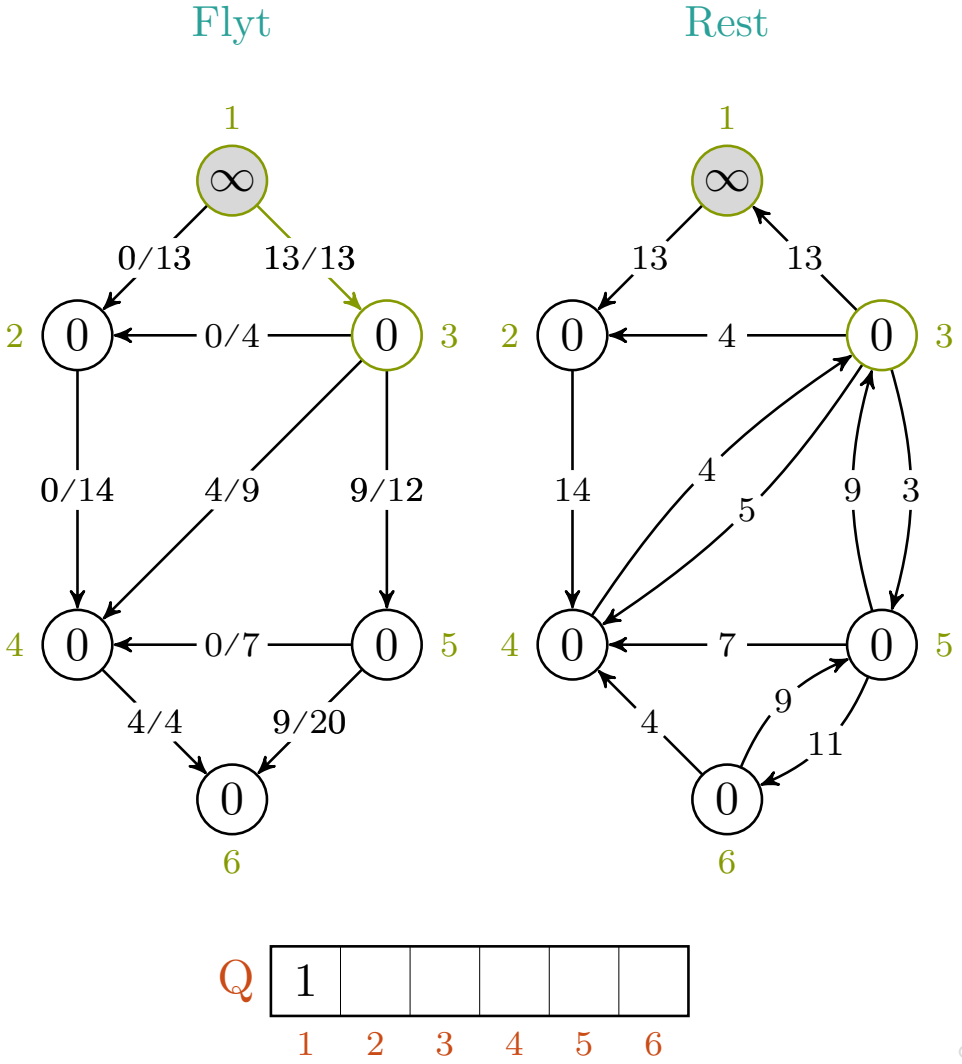
Q

1					
1	2	3	4	5	6

BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

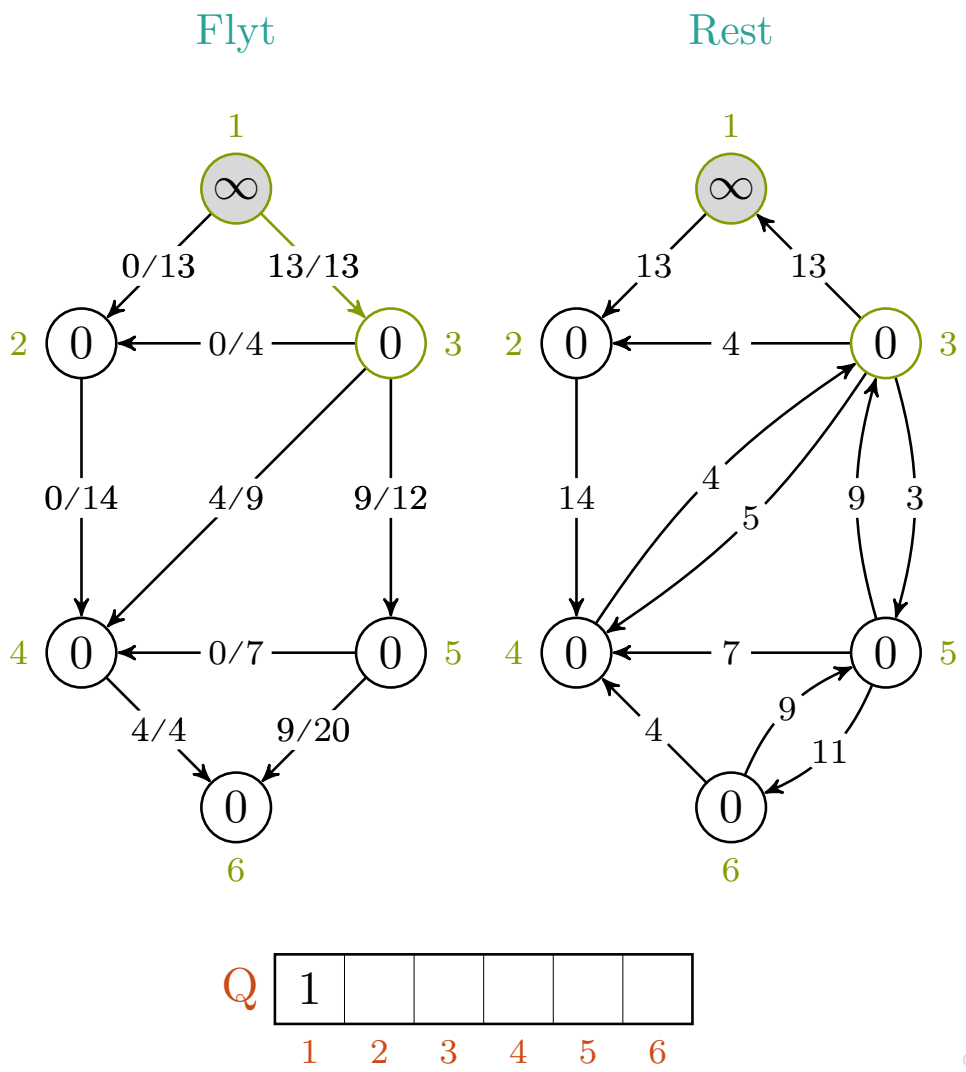
$c_f(p), u, v \;=\; -, -, - \,\rangle\, -, 1, 3$



BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \,\rangle\, -, 1, 3$



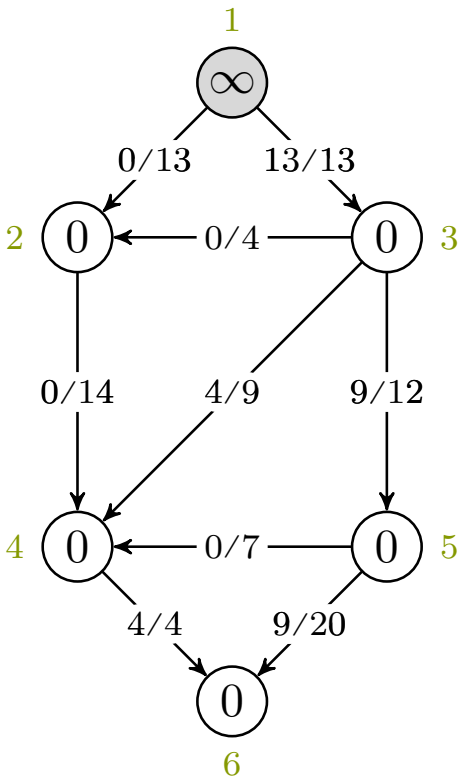
Q | 1 | | | | | |

1 2 3 4 5 6

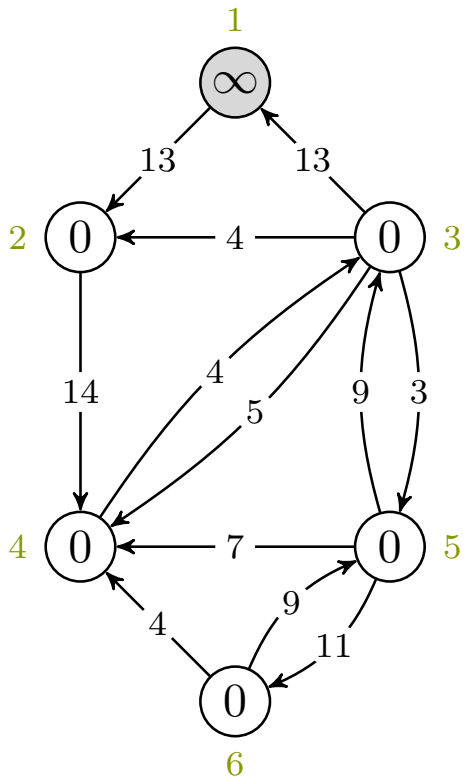
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

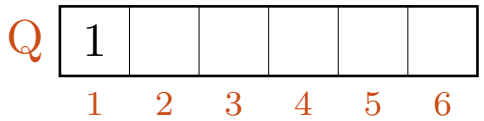
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, -$

Flyt



Rest

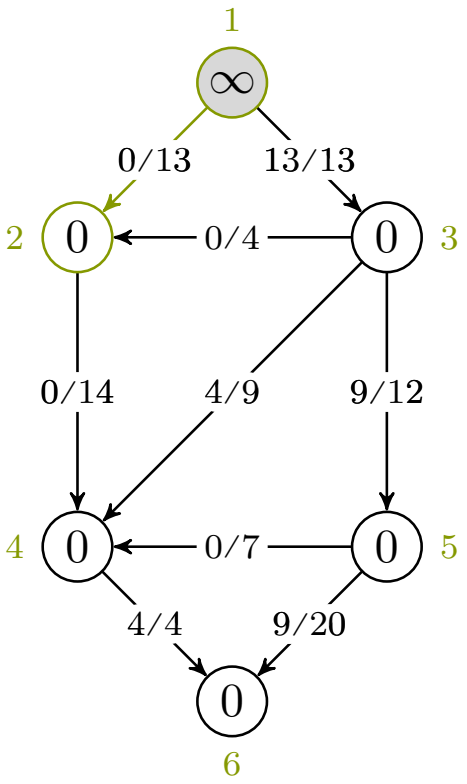




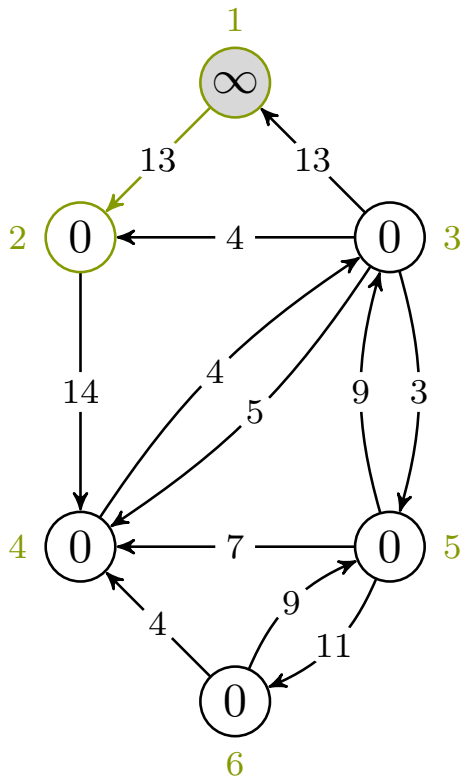
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

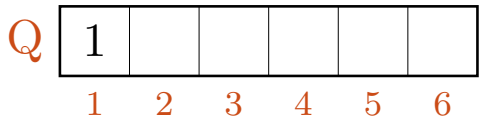
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 2$

Flyt



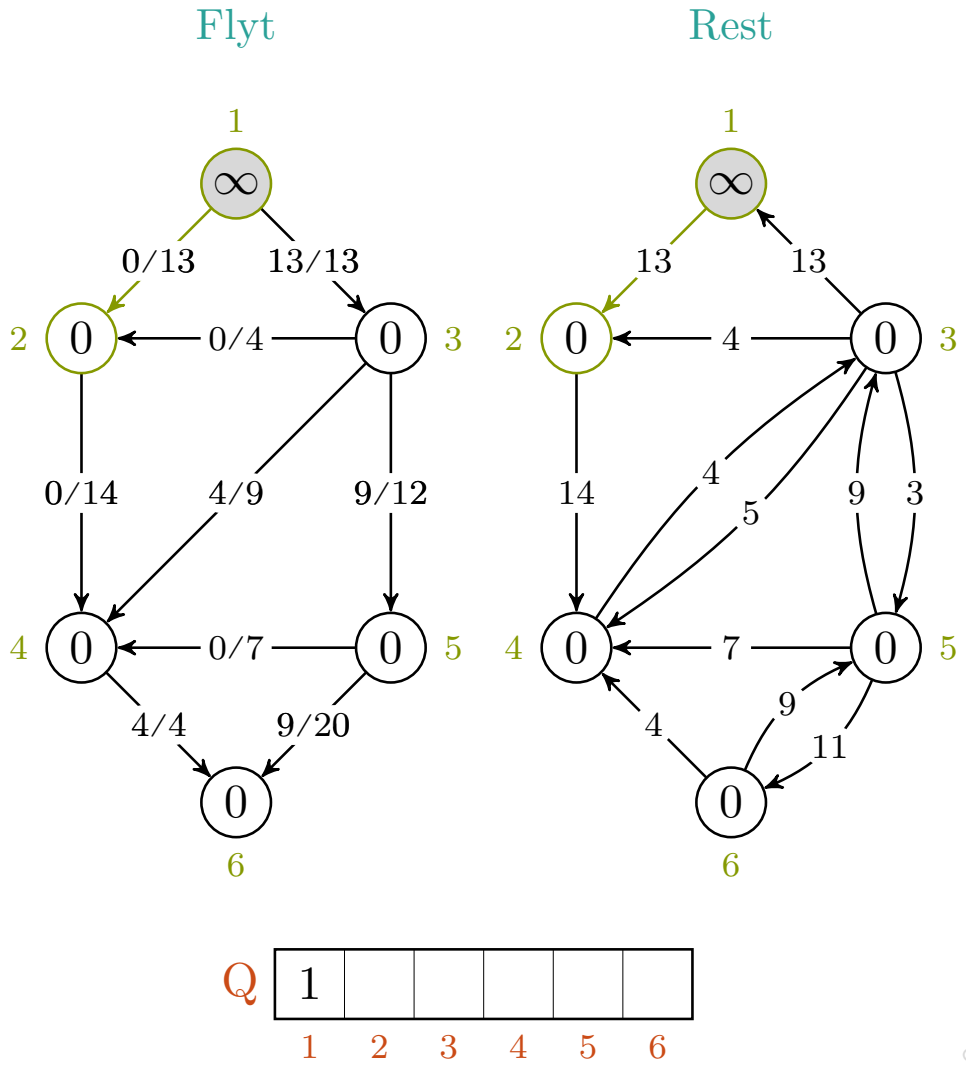
Rest





```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 2$



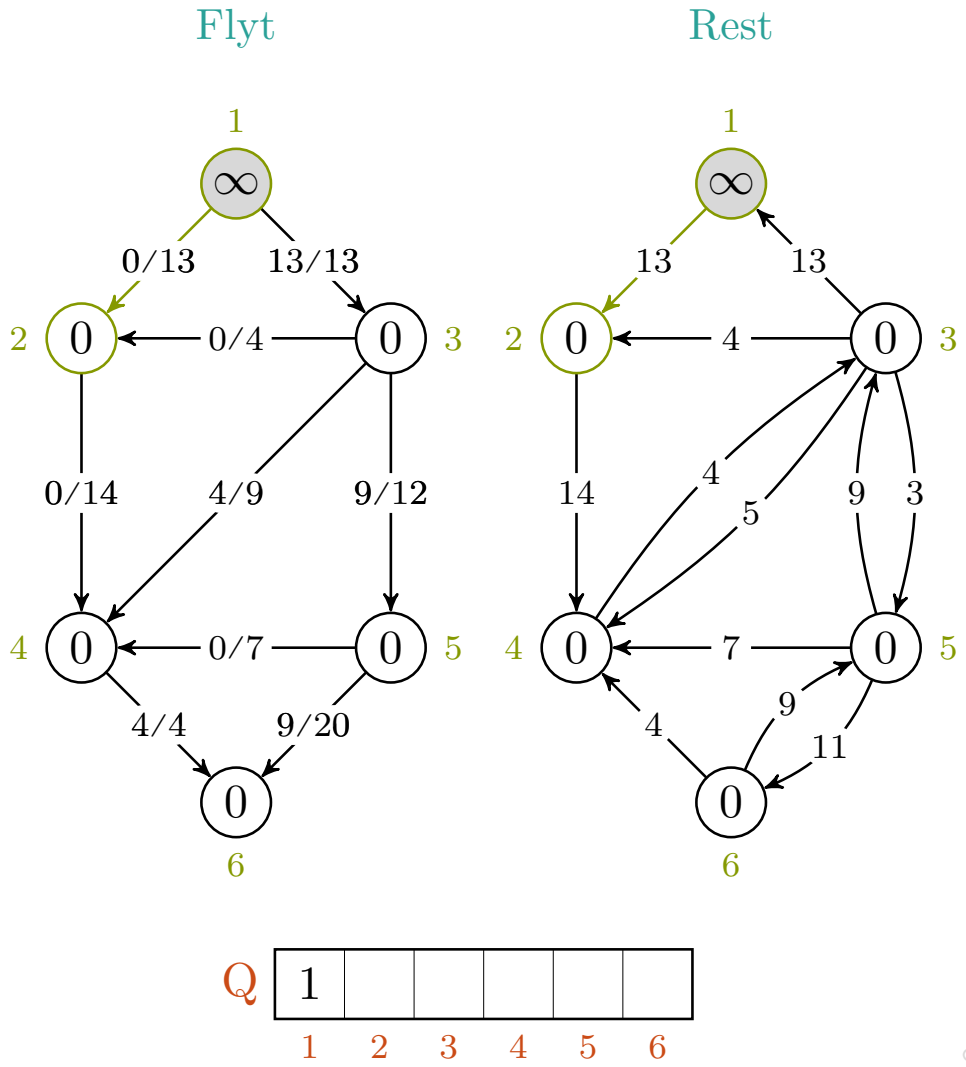
Q

1					
1	2	3	4	5	6

BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

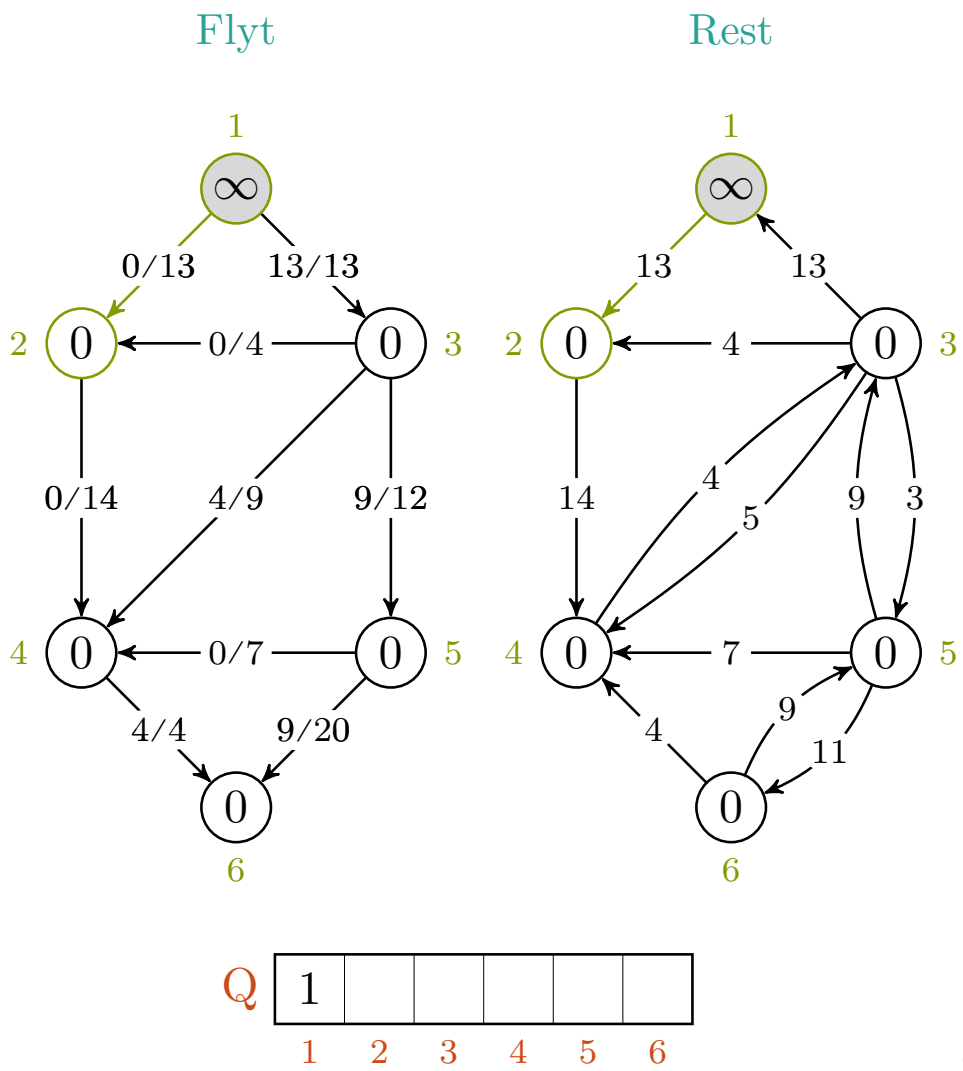
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 2$



BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

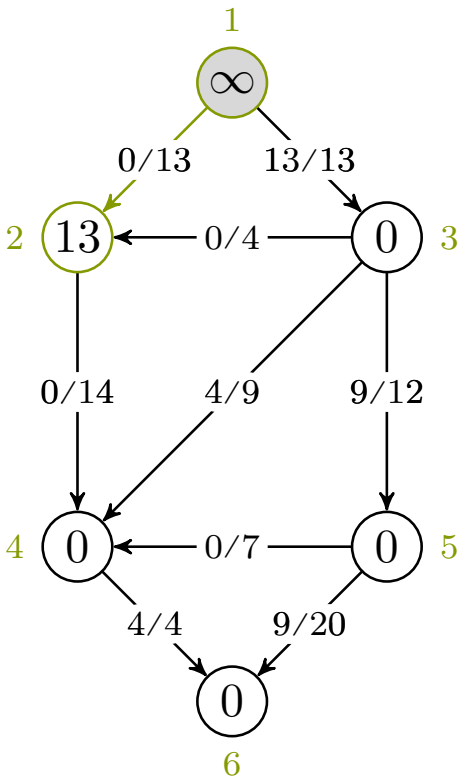
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 2$



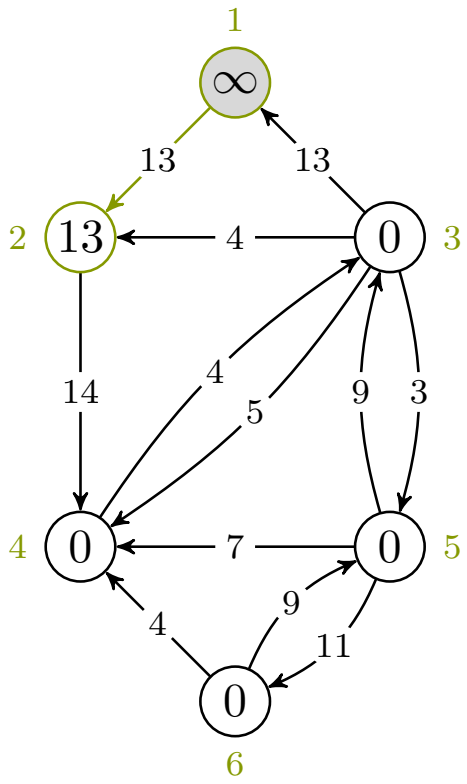
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2        u.f = 0
 3        u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
16                    Enqueue(Q, v)
17  return t.f ≠ 0
```

