Melhores momentos

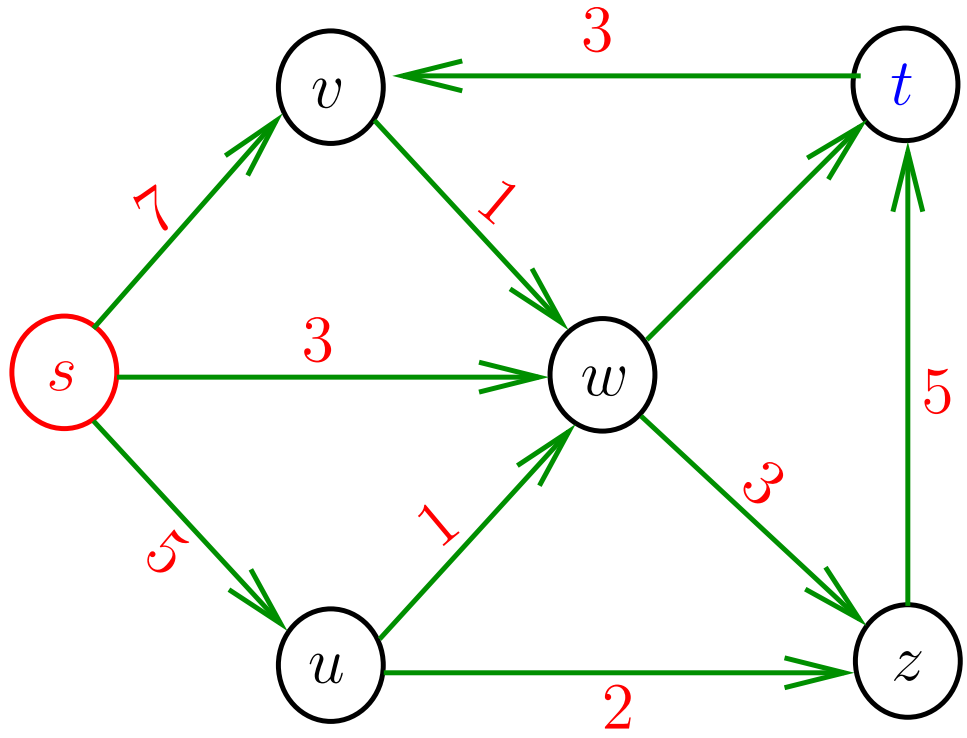
AULA 8

Problema

Problema do caminho mínimo sob custo não-negativo:
Dada uma rede (N, A, c) com função-custo $c : A \to \mathbb{Z}_{\geq}$ e um nó s, encontrar, para cada nó t, um caminho de custo mínimo de s a t.

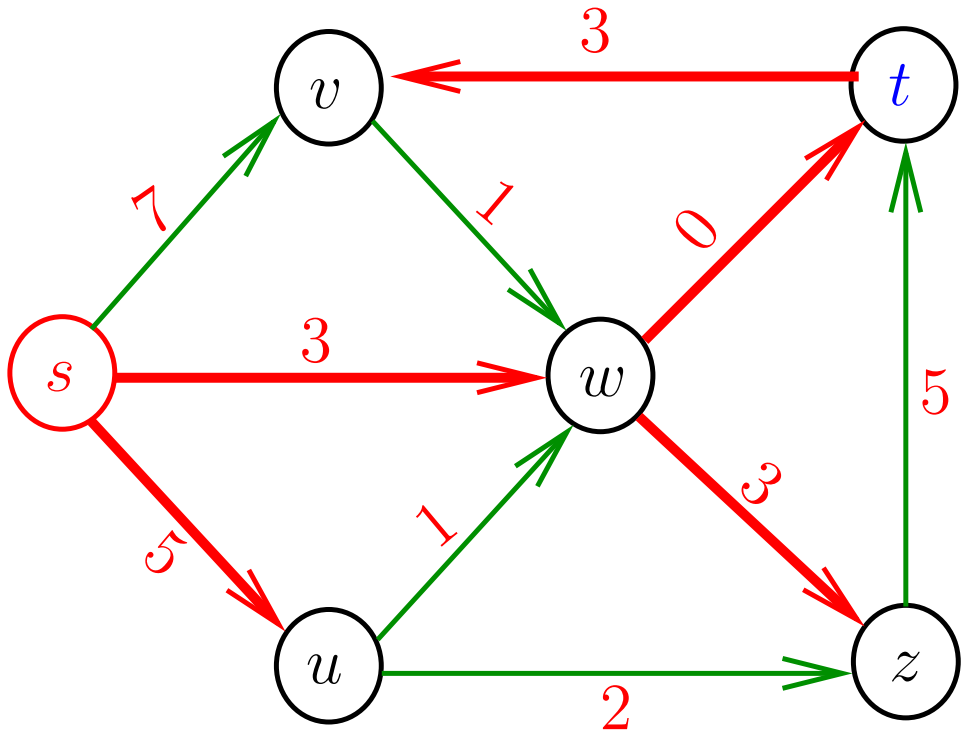
Problema

Problema do caminho mínimo sob custo não-negativo:
Dada uma rede (N, A, c) com função-custo $c : A \to \mathbb{Z}_{\geq}$ e um nó s, encontrar, para cada nó t, um caminho de custo mínimo de s a t.
Entra:



Problema

Problema do caminho mínimo sob custo não-negativo:
Dada uma rede (N, A, c) com função-custo $c : A \to \mathbb{Z}_{\geq}$ e um nó s, encontrar, para cada nó t, um caminho de custo mínimo de s a t.
Sai:



Algoritmo genérico

Recebe uma rede (N, A, c) com função custo $c : A \to \mathbb{Z}_{\geq}$ e um nó s e devolve uma função-predecessor π e uma função-potencial y com a seguinte propriedade: para todo nó t, a função π determina um caminho P de s a t tal que $c(P) = y(t) - y(s)$.

FORD (N, A, c, s) $\quad \triangleright\ c \geq 0$