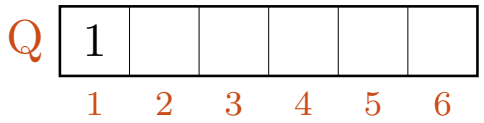
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 2$

Flyt



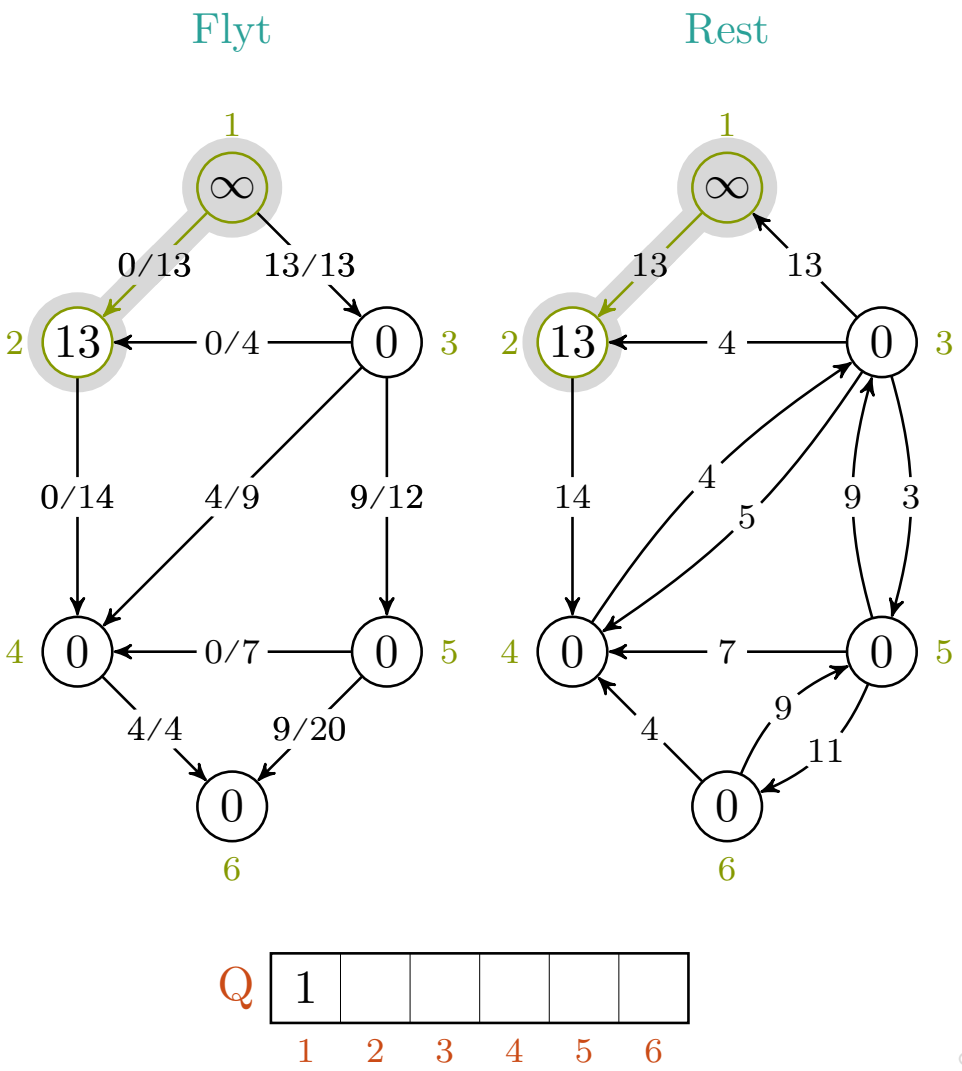
Rest





```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, 2$

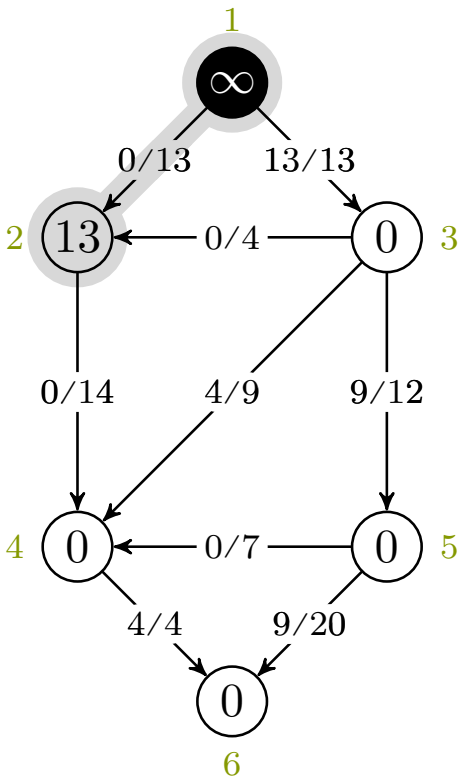


BFS-Labeling(G, s, t)

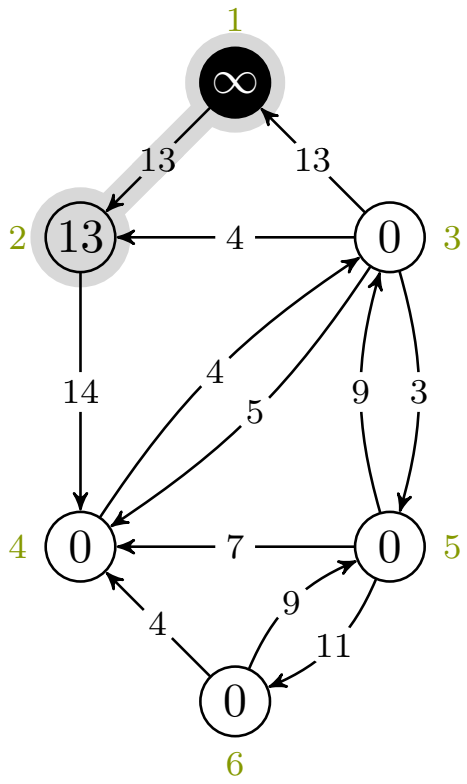
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

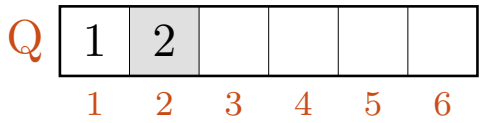
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, -$

Flyt



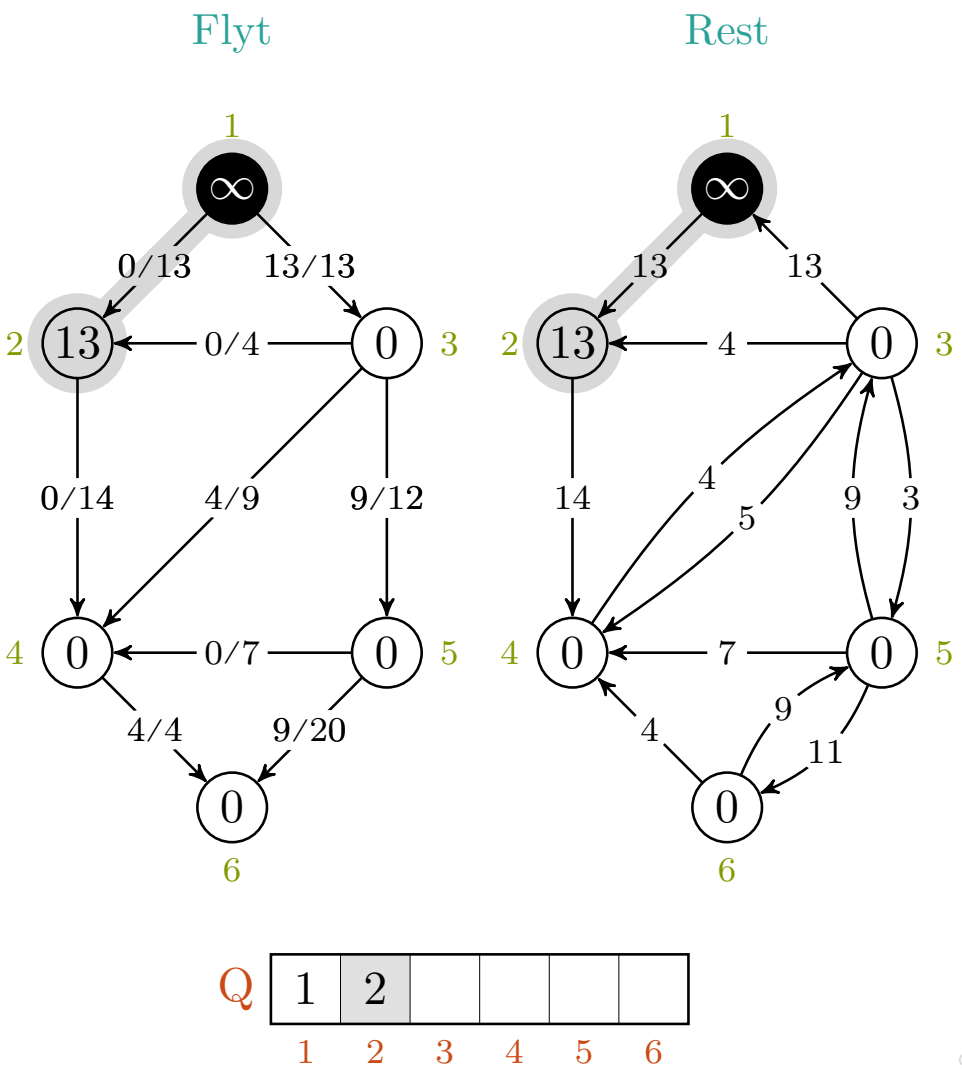
Rest





```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \rightarrow -, 1, -$

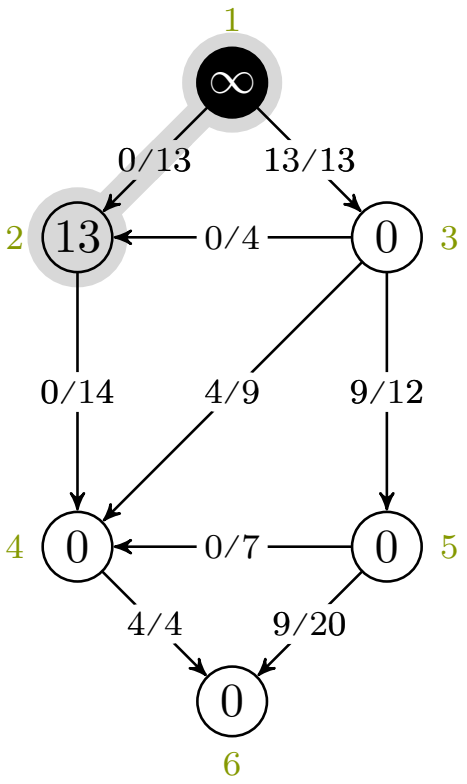


BFS-Labeling(G, s, t)

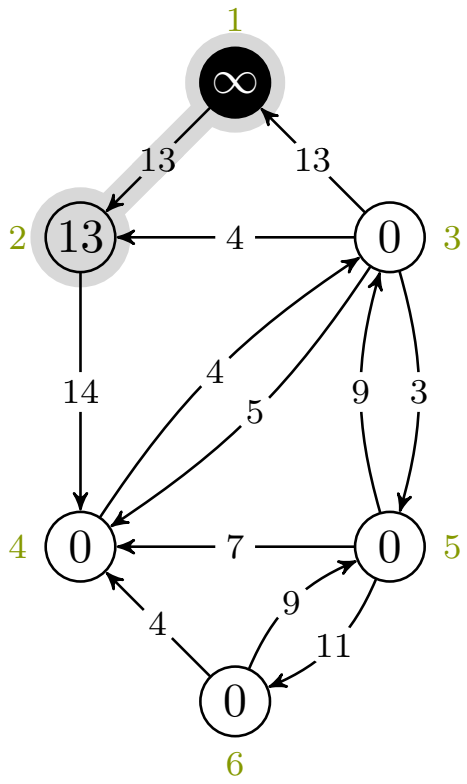
```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

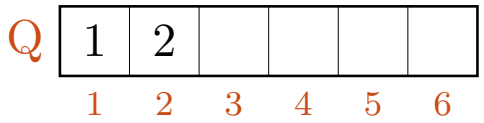
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, -$

Flyt



Rest



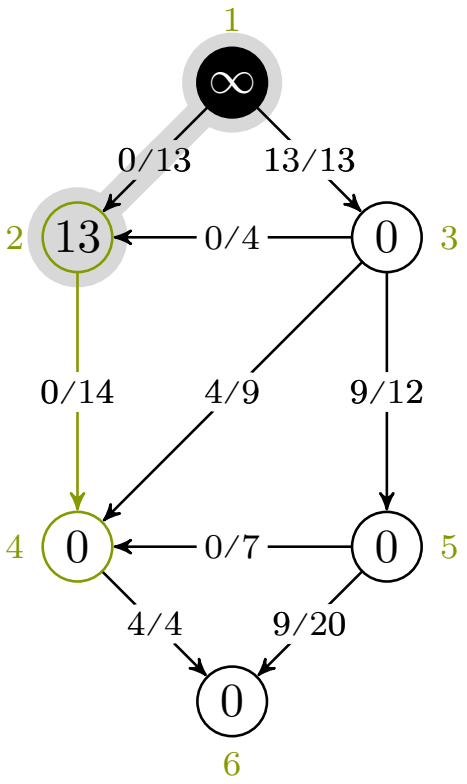


BFS-Labeling(G, s, t)

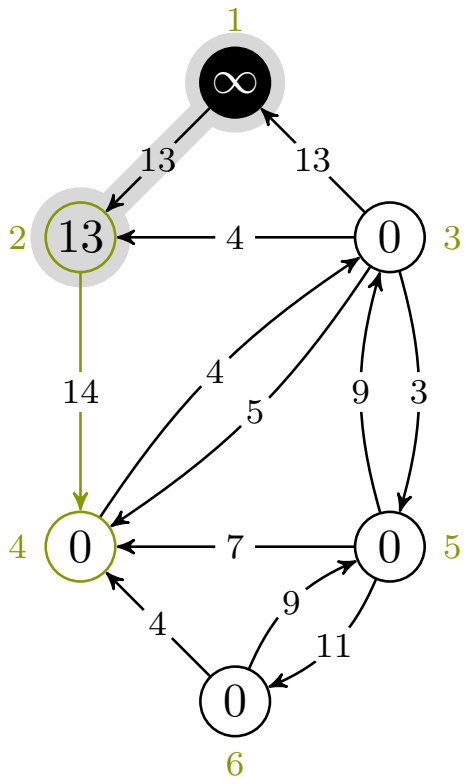
```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

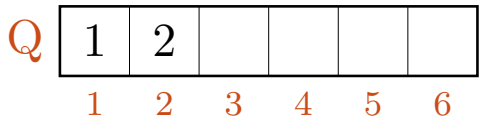
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 2, 4$

Flyt



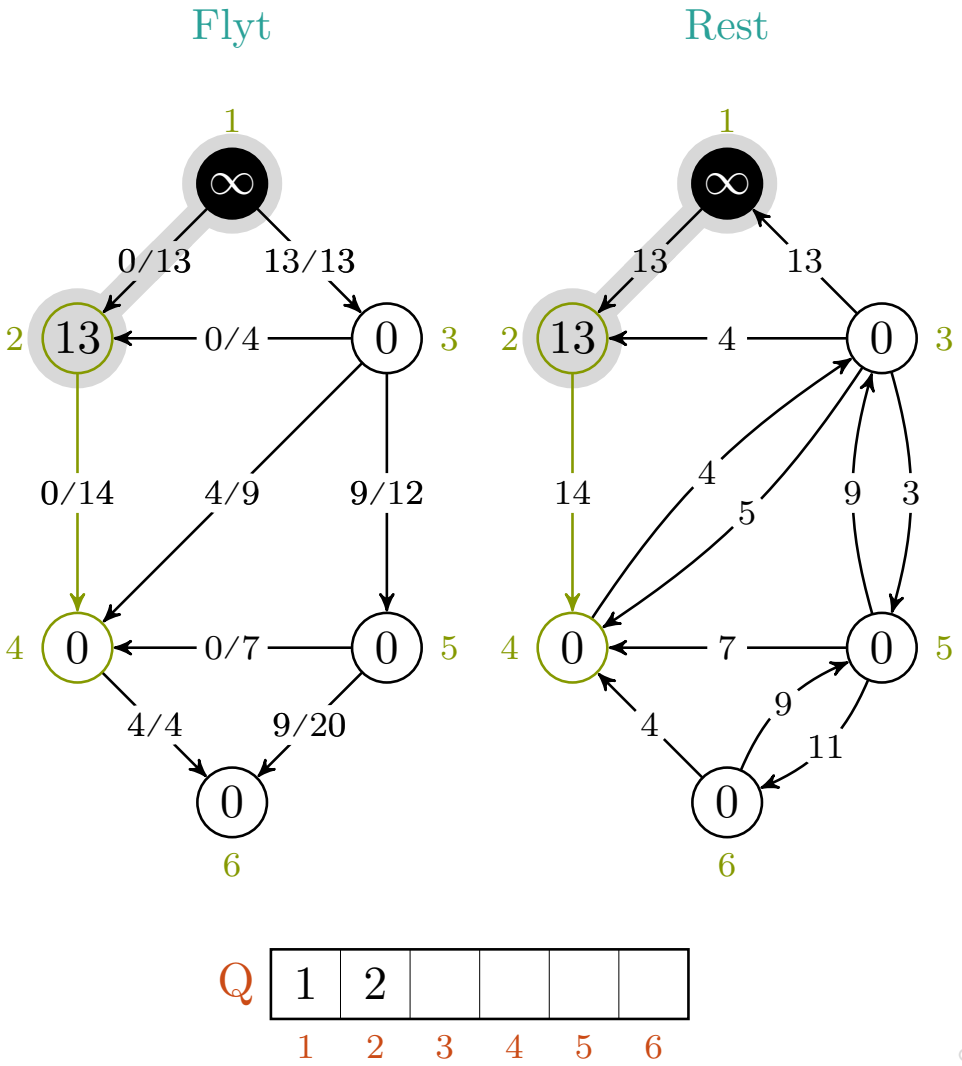
Rest





```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

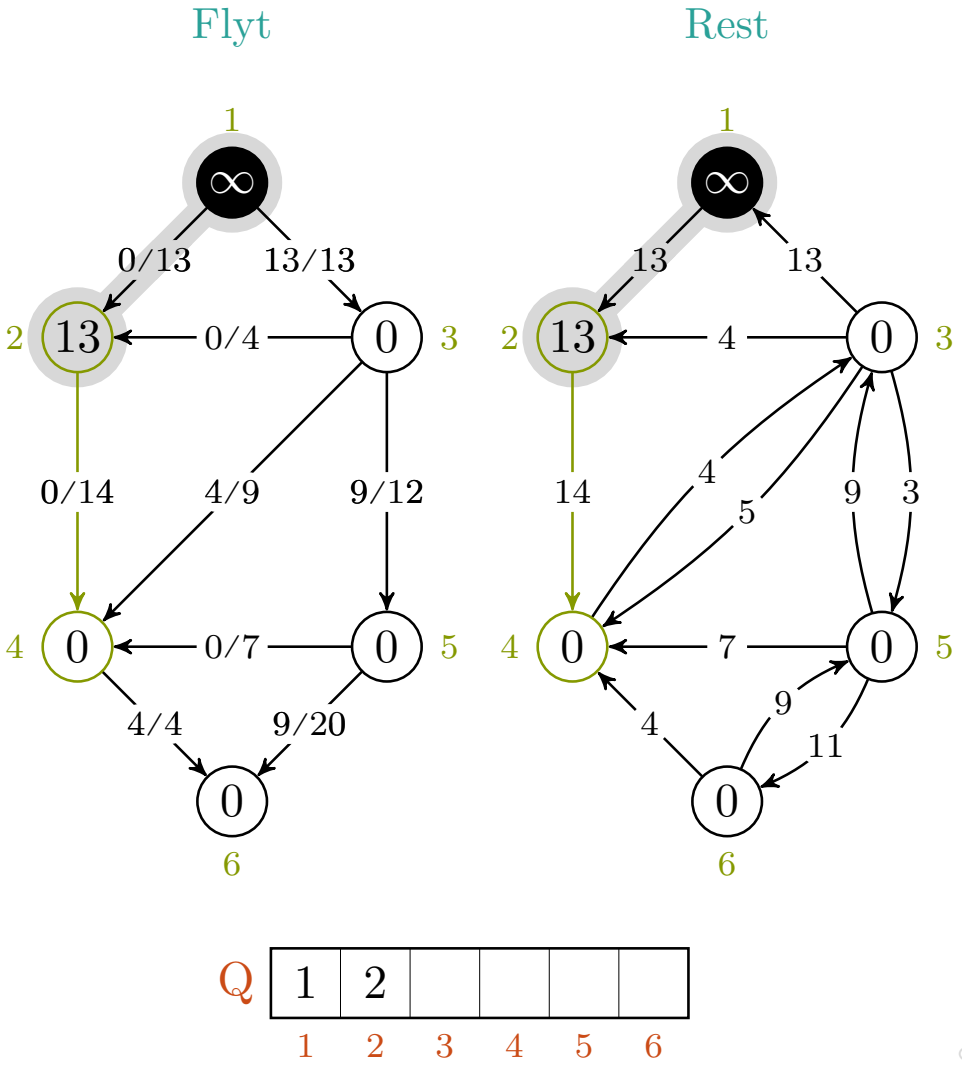
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 2, 4$



BFS-Labeling(G, s, t)

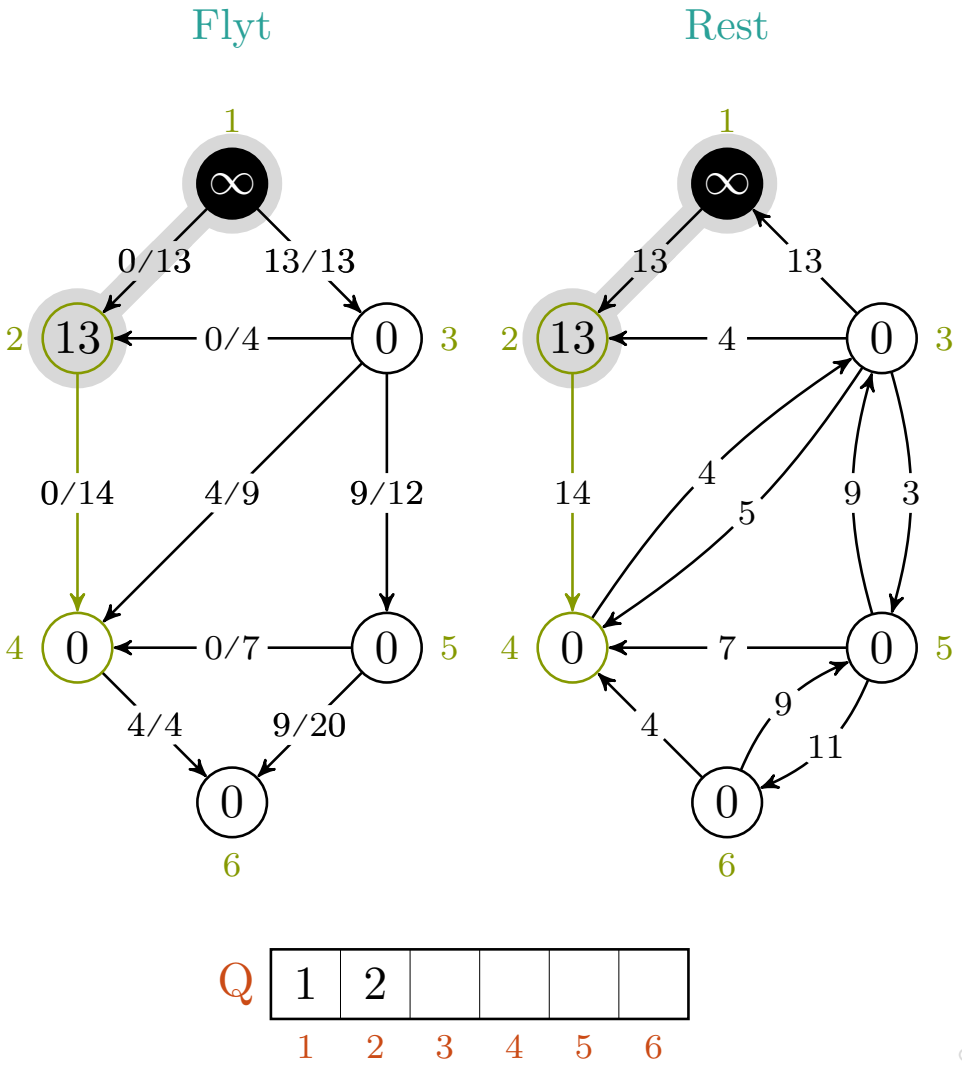
```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 4$