1 **para cada** i em N **faça**
2 $\quad y(i) \leftarrow nC + 1 \quad \triangleright\ C := \max\{c(ij) : ij \in A\}$
3 $\quad \pi(i) \leftarrow \text{NIL}$
4 $y(s) \leftarrow 0$
5 **enquanto** $y(j) > y(i) + c(ij)$ **para** algum $ij \in A$ **faça**
6 $\quad y(j) \leftarrow y(i) + c(ij)$
7 $\quad \pi(j) \leftarrow i$
8 **devolva** π e y

Conclusão

Da propriedade dos c-potenciais (lema da dualidade) e da correção do algoritmo FORD concluímos o seguinte:

(Teorema dualidade) Se s e t são nós de uma rede (N, A, c) com função custo $c : A \to \mathbb{Z}_{\geq}$ e t está ao alcance de s então

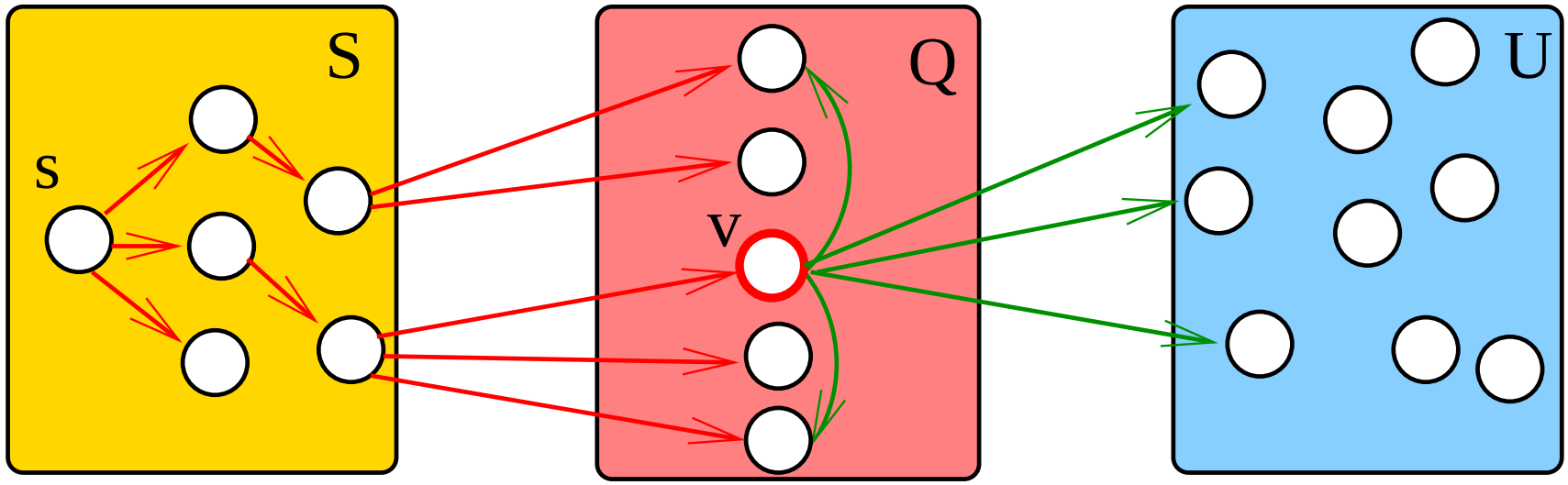
$$\min\{c(P) : P \text{ é um } st\text{-caminho}\}$$
$$= \max\{y(t) - y(s) : y \text{ é um } c\text{-potencial}\}.$$

Conclusão

O consumo de tempo do algoritmo CAMINHO-CURTO-GENÉRICO é $\mathrm{O}(n^2 m(C+1))$.

Este consumo de tempo não é polinomial.

Iteração de Dijkstra



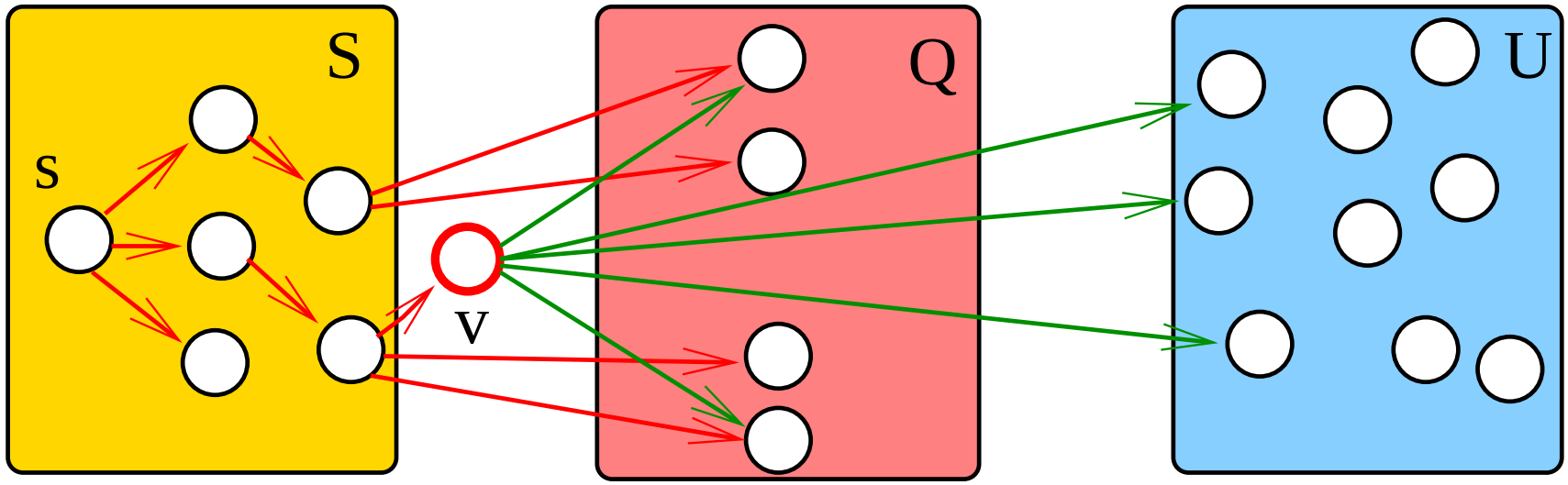
$\mathtt{S}$ = nós já examinados

$\mathtt{Q}$ = nós não examinados com potencial $< \infty$

$\mathtt{U}$ = nós não examinados com potencial $= \infty$

$\mathtt{v}$ = nó em $\mathtt{Q}$ com menor potencial

Iteração de Dijkstra



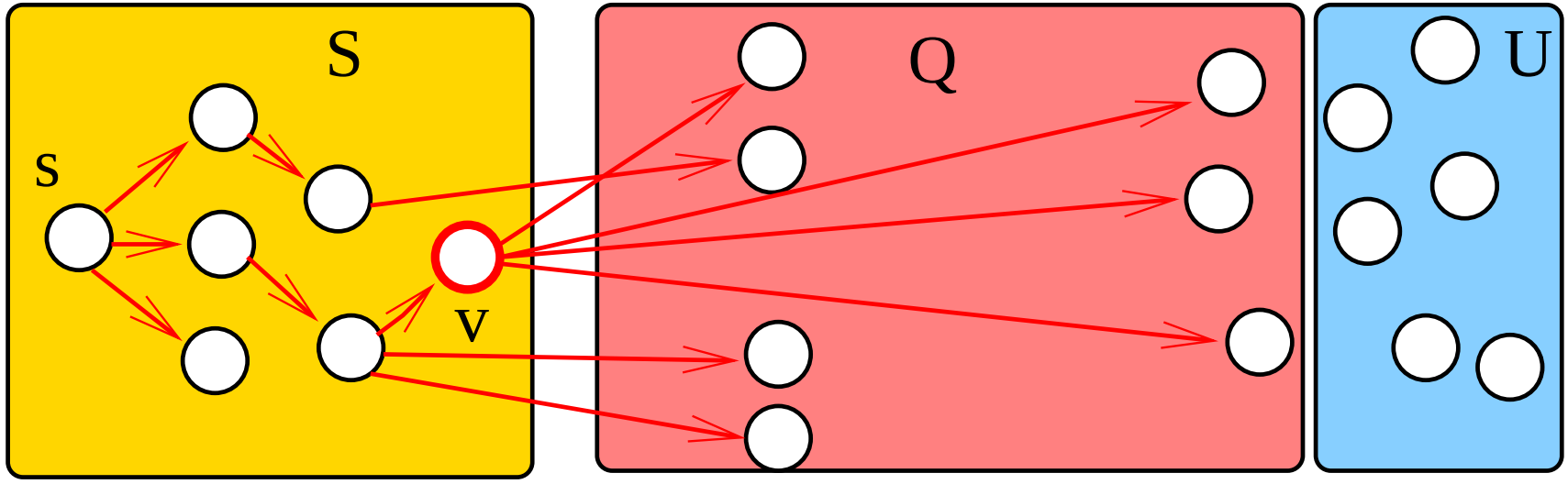
$\mathtt{S} =$ nós já examinados

$\mathtt{Q} =$ nós não examinados com potencial $< \infty$

$\mathtt{U} =$ nós não examinados com potencial $= \infty$

$\mathtt{v} =$ nó em $\mathtt{Q}$ com menor potencial

Iteração de Dijkstra



$\mathtt{S} =$ nós já examinados

$\mathtt{Q} =$ nós não examinados com potencial $< \infty$

$\mathtt{U} =$ nós não examinados com potencial $= \infty$

$\mathtt{v} =$ nó em $\mathtt{Q}$ com menor potencial

Invariantes

Na linha 6, antes da verificação da condição "$Q \neq \langle\rangle$" valem as seguintes invariantes:

(i0) para cada arco pq no grafo de predecessores tem-se $y(q) - y(p) = c(pq)$ (igual!);

(i1) $\pi(s) = \text{NIL}$ e $y(s) = 0$;

(i2) para cada nó v distinto de s, $y(v) < nC + 1 \Leftrightarrow \pi(v) \neq \text{NIL}$;

(i3) para cada nó v, se $\pi(v) \neq \text{NIL}$ então existe um caminho de s a v no grafo de predecessores.