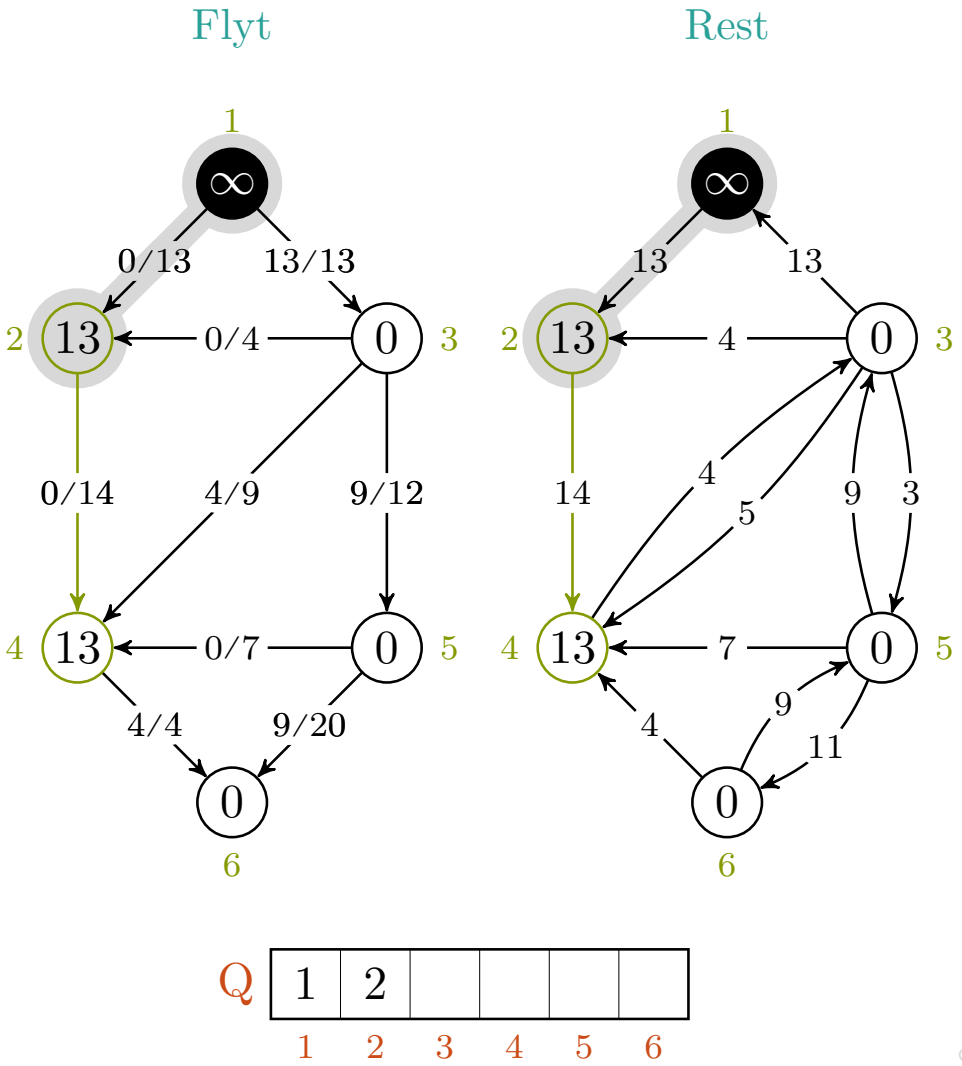
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \rightarrow -, 2, 4$



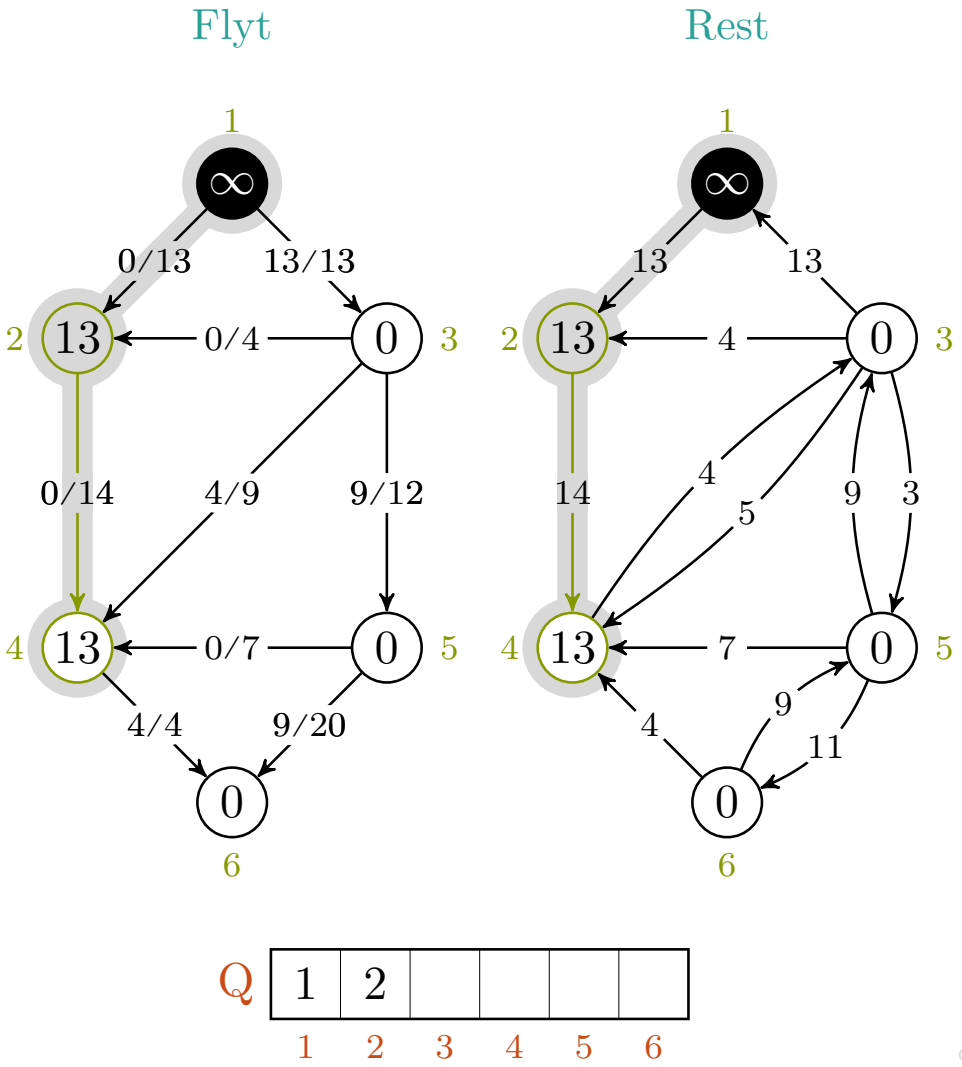
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2        u.f = 0
 3        u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
16                    Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 4$



```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

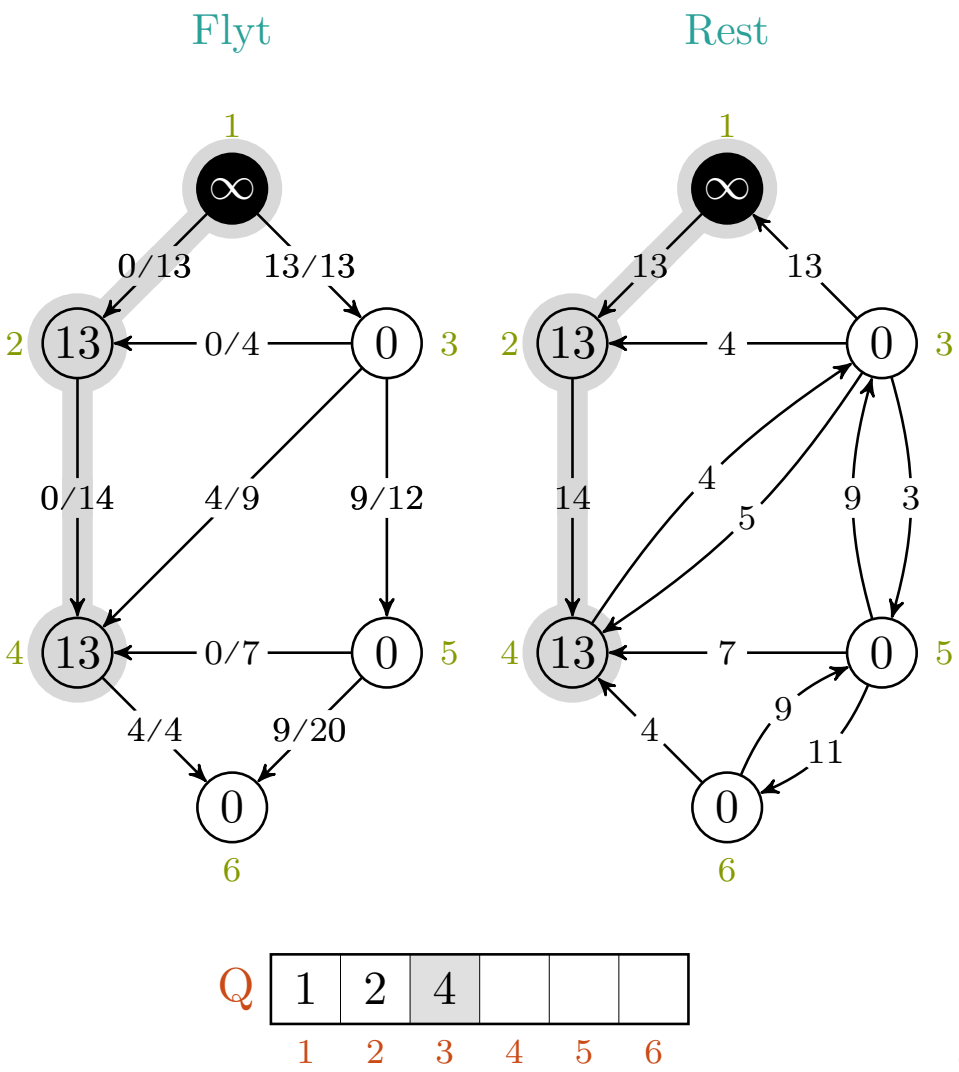
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 4$



BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

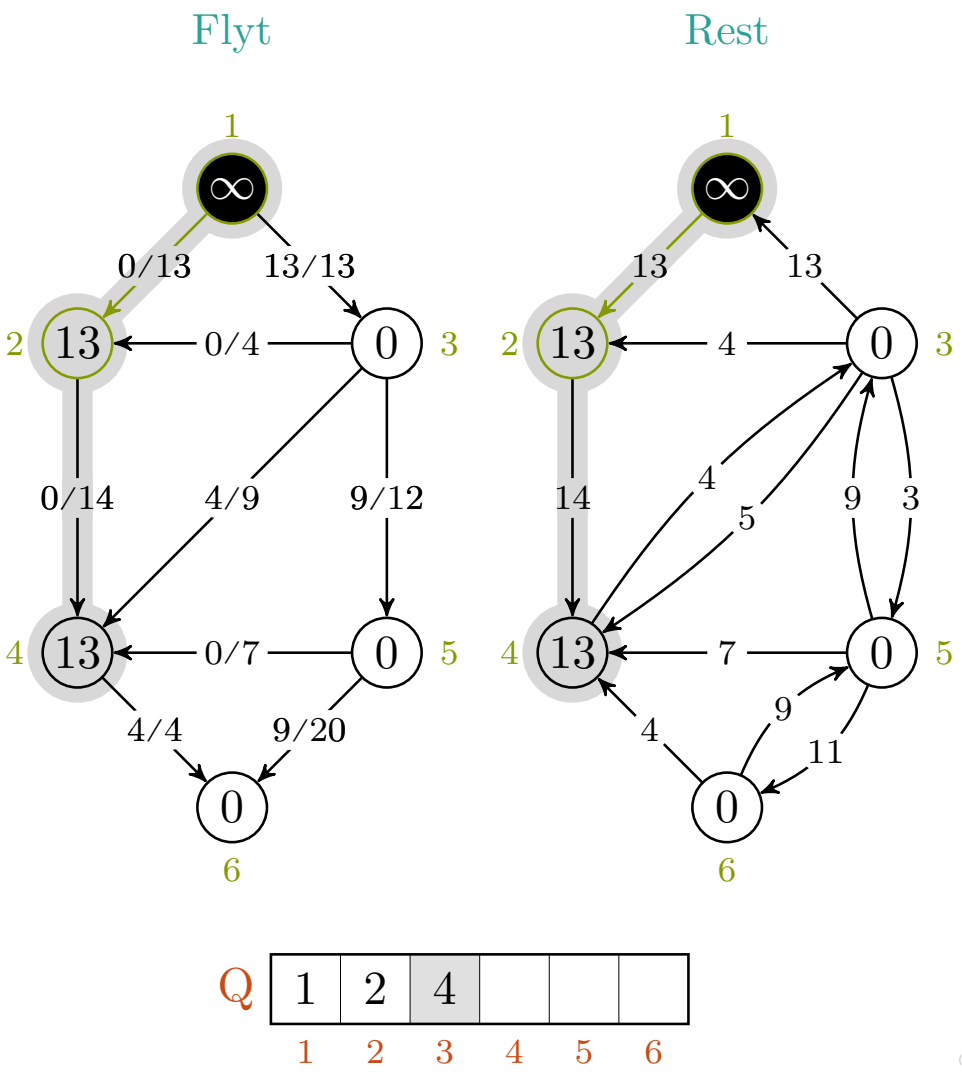
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, -$



BFS-Labeling(G, s, t)

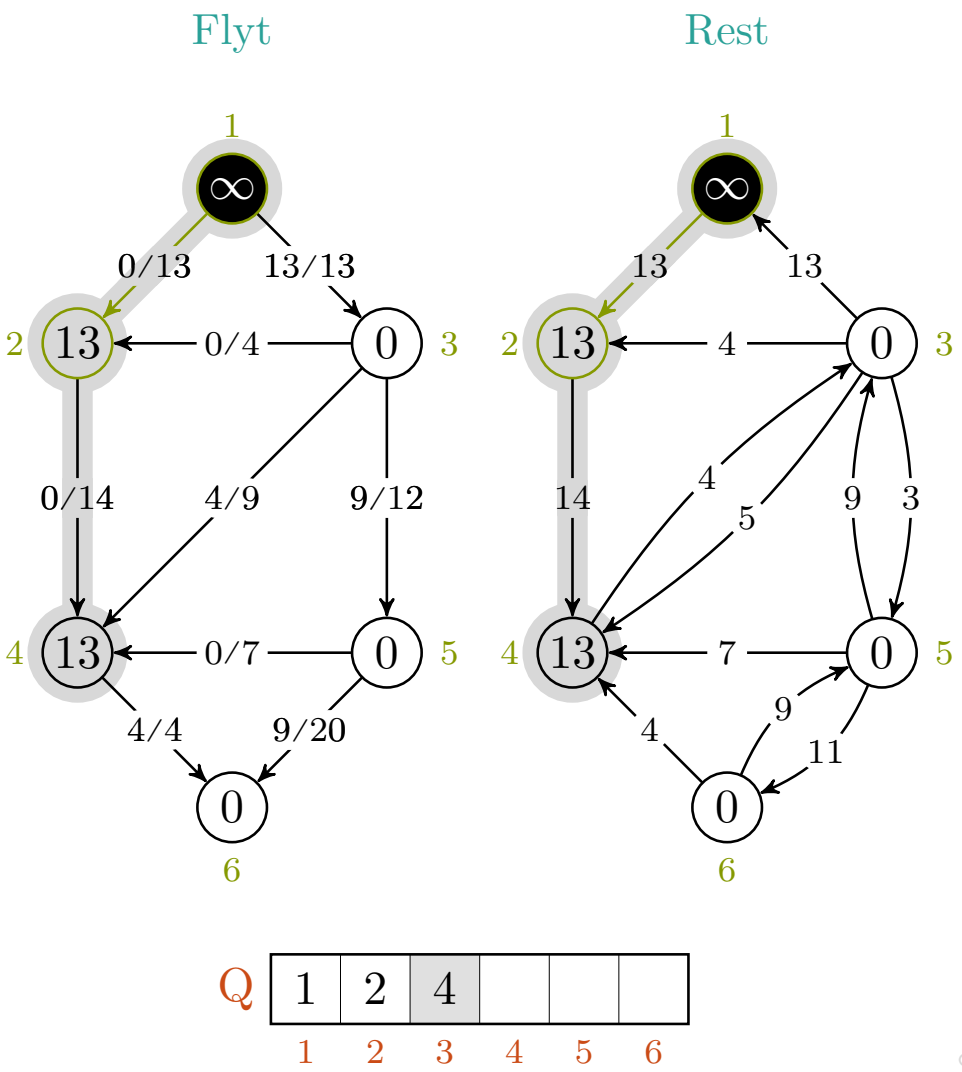
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 1$



```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

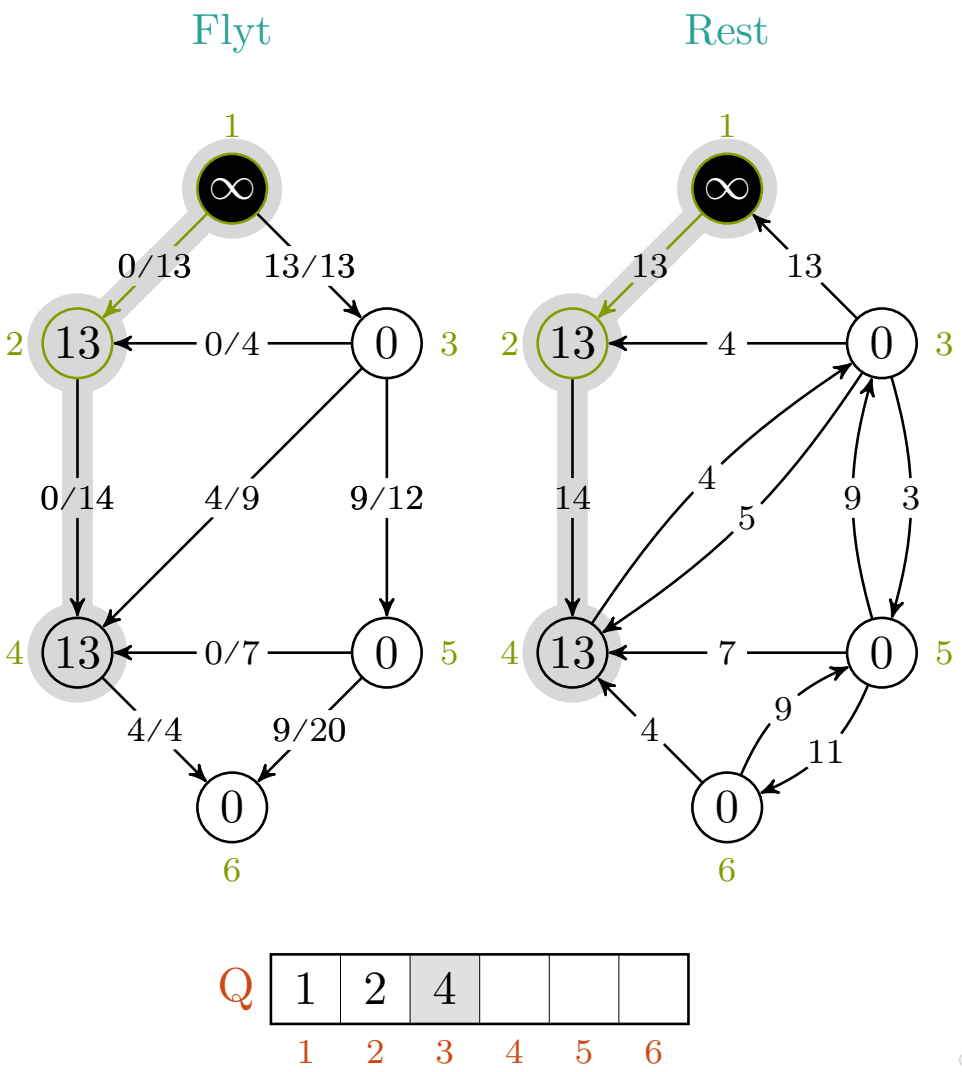
$c_f(p), u, v \;=\; -, -, - \rightarrow -, 2, 1$



BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

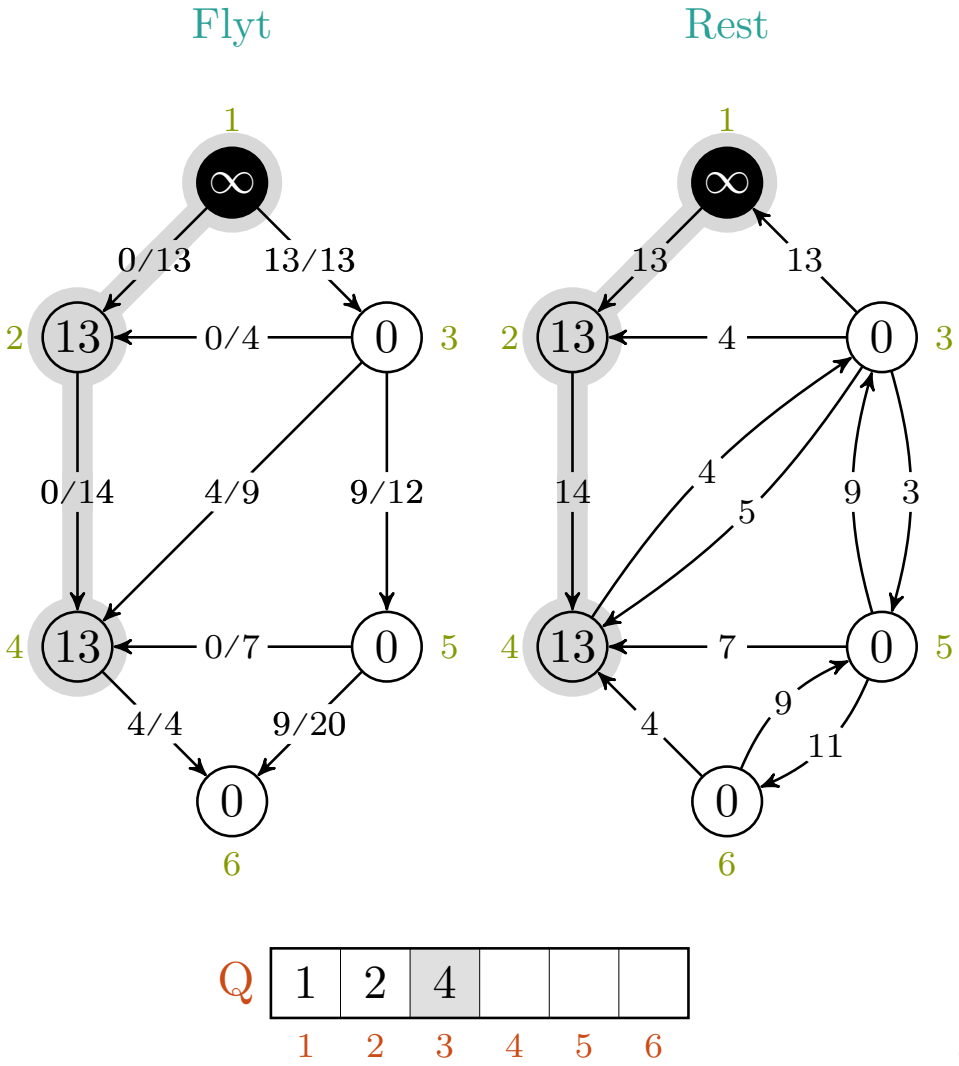
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 1$



BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

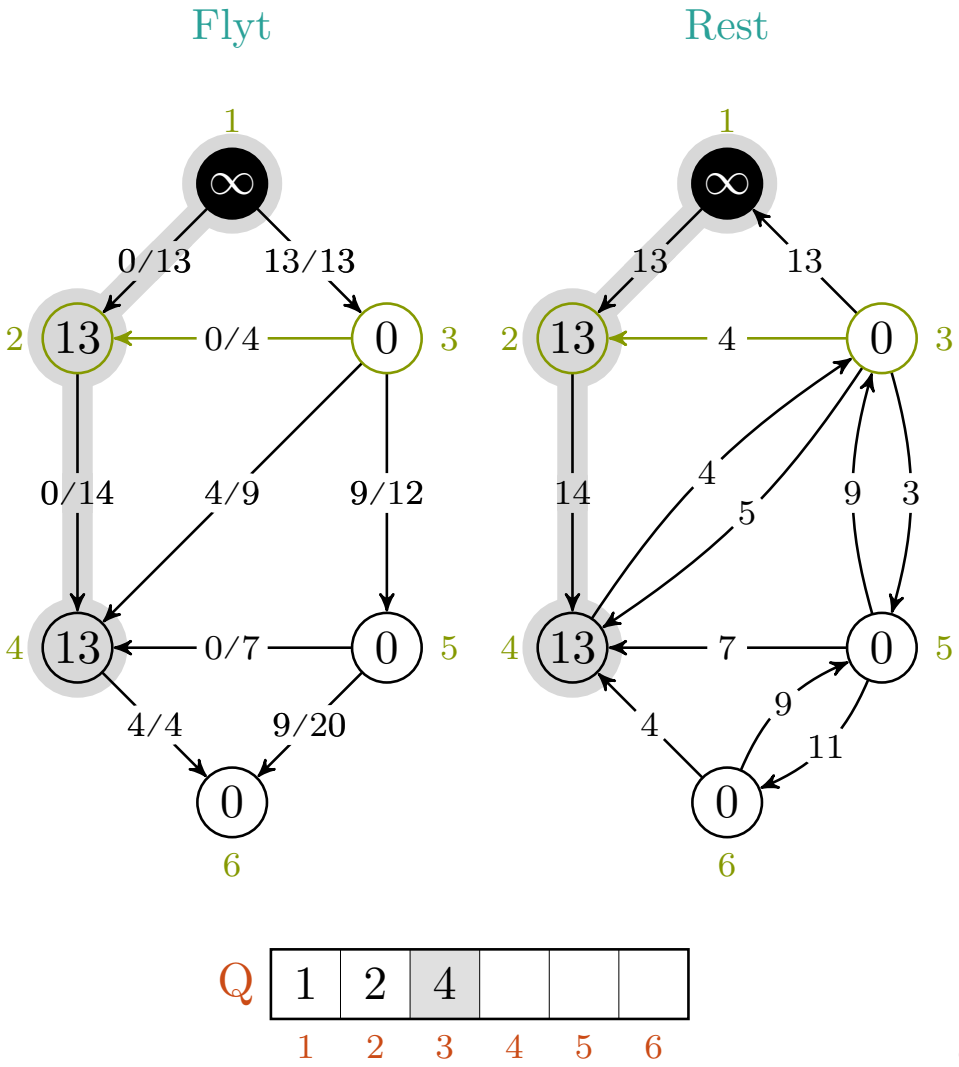
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, -$



BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 3$

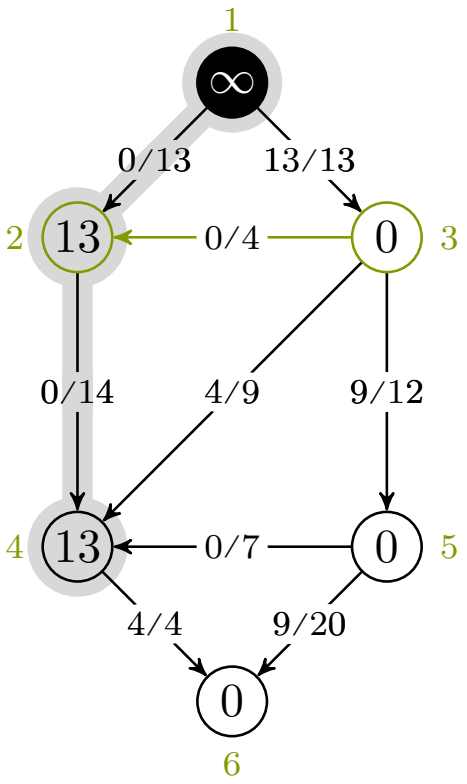


BFS-Labeling(G, s, t)

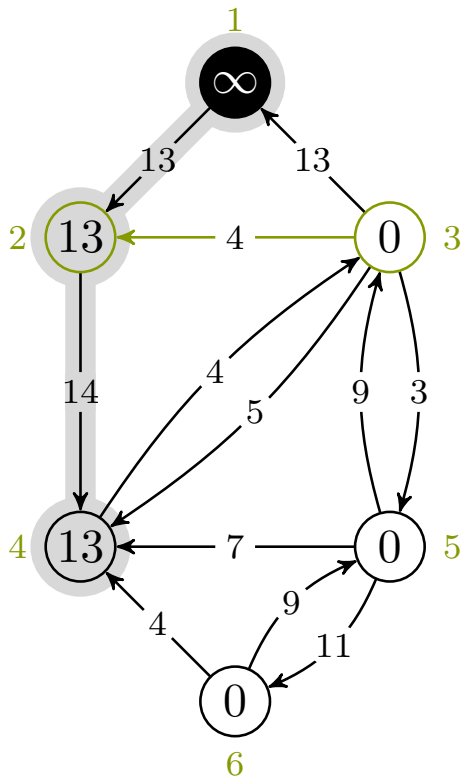
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

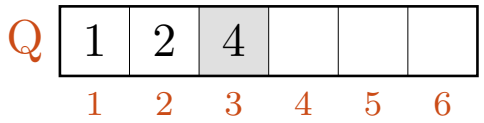
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 3$

Flyt



Rest

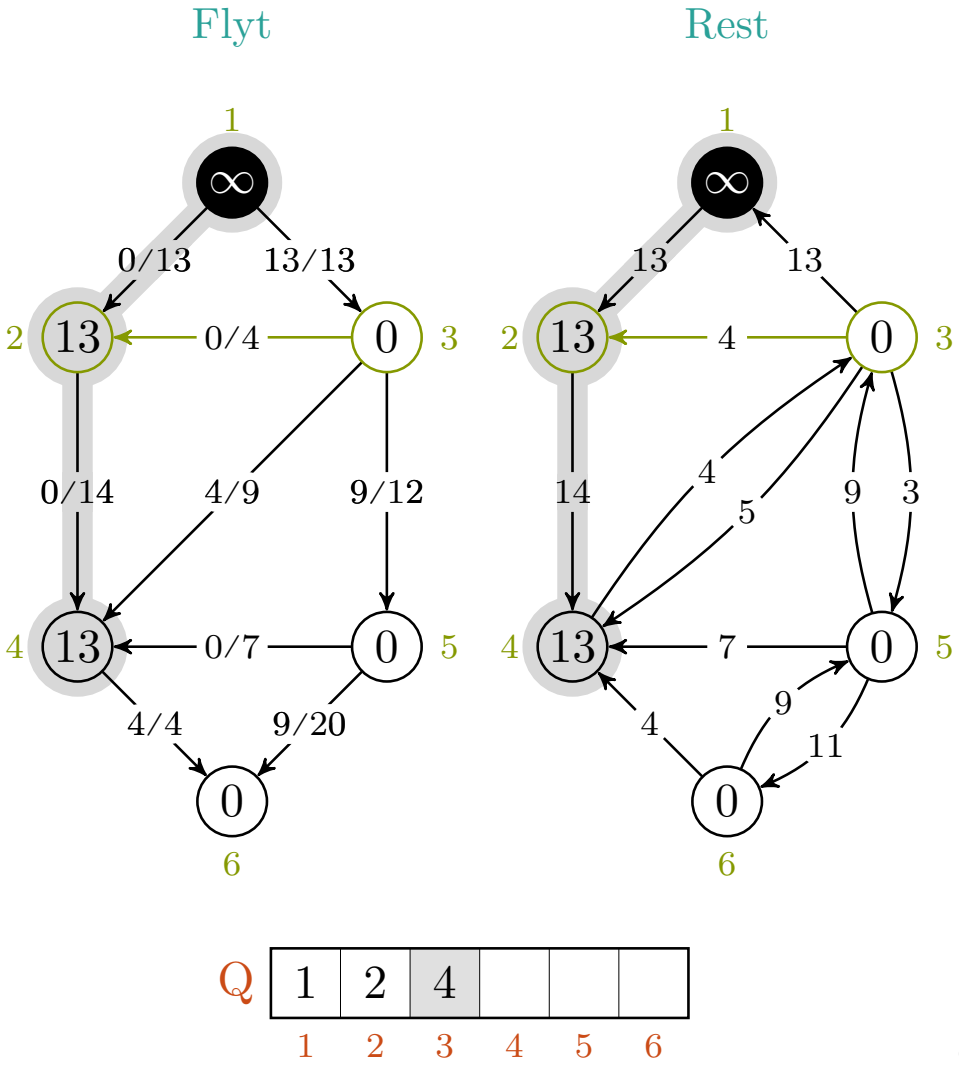




BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

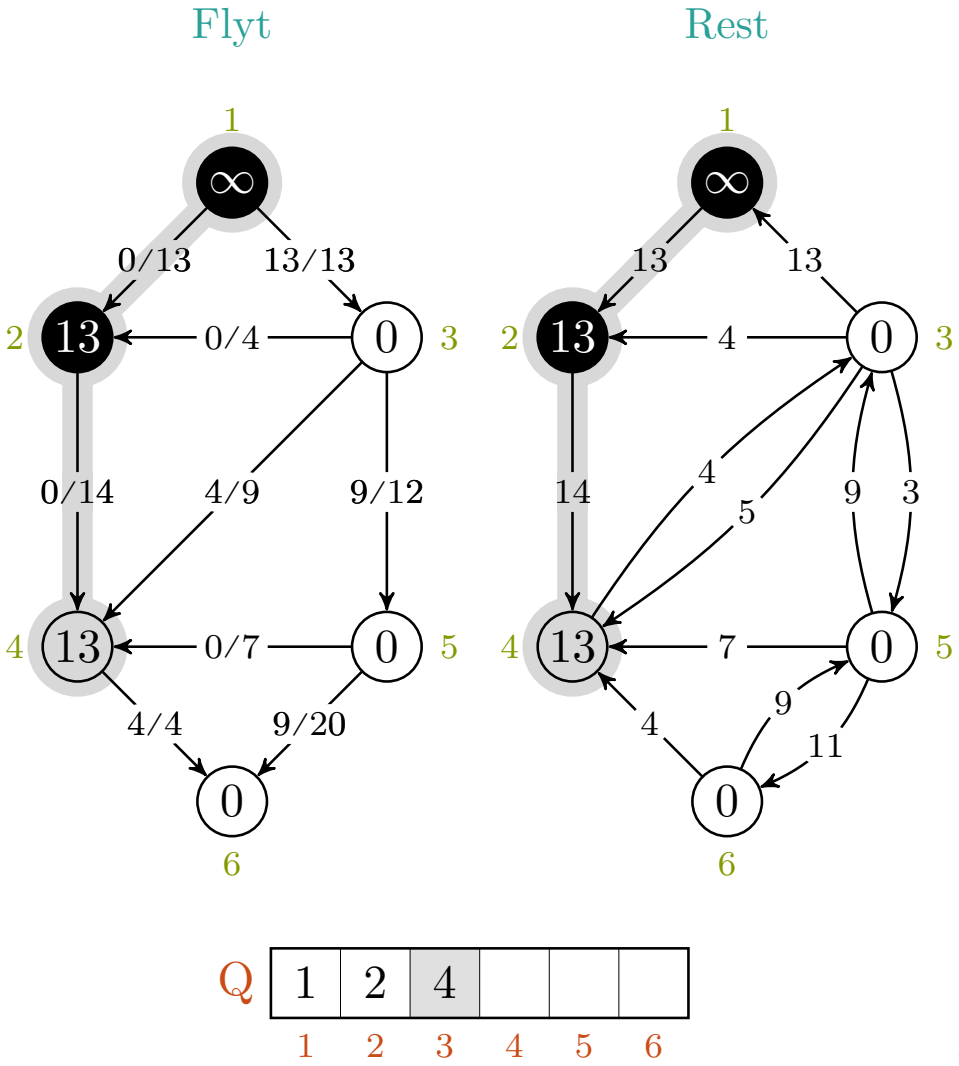
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, 3$



BFS-Labeling(G, s, t)

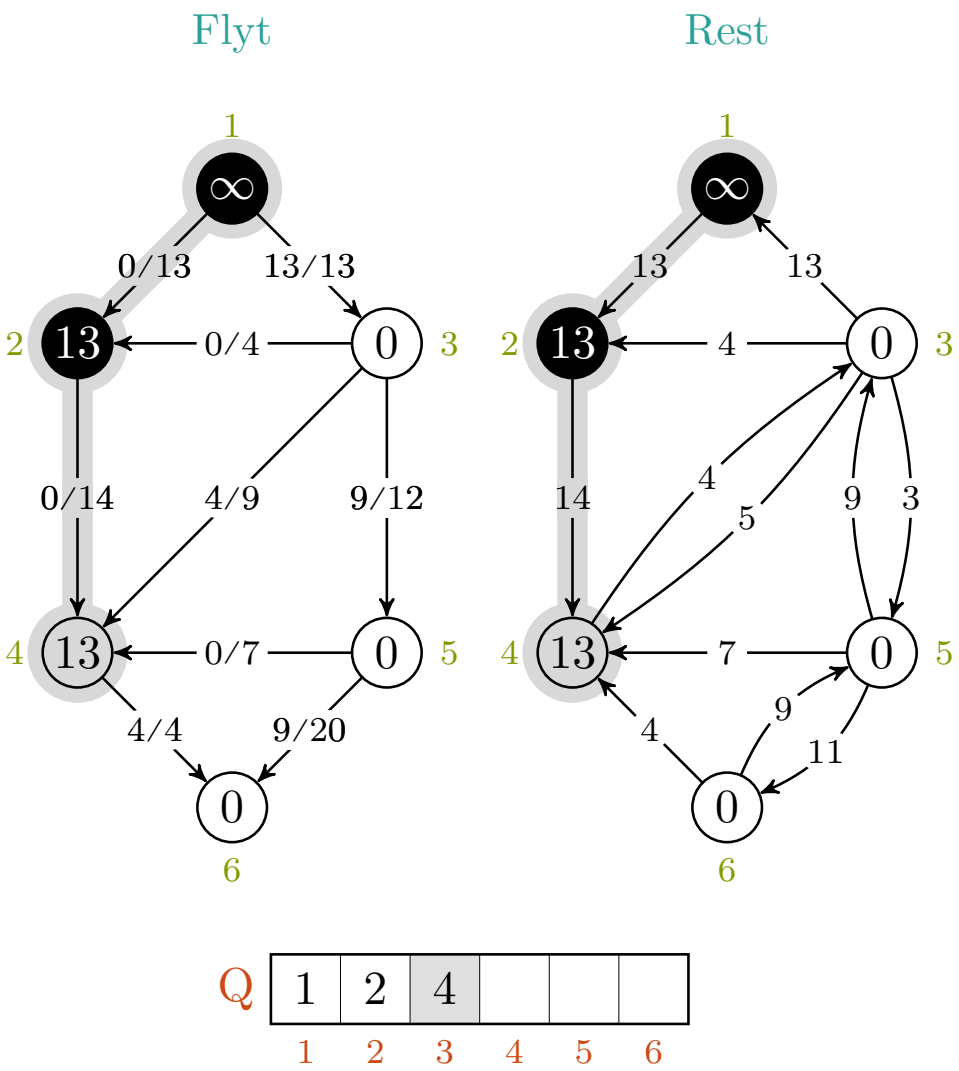
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 2, -$



```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

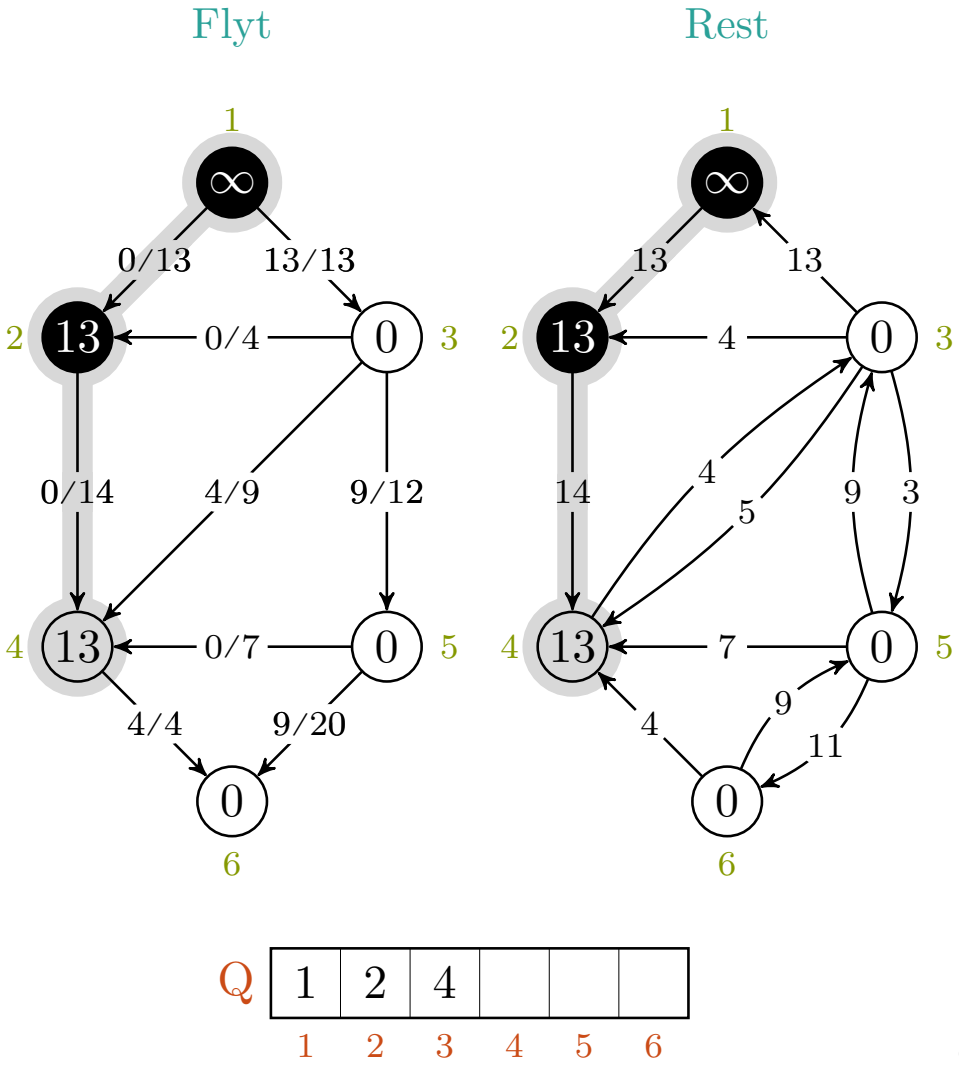
$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 2, -$



BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

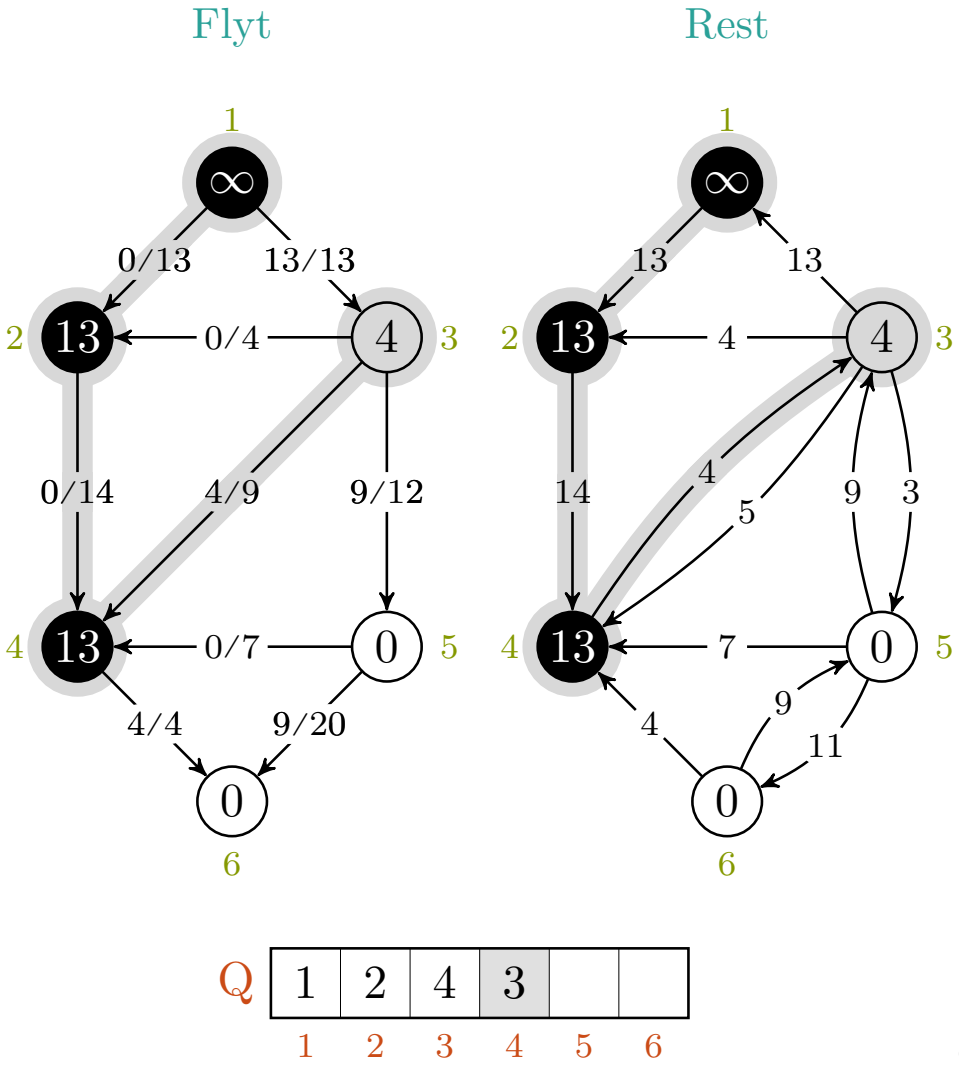
$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 4, -$



BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 4, -$

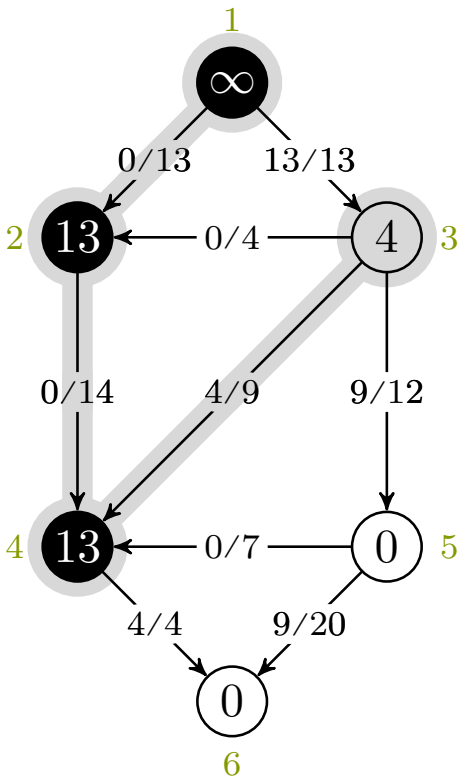


BFS-Labeling(G, s, t)

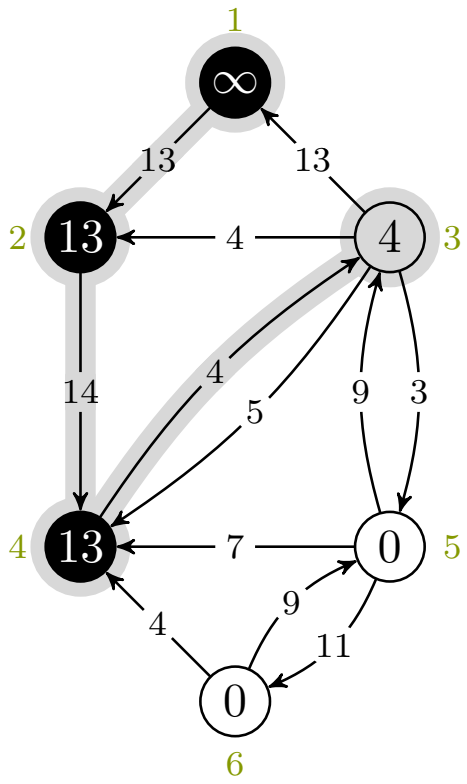
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 4, -$

Flyt



Rest



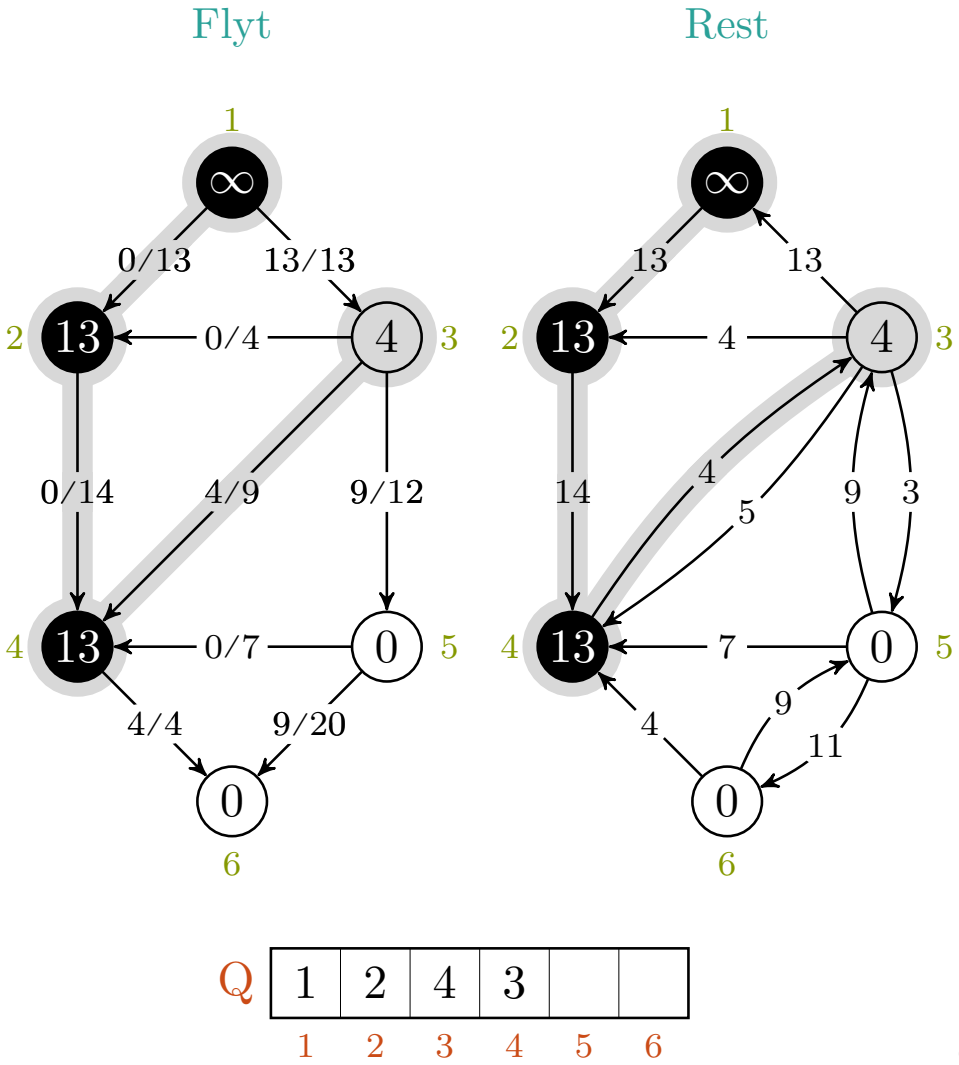
Q

1	2	4	3		
1	2	3	4	5	6

BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

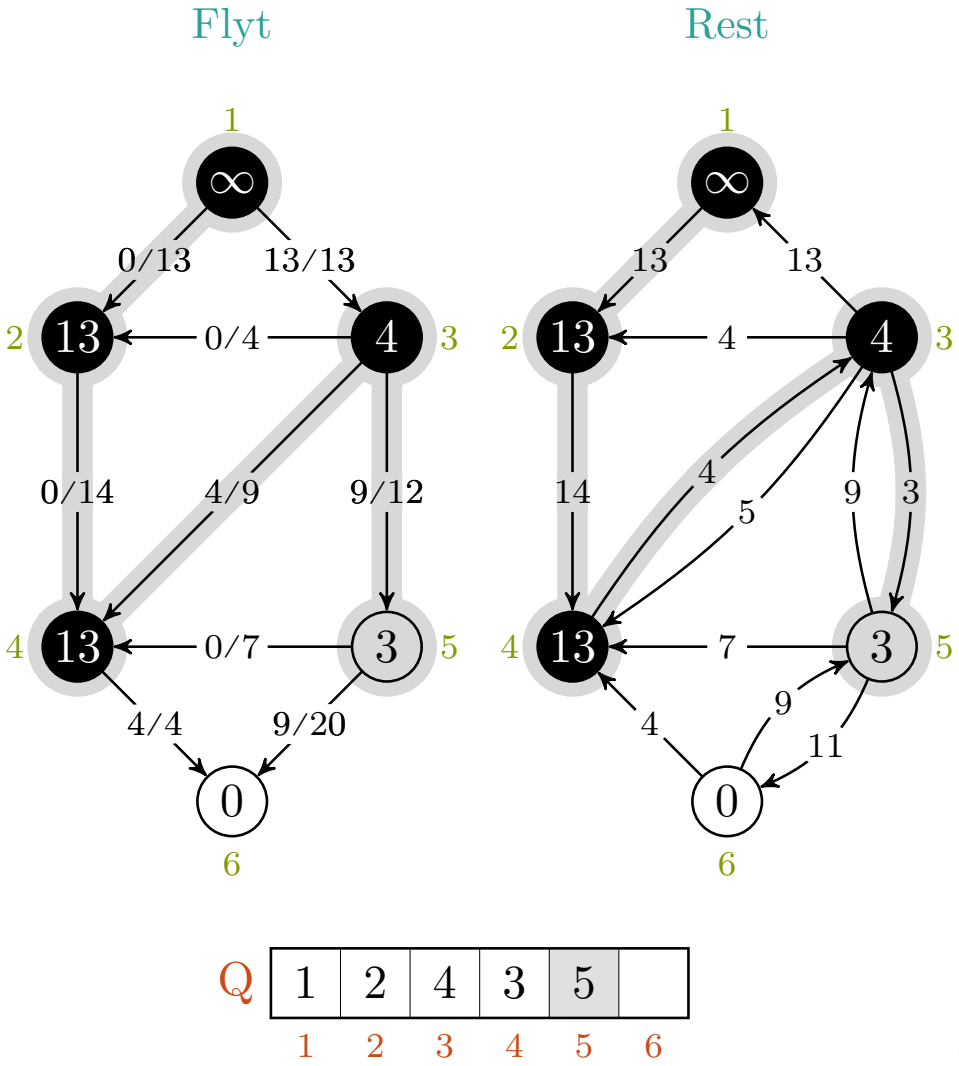
$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 3, -$



BFS-Labeling(G, s, t)

```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

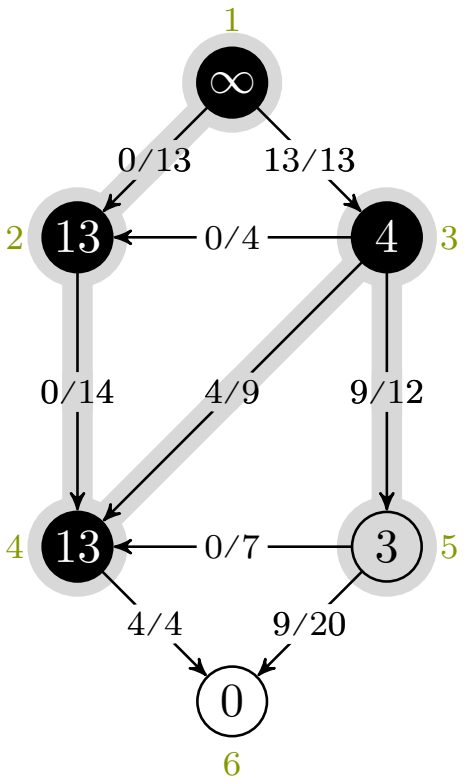
$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 3, -$



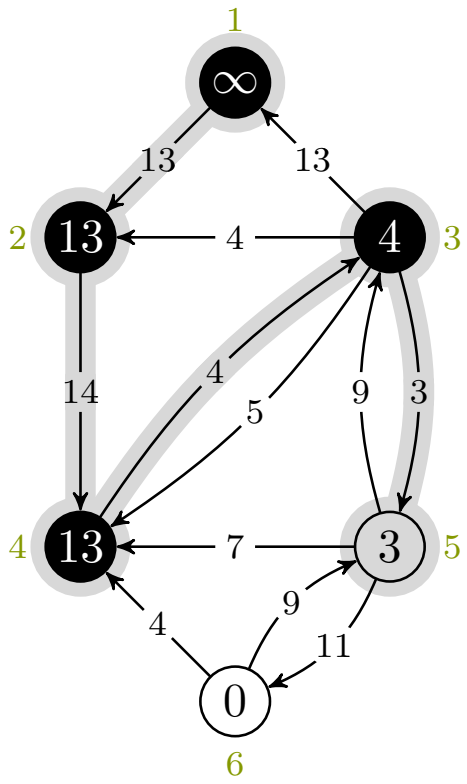
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2       u.f = 0
 3       u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8       u = Dequeue(Q)
 9       for all edges (u, v), (v, u) ∈ G.E
10            if (u, v) ∈ G.E
11                 c_f(u, v) = c(u, v) − (u, v).f
12            else c_f(u, v) = (v, u).f
13            if c_f(u, v) > 0 and v.f == 0
14                 v.f = min(u.f, c_f(u, v))
15                 v.π = u
16                 Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rightarrow\; -, 3, -$

Flyt



Rest



Q

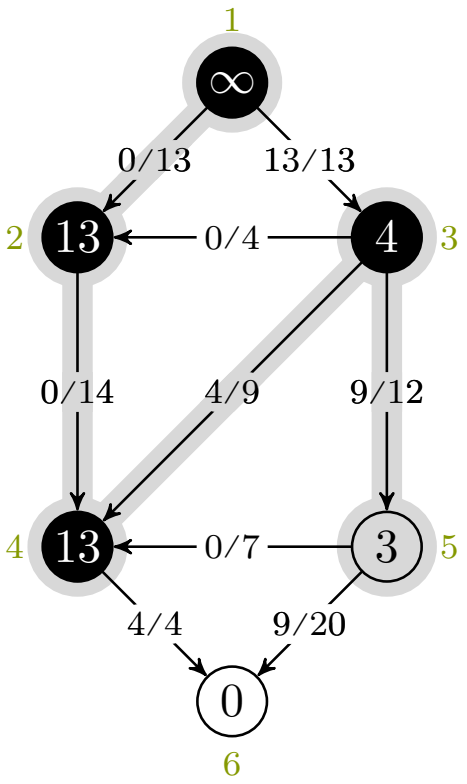
1	2	4	3	5	
1	2	3	4	5	6

BFS-Labeling(G, s, t)

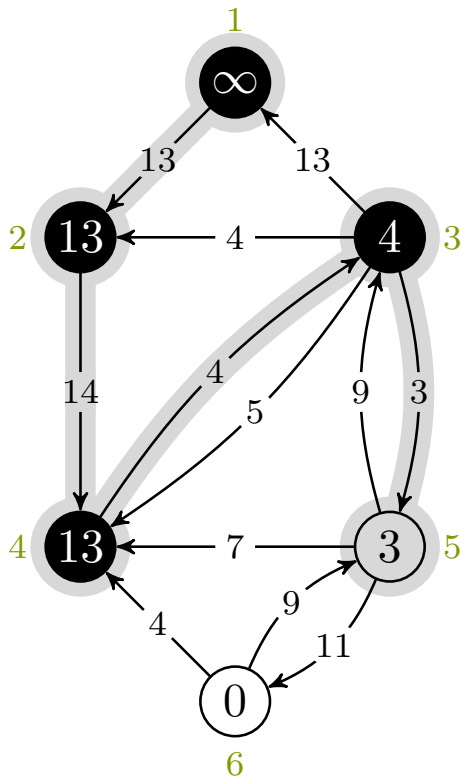
```
 1  for each vertex u ∈ G.V
 2      u.f = 0
 3      u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8      u = Dequeue(Q)
 9      for all edges (u, v), (v, u) ∈ G.E
10          if (u, v) ∈ G.E
11              c_f(u, v) = c(u, v) − (u, v).f
12          else c_f(u, v) = (v, u).f
13          if c_f(u, v) > 0 and v.f == 0
14              v.f = min(u.f, c_f(u, v))
15              v.π = u
16              Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \,\rangle\, -, 5, -$

Flyt



Rest



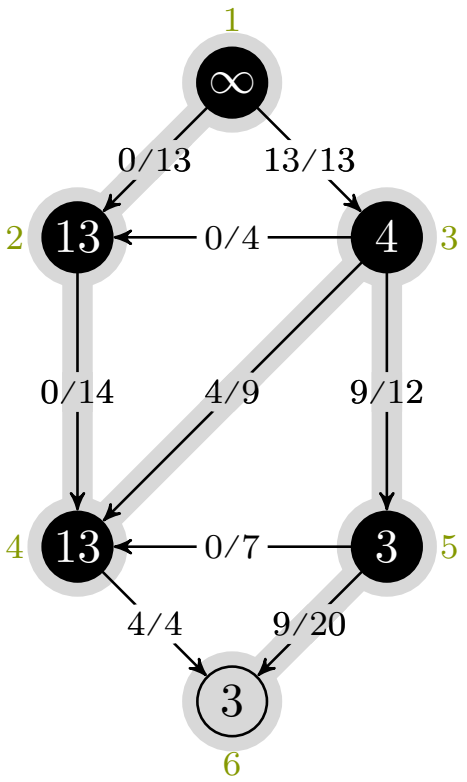
Q	1	2	4	3	5	
	1	2	3	4	5	6

BFS-Labeling(G, s, t)

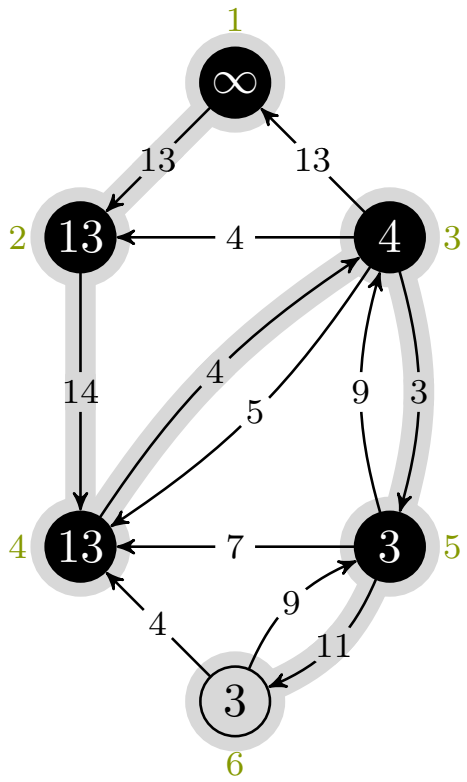
```
1  for each vertex u ∈ G.V
2      u.f = 0
3      u.π = NIL
4  s.f = ∞
5  Q = ∅
6  Enqueue(Q, s)
7  while Q ≠ ∅ and t.f == 0
8      u = Dequeue(Q)
9      for all edges (u, v), (v, u) ∈ G.E
10         if (u, v) ∈ G.E
11             c_f(u, v) = c(u, v) − (u, v).f
12         else c_f(u, v) = (v, u).f
13         if c_f(u, v) > 0 and v.f == 0
14             v.f = min(u.f, c_f(u, v))
15             v.π = u
16             Enqueue(Q, v)
17 return t.f ≠ 0
```

$c_f(p), u, v \;=\; -, -, - \;\rangle\; -, 5, -$

Flyt



Rest

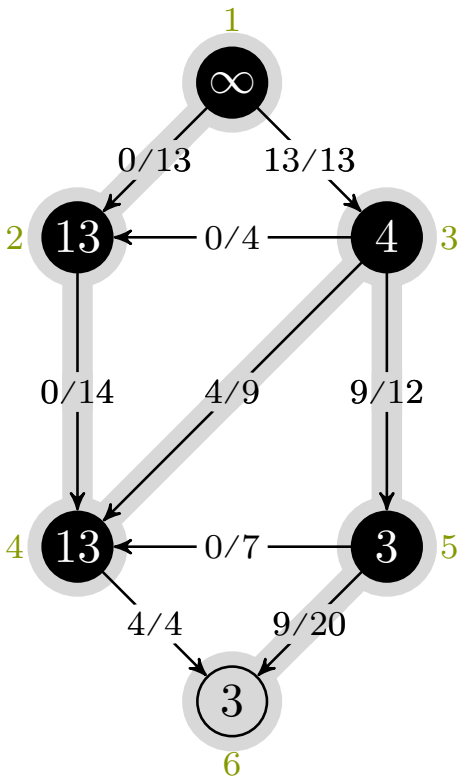


Q	1	2	4	3	5	6
	1	2	3	4	5	6

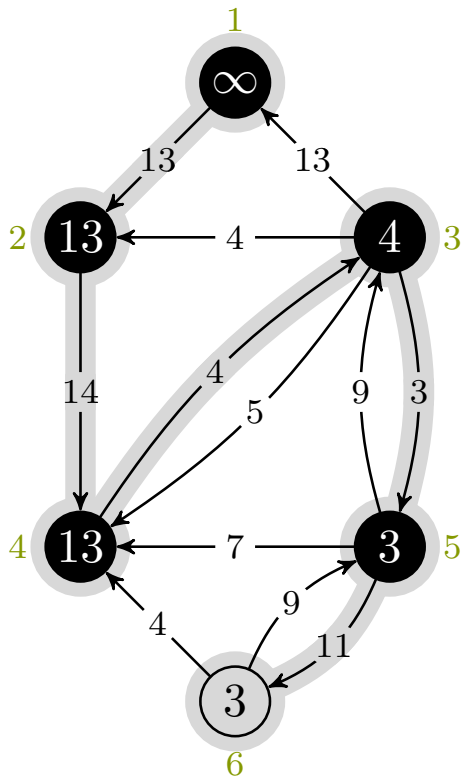
```
BFS-Labeling(G, s, t)
 1  for each vertex u ∈ G.V
 2        u.f = 0
 3        u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
16                    Enqueue(Q, v)
17  return t.f ≠ 0
```

$c_f(p), u, v \;=\; -,-,- \,\rangle\, -,-,-$

Flyt



Rest



Q	1	2	4	3	5	6
	1	2	3	4	5	6

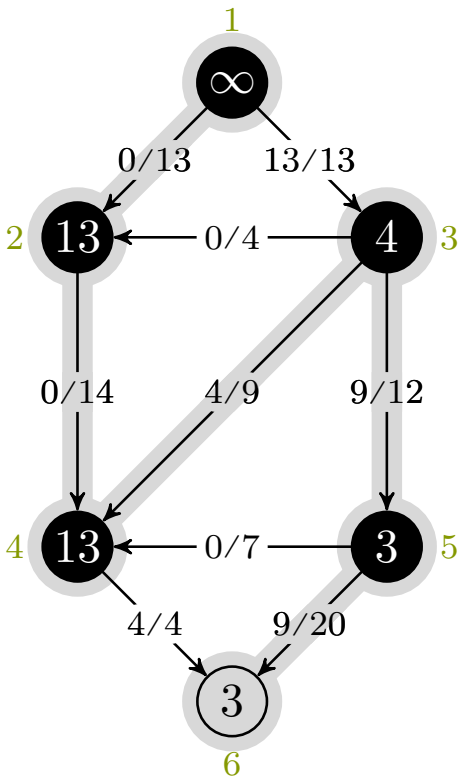
BFS-Labeling(G, s, t)