Mais invariantes

Na linha 6, antes da verificação da condição $Q \neq \langle\rangle$ valem, além de (i0)-(i3), as seguintes invariantes:

(i4) para cada arco pq com $y(q) - y(p) > c(pq)$ tem-se que p e q estão Q;

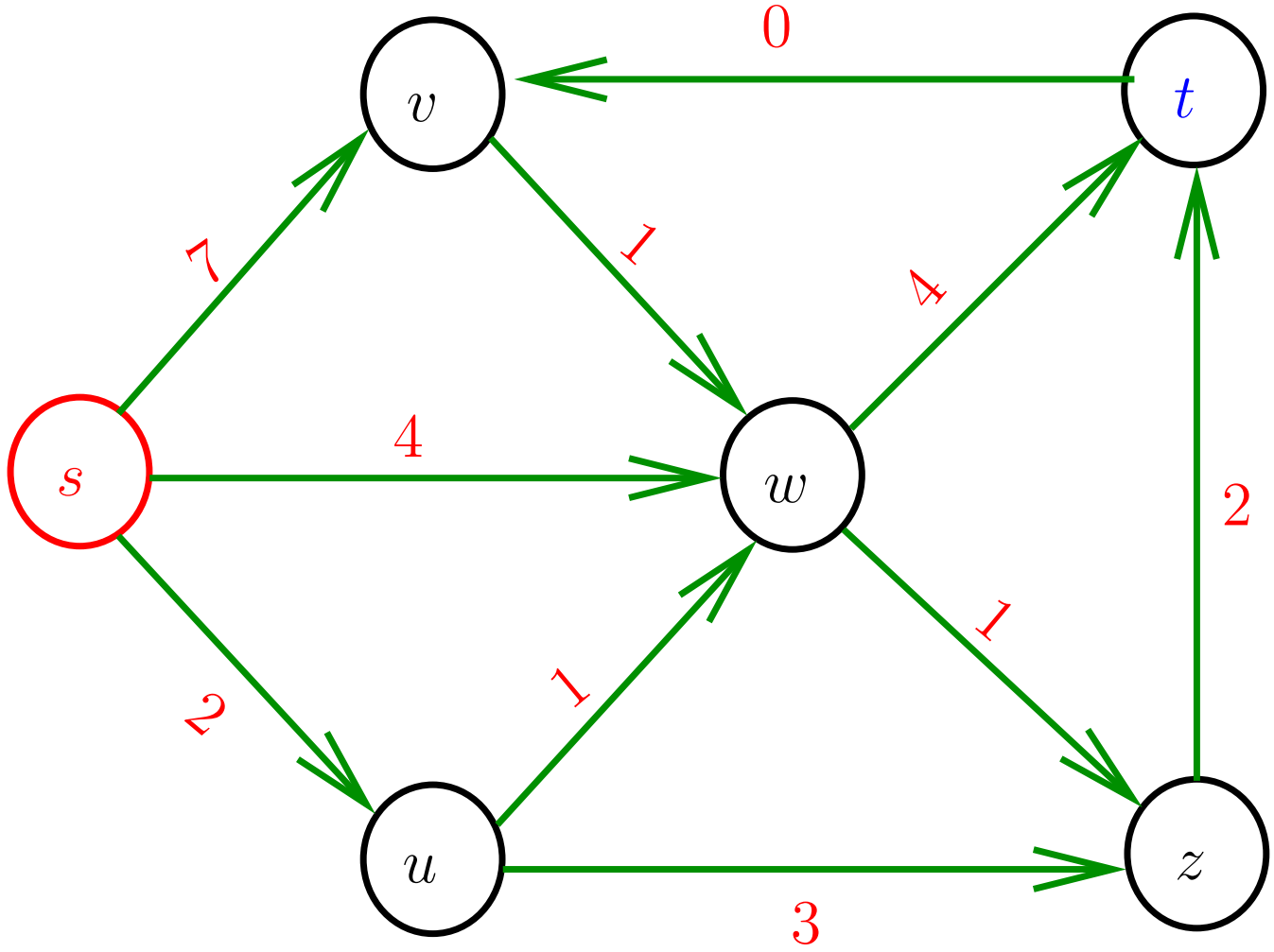
(i5) (monotonicidade) para quaisquer w em $N - Q$, i em Q vale que