```
 1  for each vertex u ∈ G.V
 2        u.f = 0
 3        u.π = NIL
 4  s.f = ∞
 5  Q = ∅
 6  Enqueue(Q, s)
 7  while Q ≠ ∅ and t.f == 0
 8        u = Dequeue(Q)
 9        for all edges (u, v), (v, u) ∈ G.E
10              if (u, v) ∈ G.E
11                    c_f(u, v) = c(u, v) − (u, v).f
12              else c_f(u, v) = (v, u).f
13              if c_f(u, v) > 0 and v.f == 0
14                    v.f = min(u.f, c_f(u, v))
15                    v.π = u
16                    Enqueue(Q, v)
17  return t.f ≠ 0
```

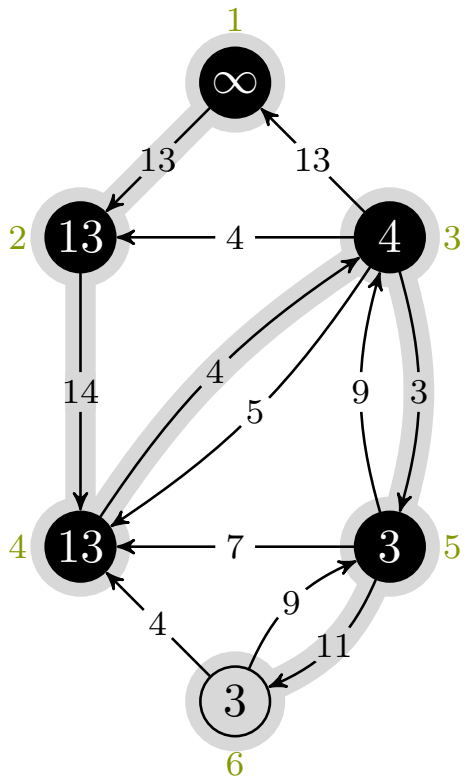
→ TRUE

$c_f(p), u, v$ = –, –, – › –, –, –

Flyt



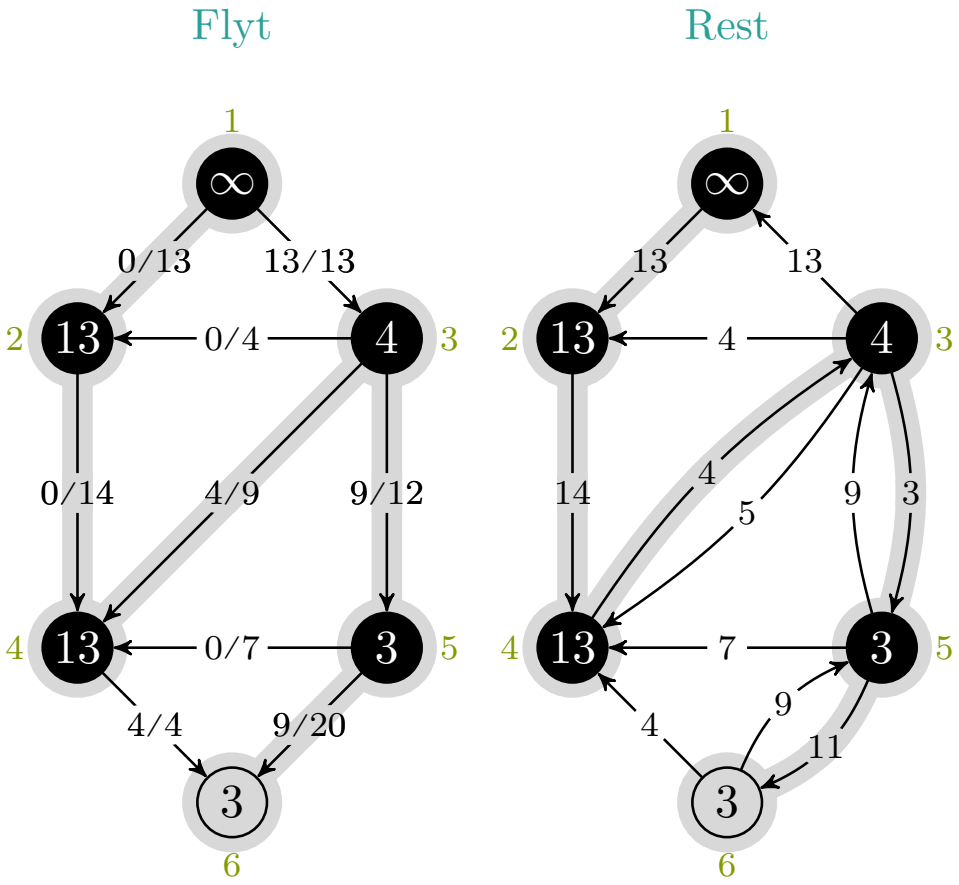
Rest



Q	1	2	4	3	5	6
	1	2	3	4	5	6

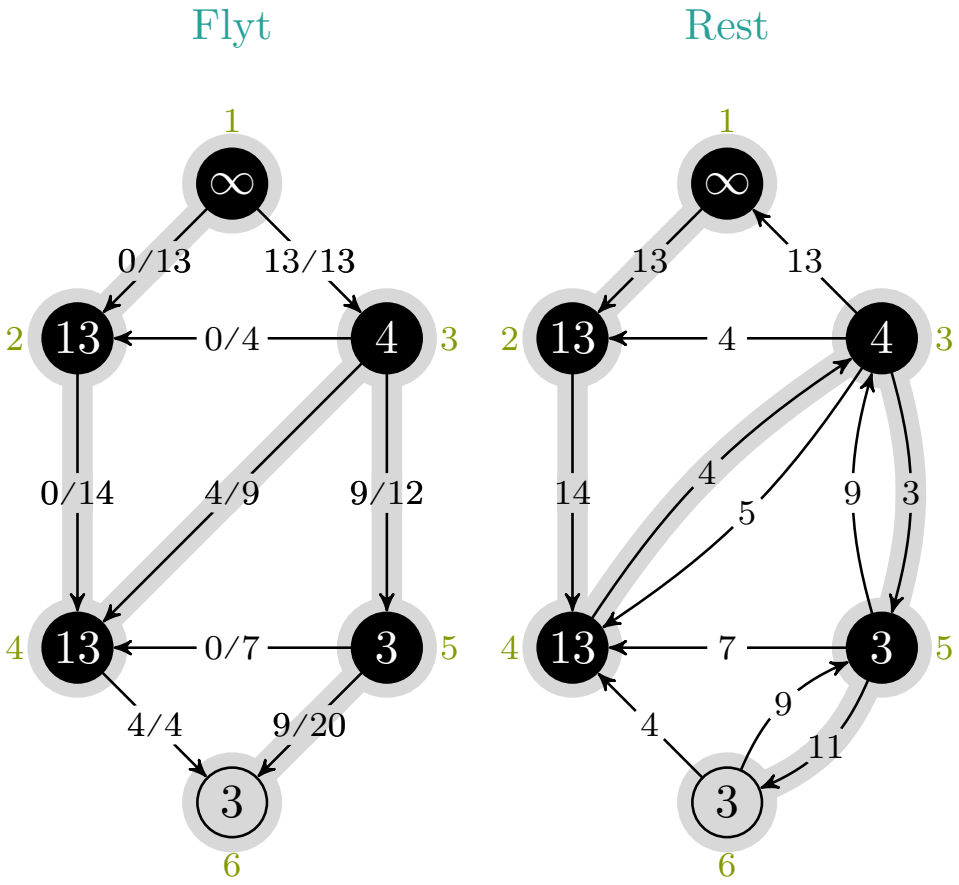
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$



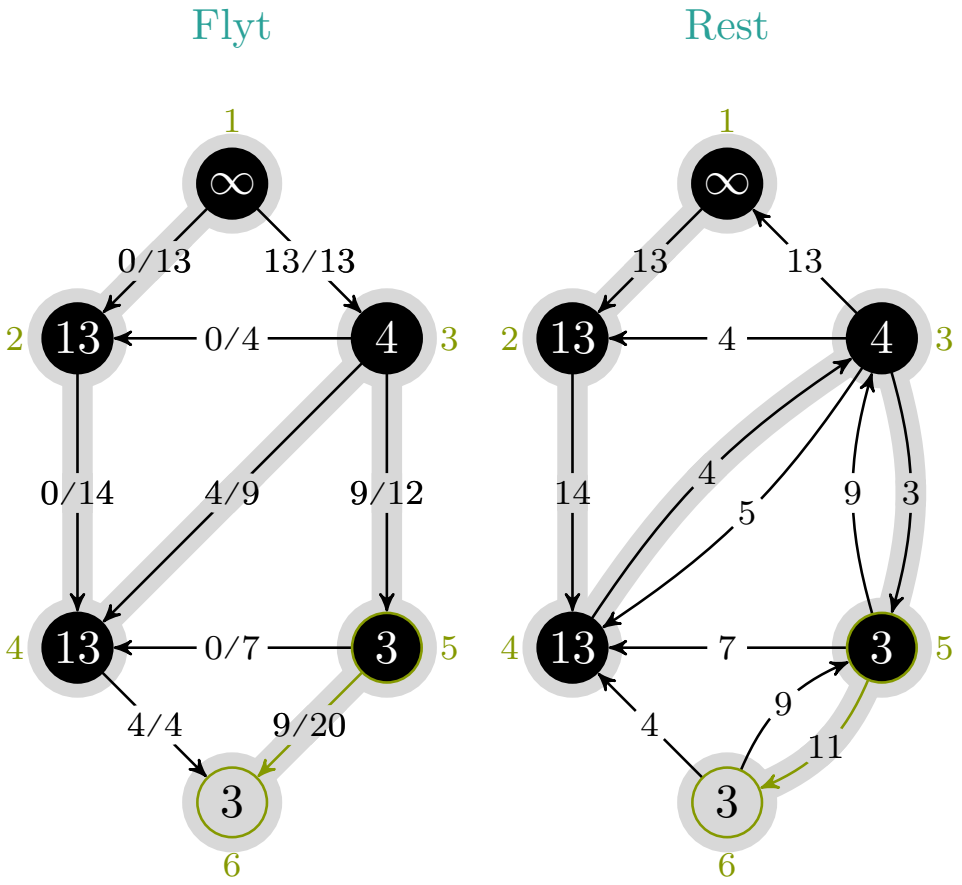
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, -, -$



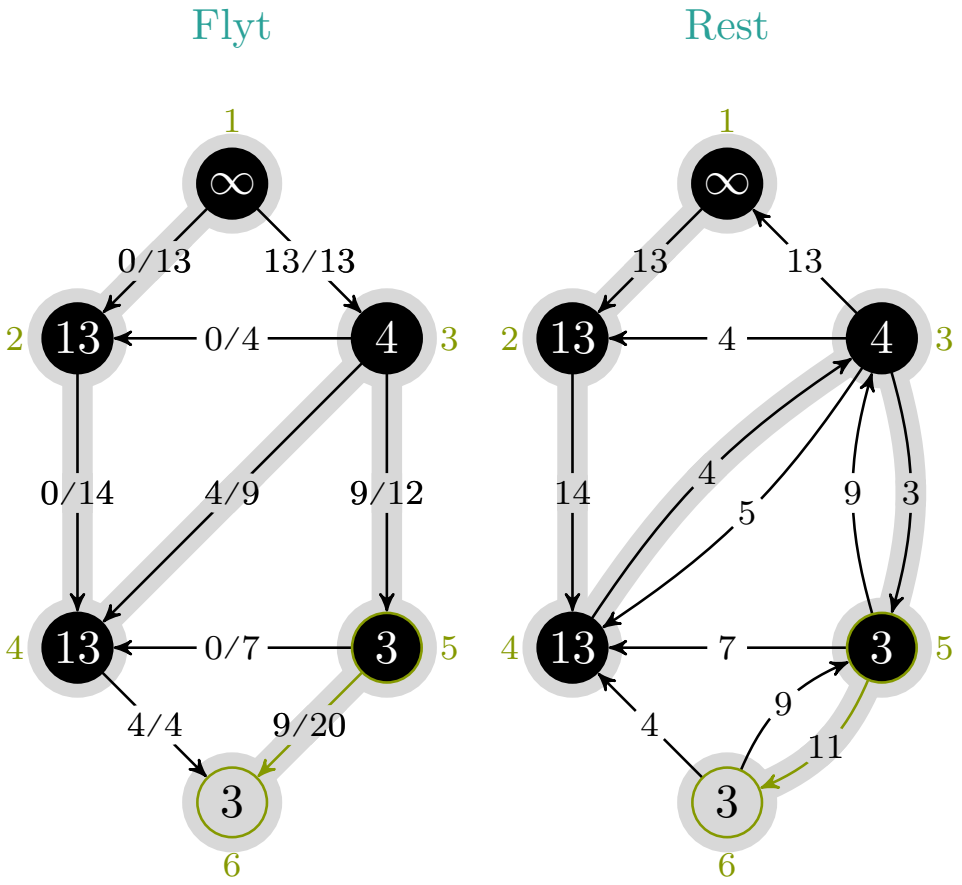
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 5, 6$



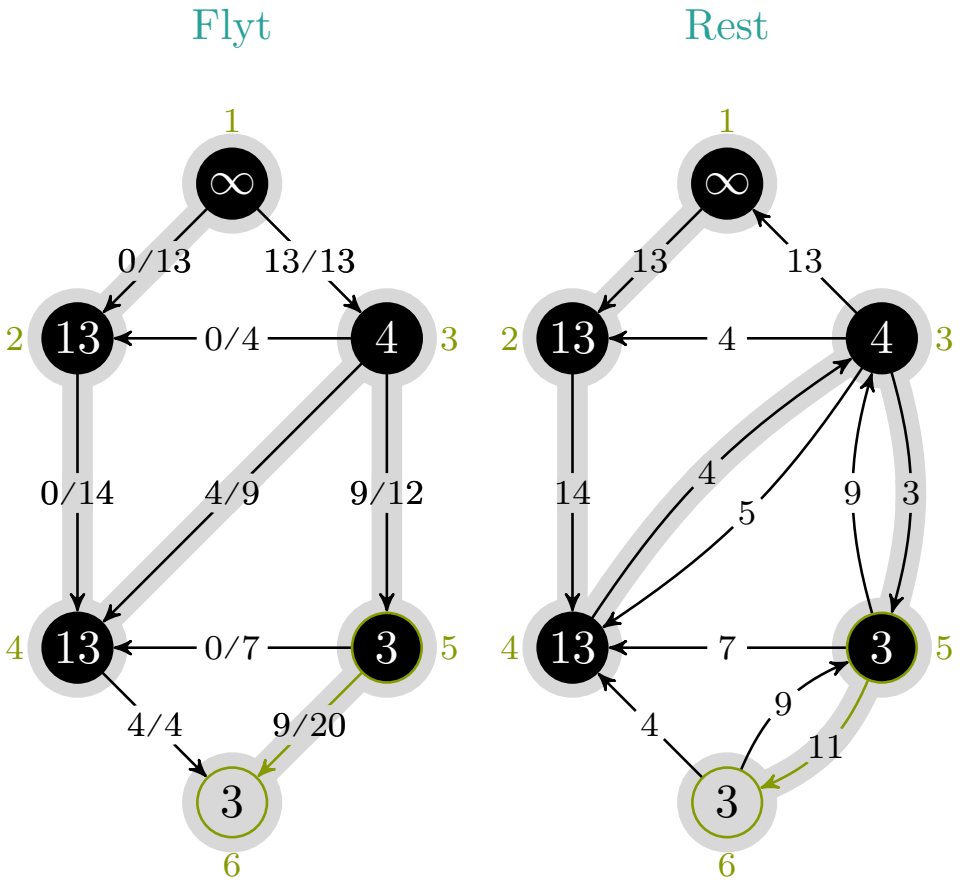
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 5, 6$



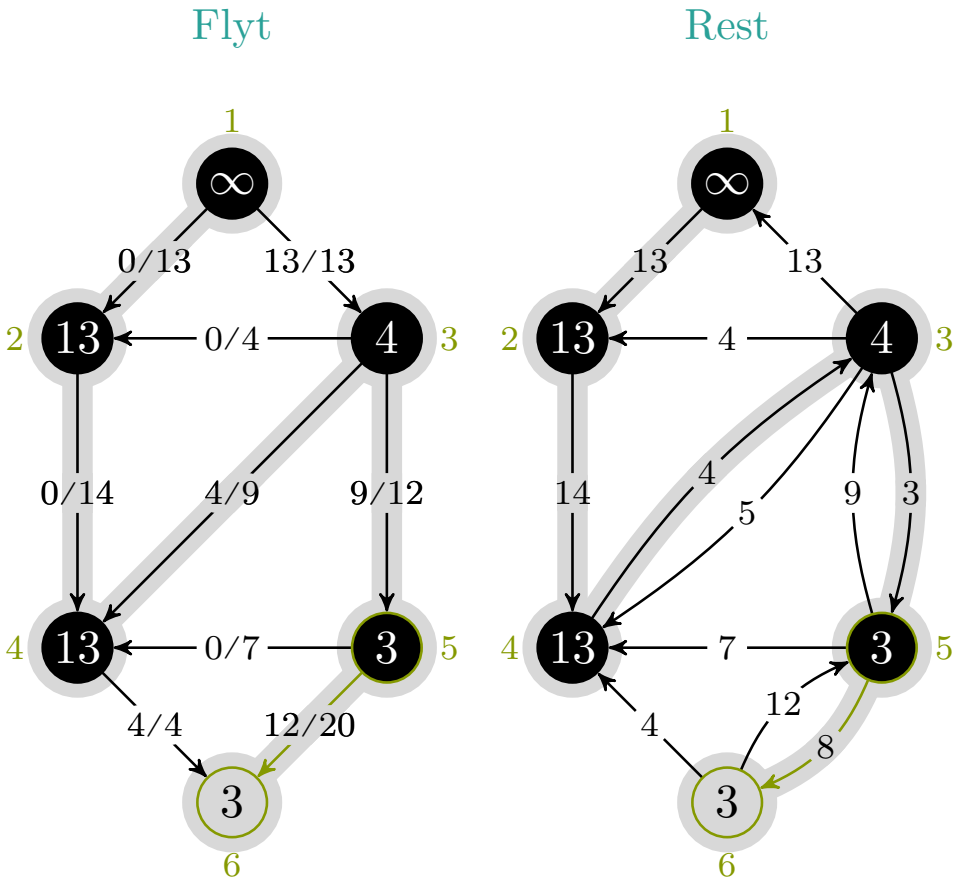
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 5, 6$



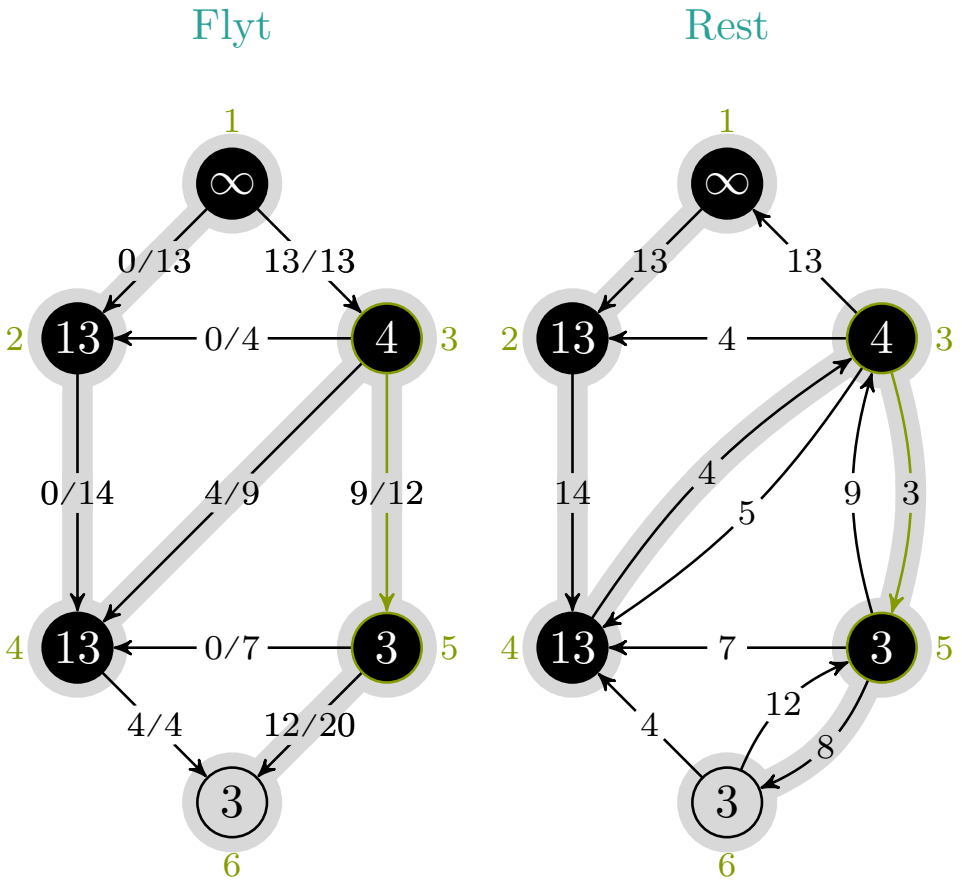
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 5, 6$



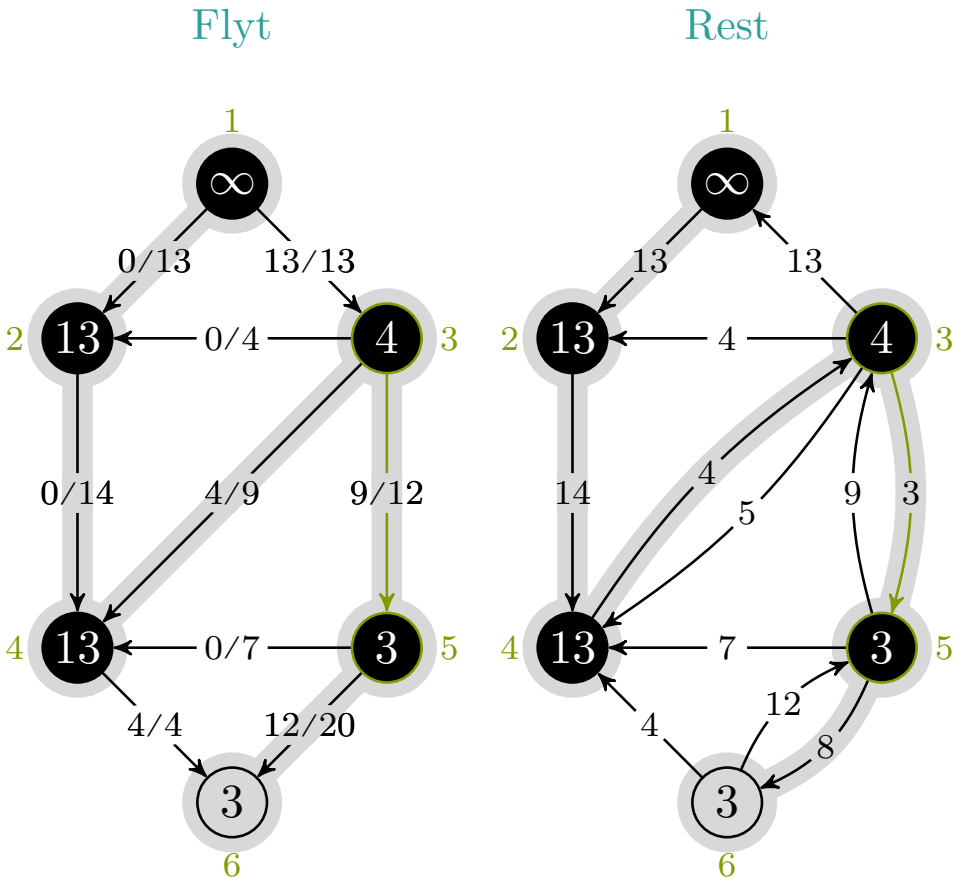
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 3, 5$



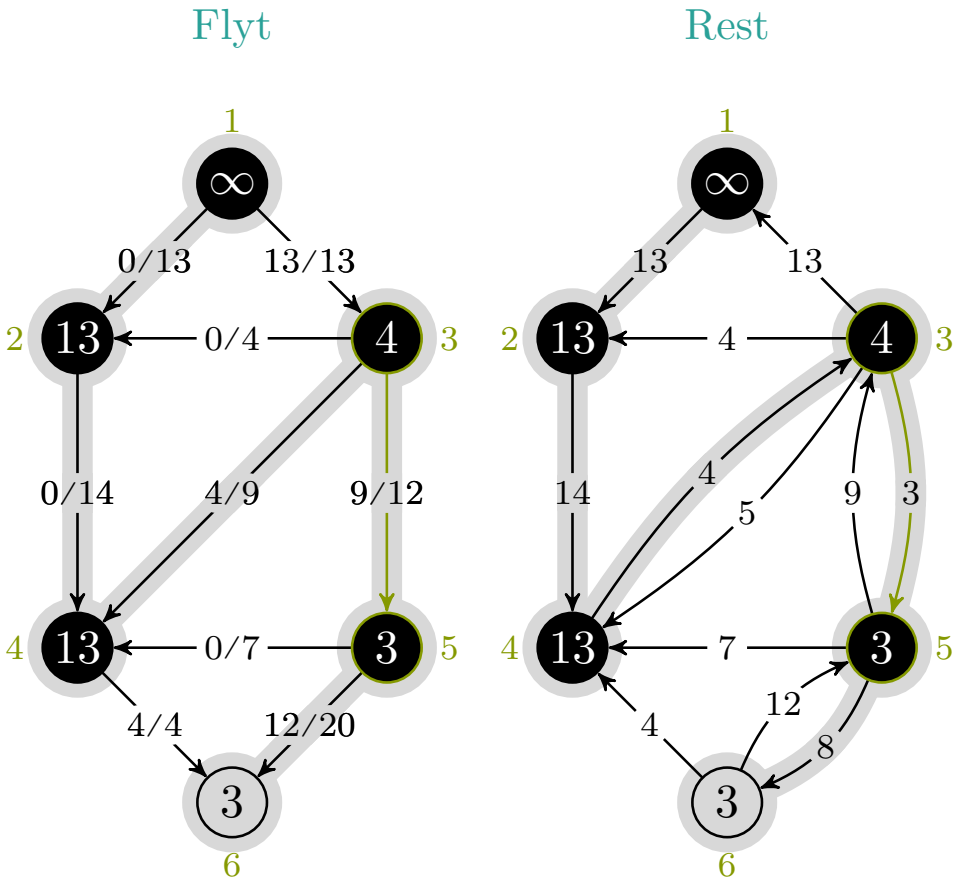
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 3, 5$



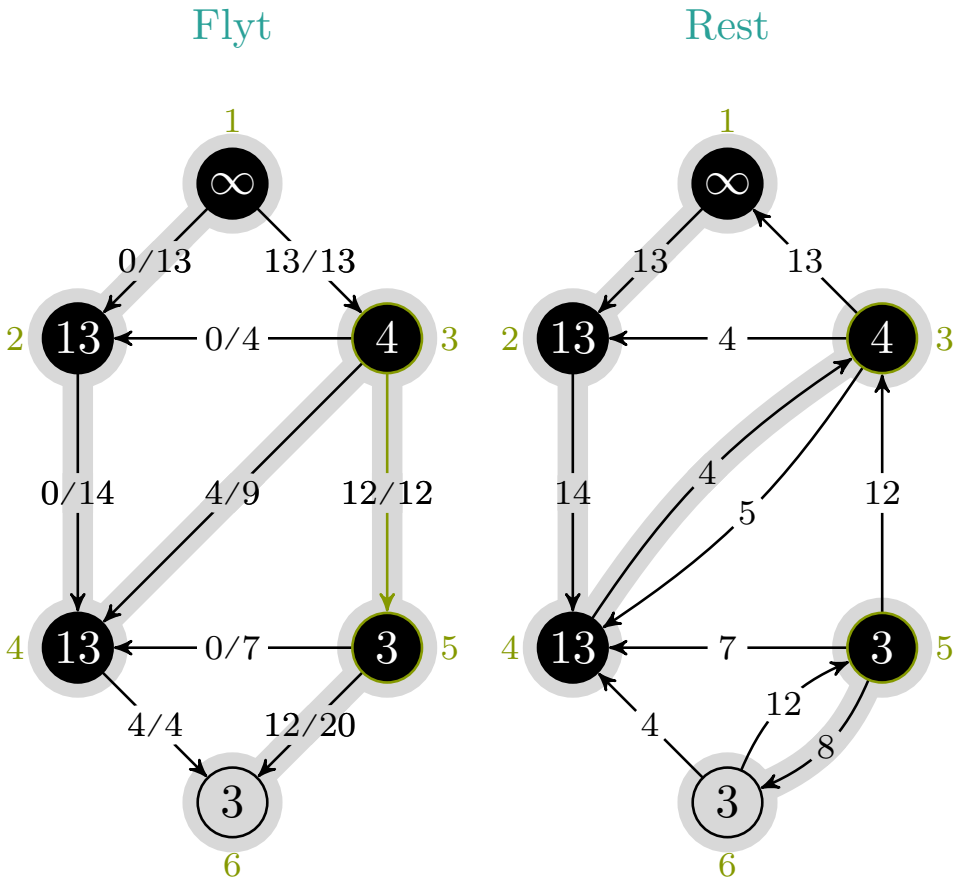
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 3, 5$



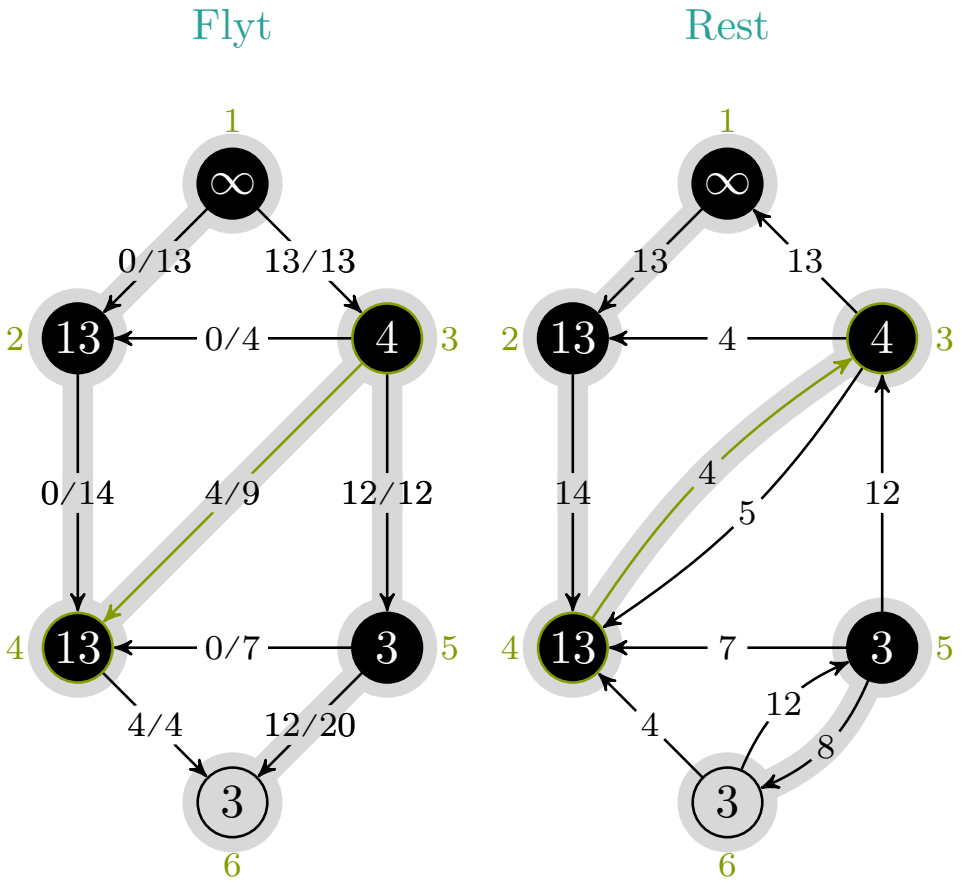
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 3, 5$



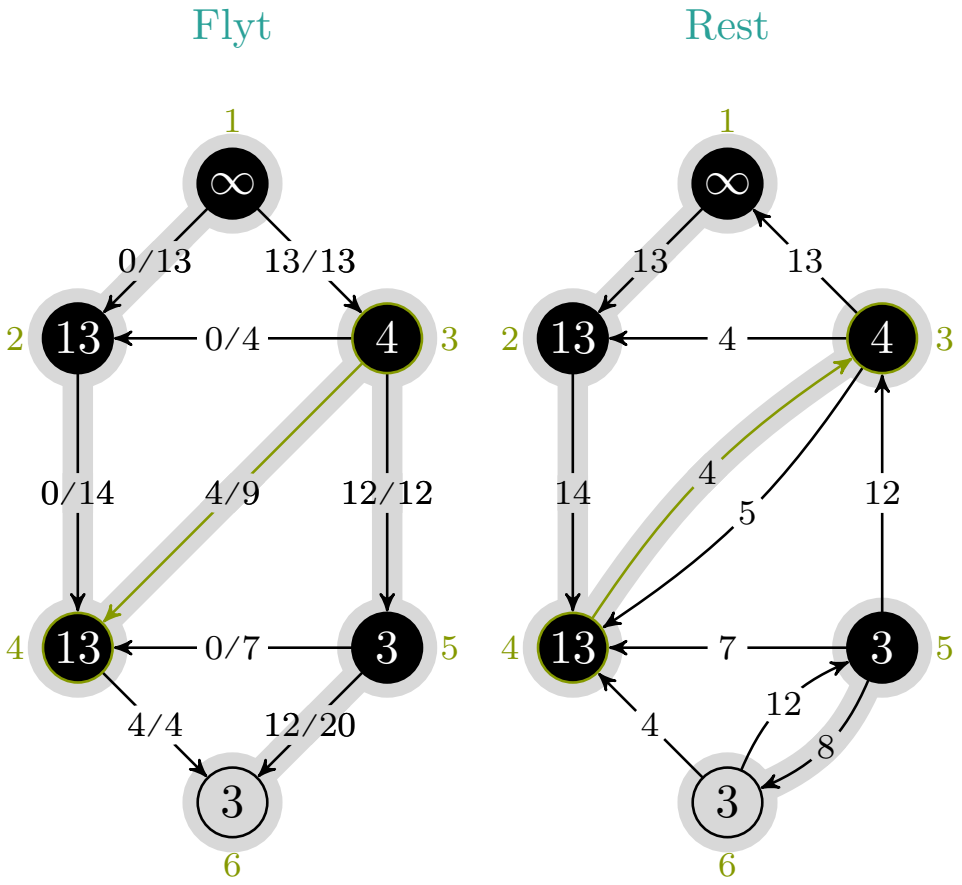
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 4, 3$



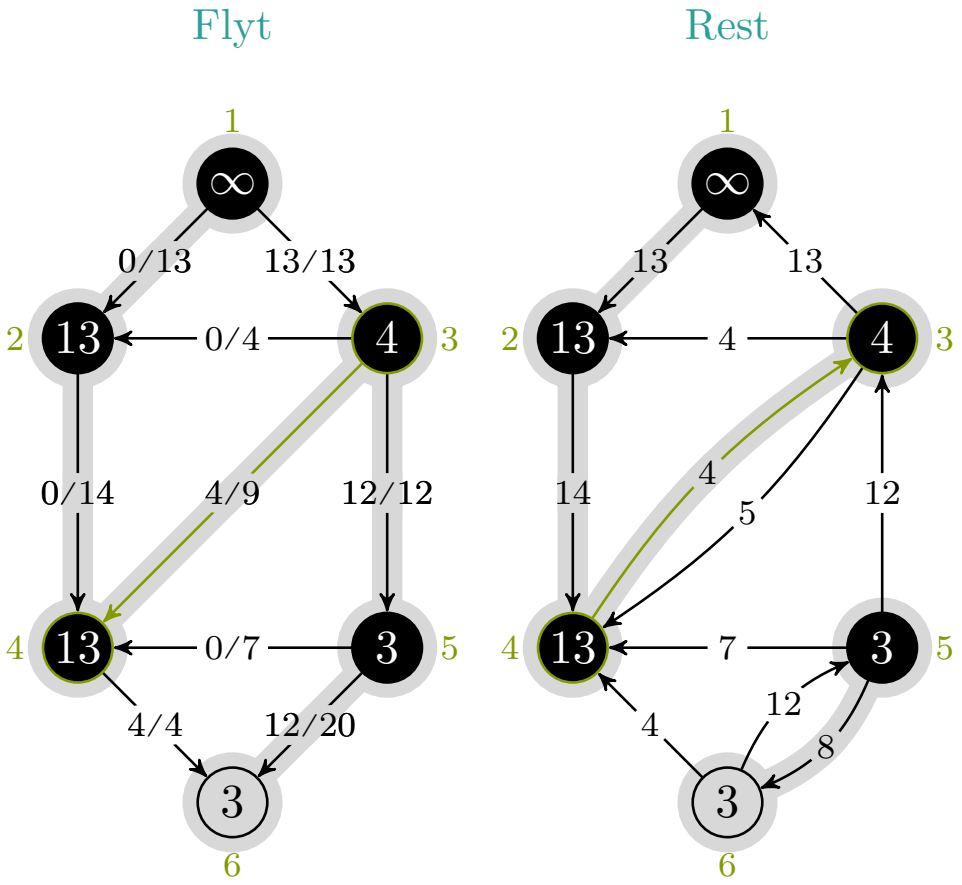
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 4, 3$



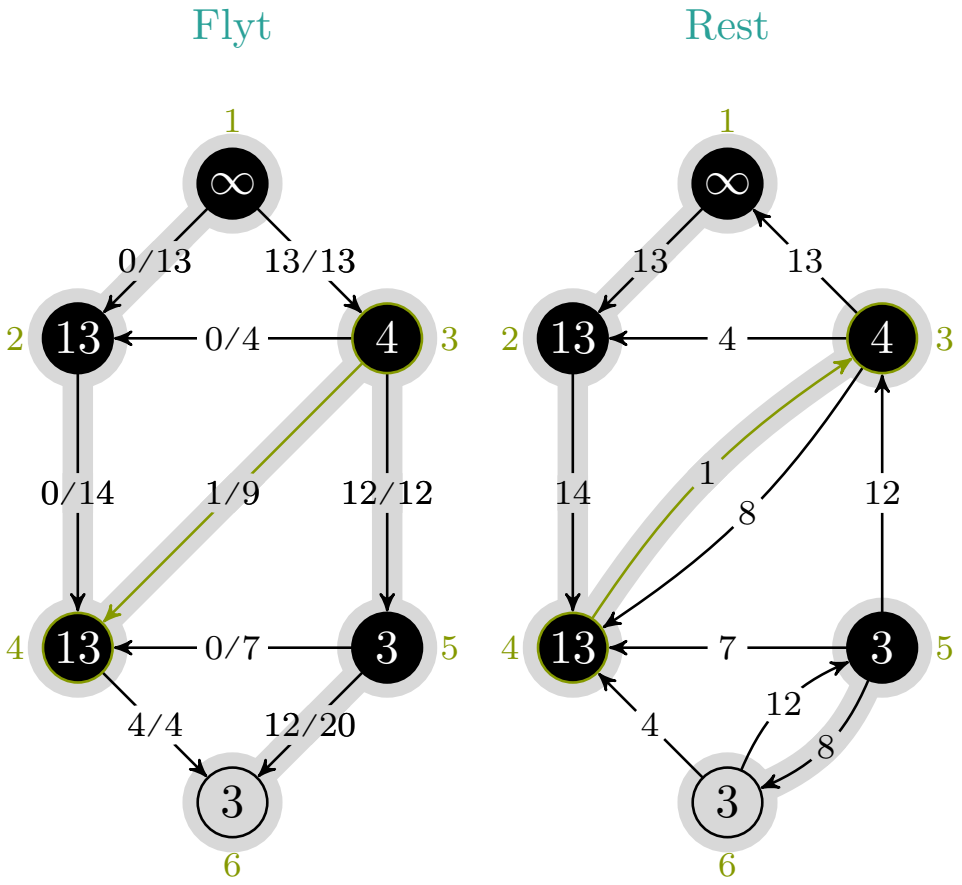
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 4, 3$



```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

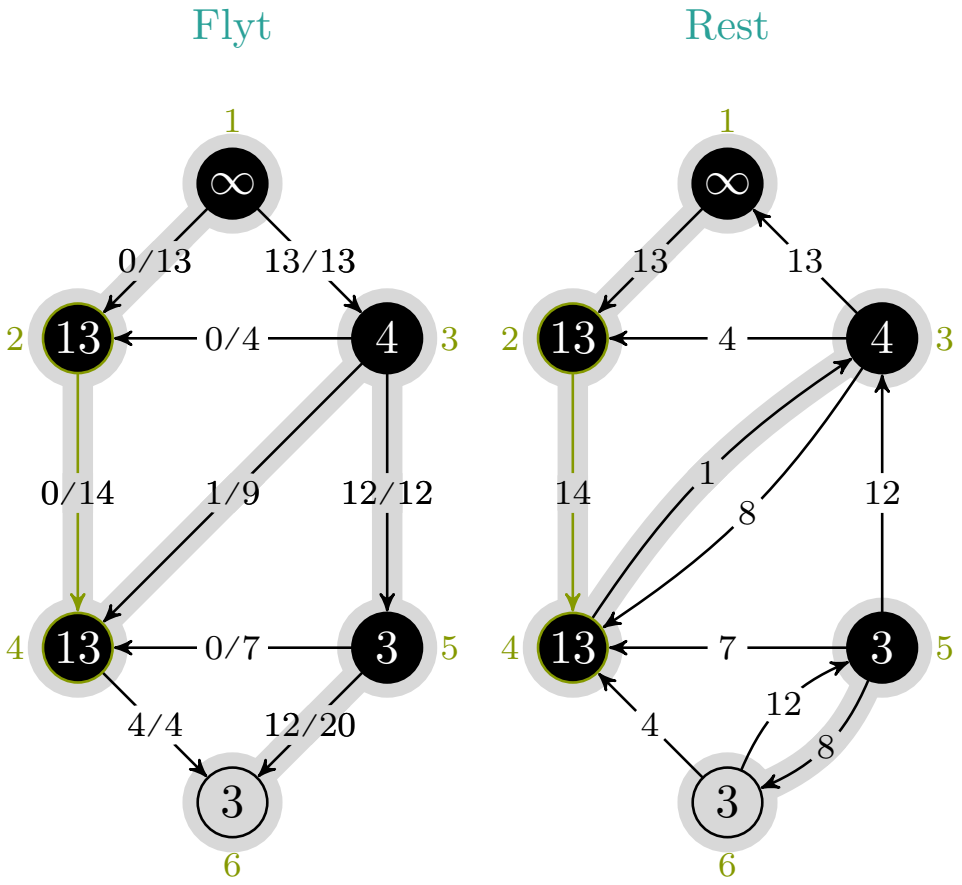
$c_f(p), u, v \;=\; 3, 4, 3$



EDMONDS-KARP(G, s, t)

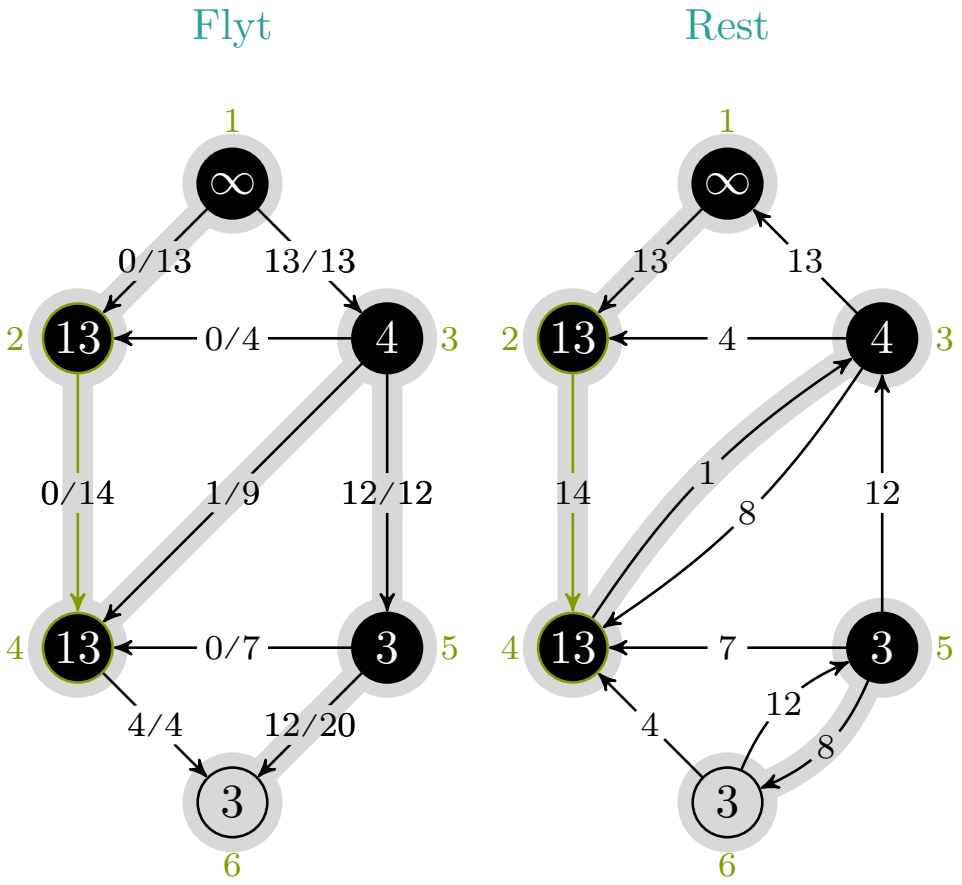
```
1  for each edge (u, v) ∈ G.E
2      (u, v).f = 0
3  while BFS-LABELING(G, s, t)
4      c_f(p) = t.f
5      u, v = t.π, t
6      while u ≠ NIL
7          if (u, v) ∈ G.E
8              (u, v).f = (u, v).f + c_f(p)
9          else (v, u).f = (v, u).f − c_f(p)
10         u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 2, 4$



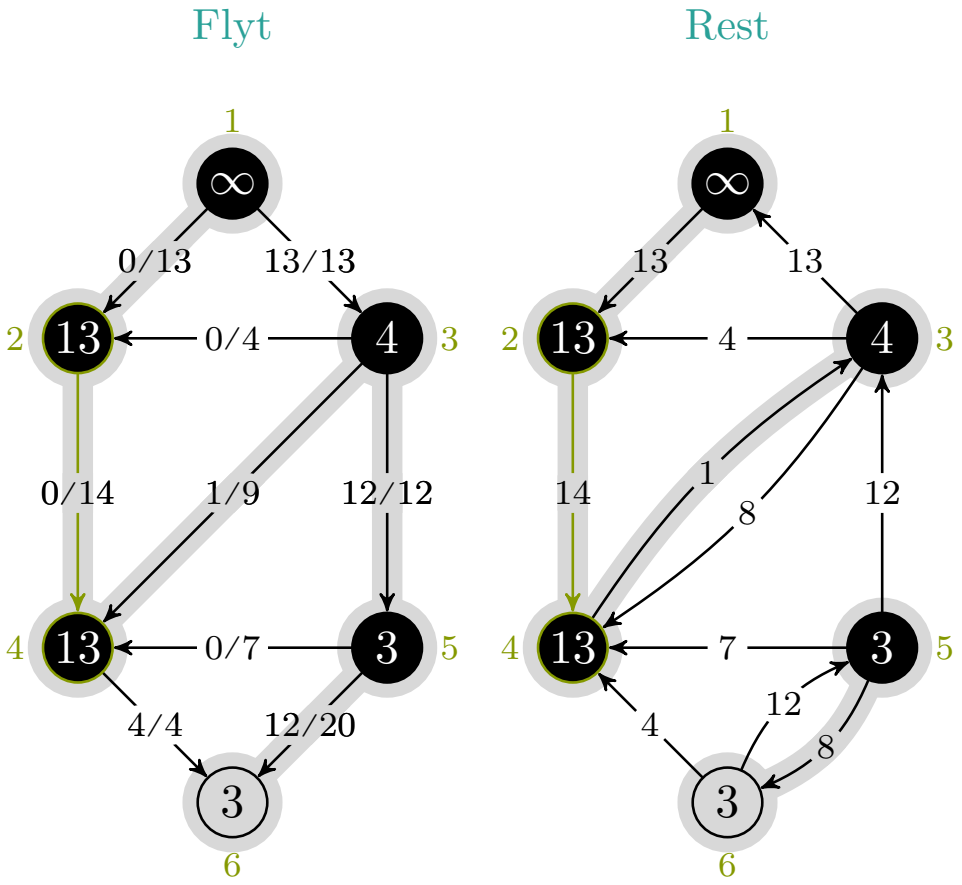
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 2, 4$



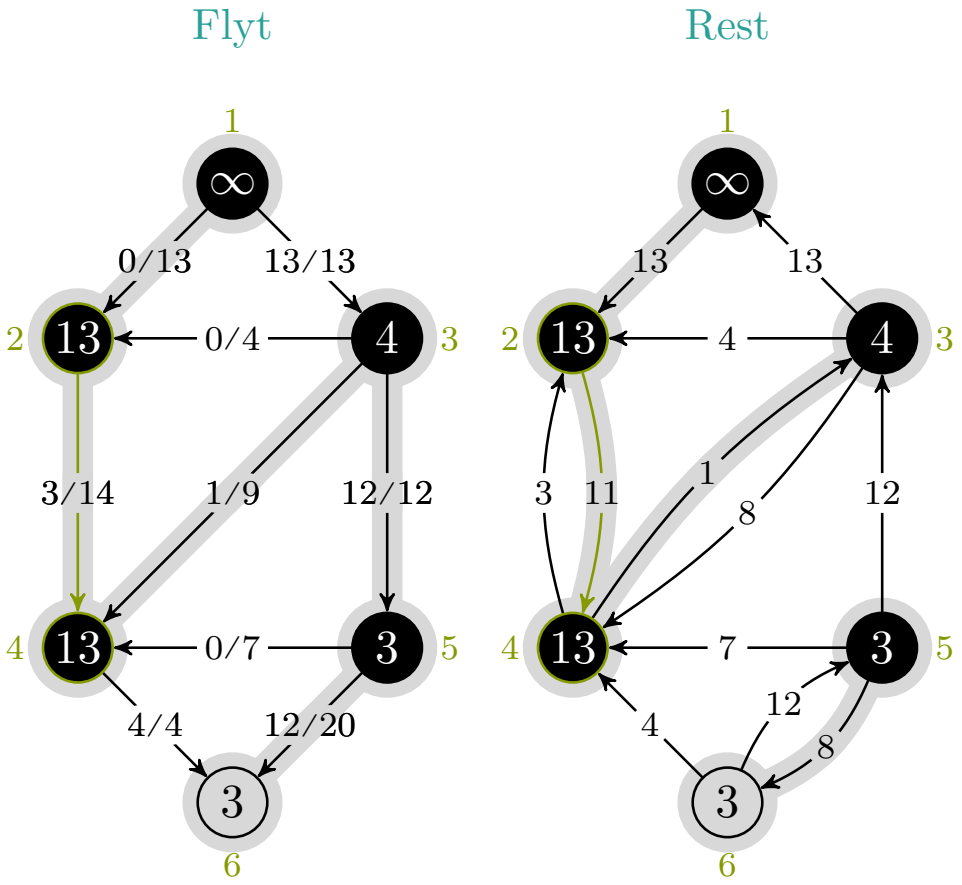
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 2, 4$



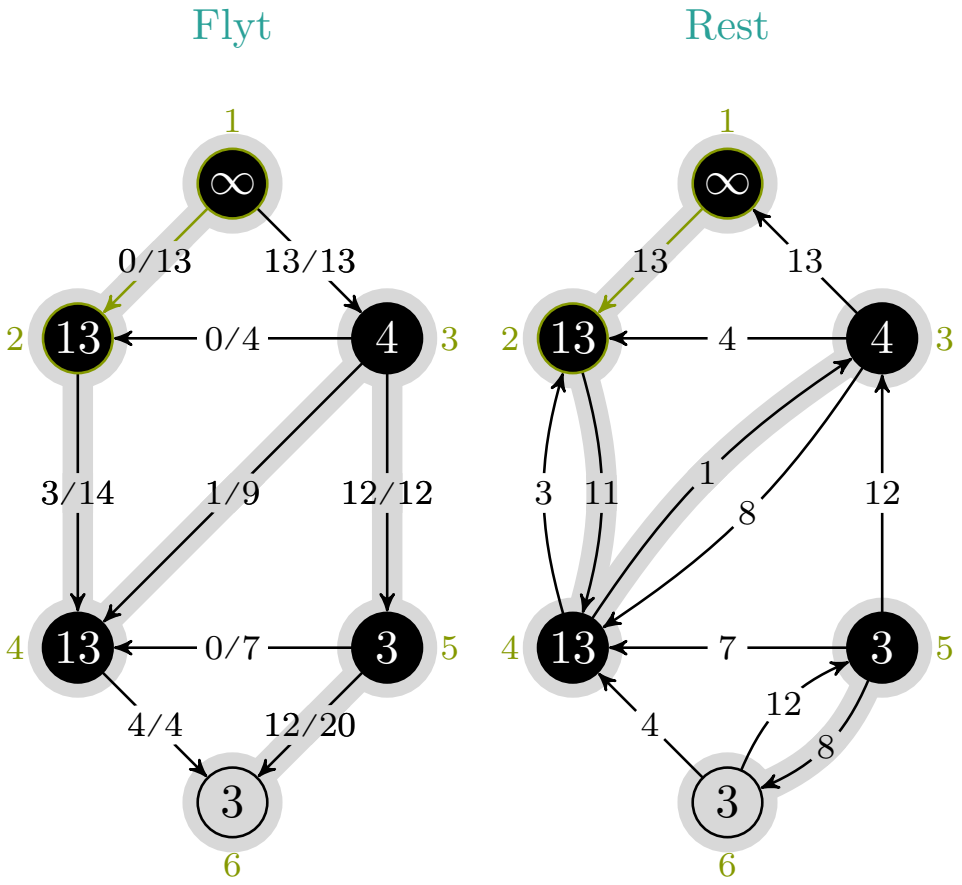
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 2, 4$



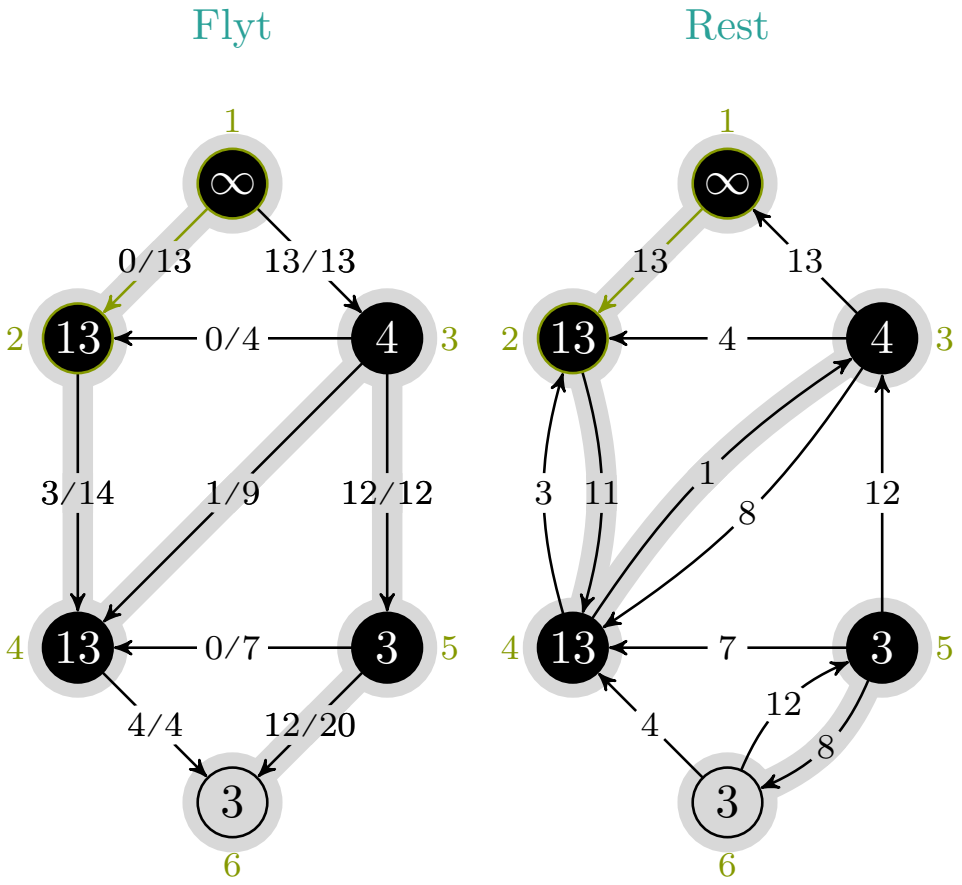
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 1, 2$



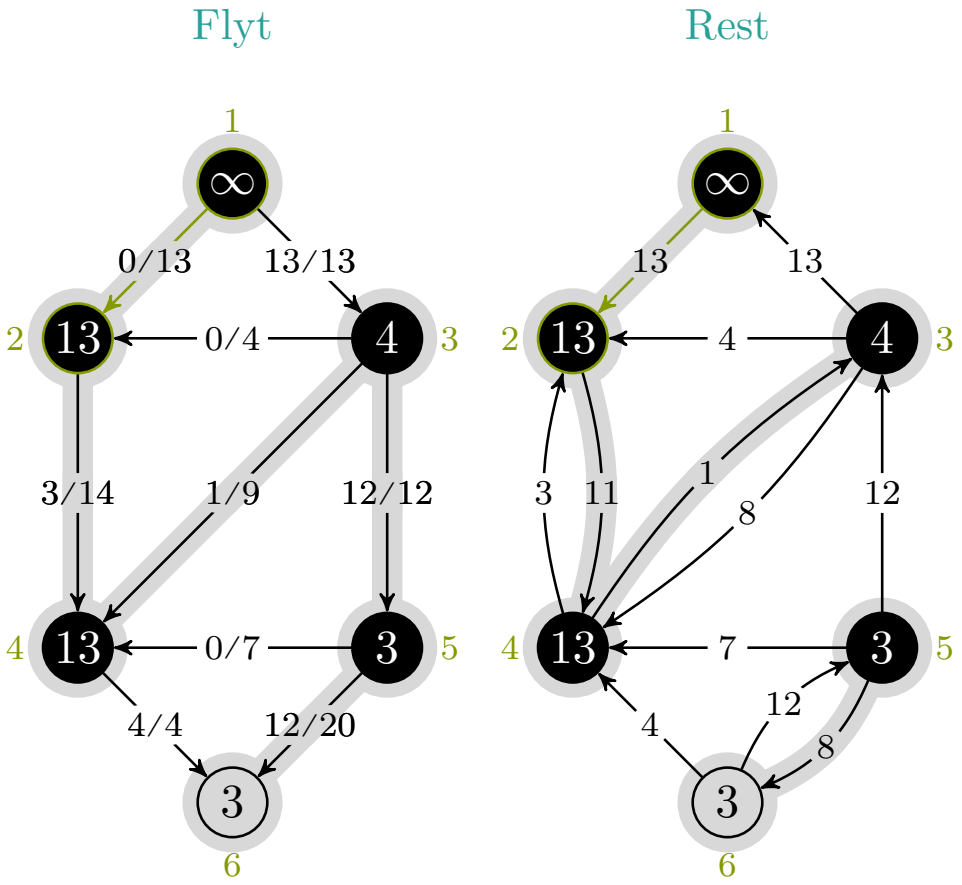
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 1, 2$



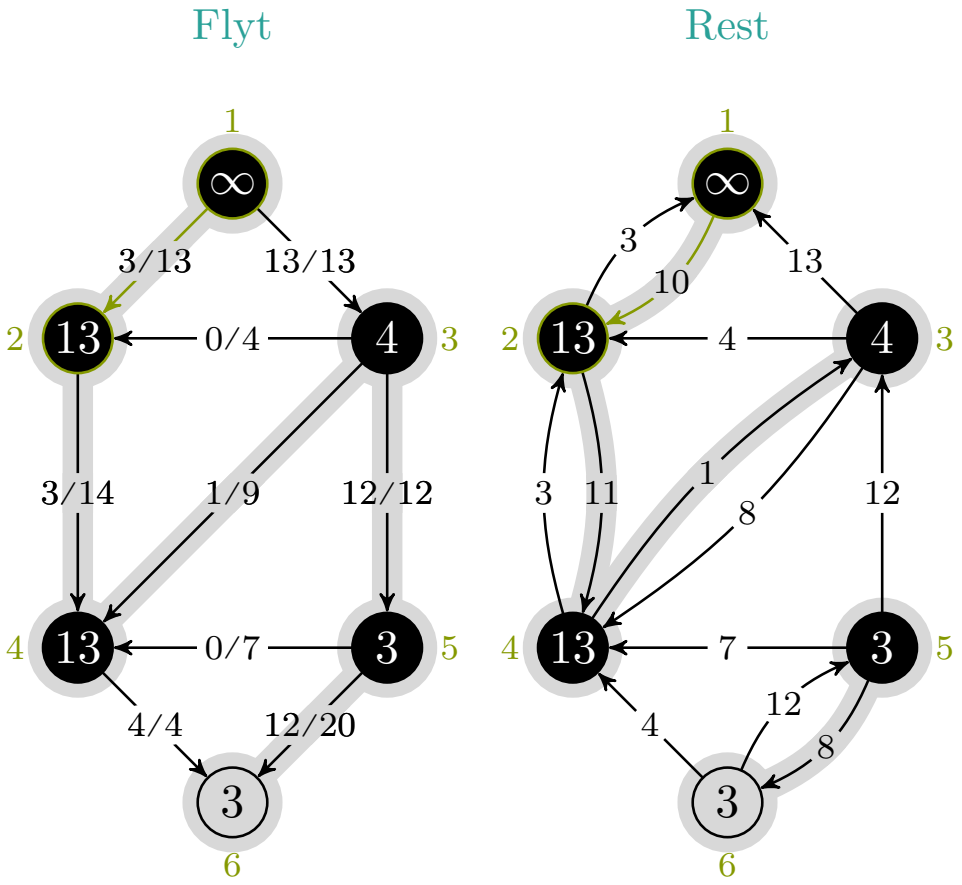
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 1, 2$



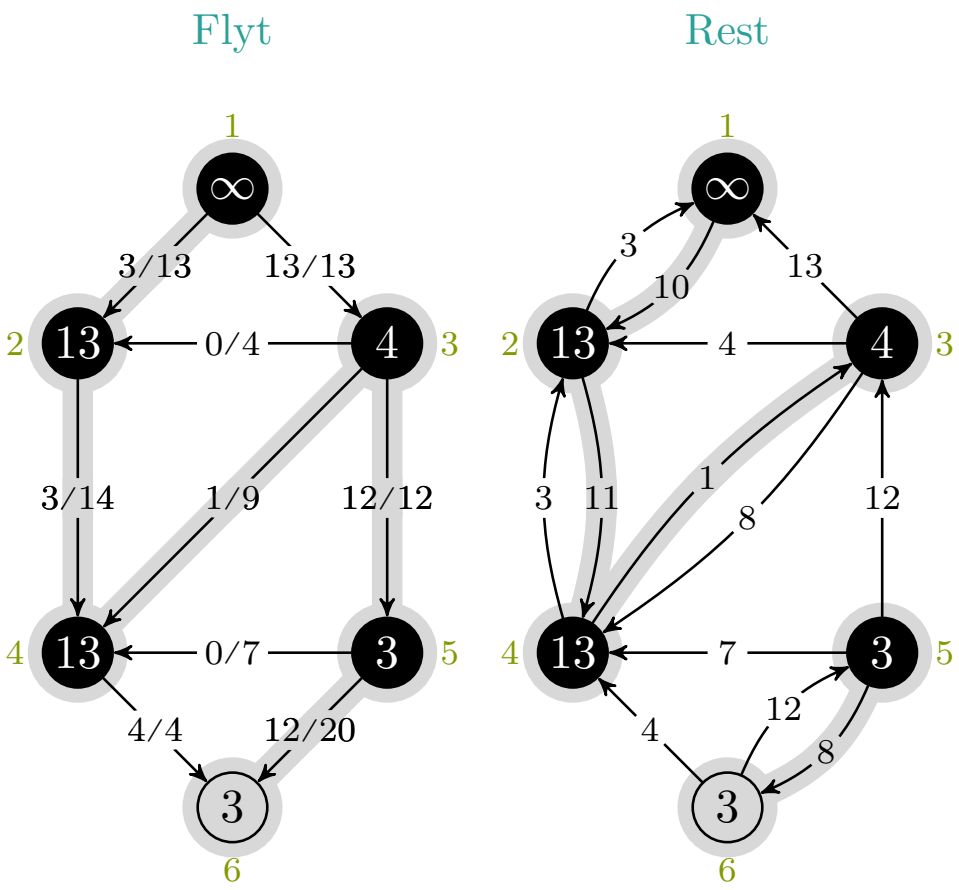
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, 1, 2$



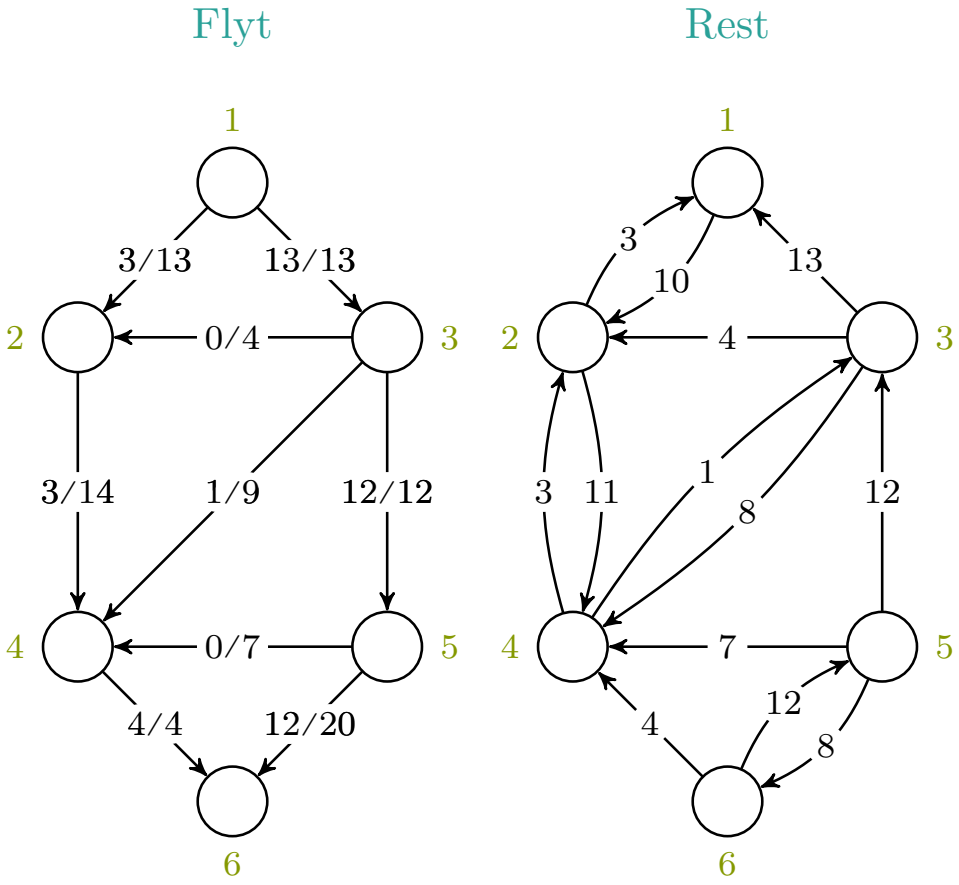
```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; 3, \text{NIL}, 1$



```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$



```
EDMONDS-KARP(G, s, t)
 1  for each edge (u, v) ∈ G.E
 2      (u, v).f = 0
 3  while BFS-LABELING(G, s, t)
 4      c_f(p) = t.f
 5      u, v = t.π, t
 6      while u ≠ NIL
 7          if (u, v) ∈ G.E
 8              (u, v).f = (u, v).f + c_f(p)
 9          else (v, u).f = (v, u).f − c_f(p)
10          u, v = u.π, u
```

$c_f(p), u, v \;=\; -, -, -$