$$y(w) \leq y(i).$$

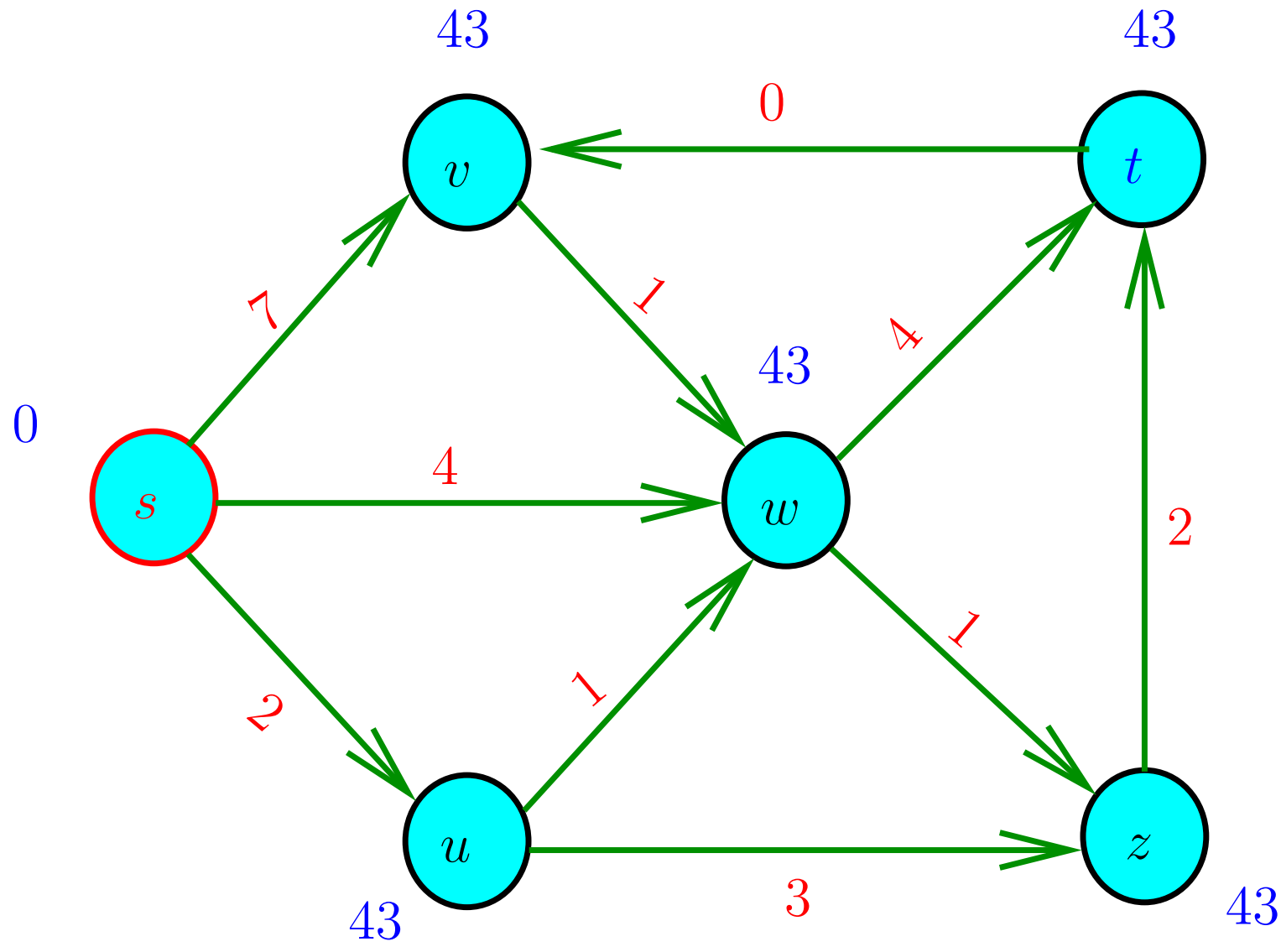
Implementação do algoritmo Ford

```
DIJKSTRA (N, A, c, s)      ▷ c ≥ 0
 1   para cada i em N faça
 2       y(i) ← nC + 1      ▷ nC + 1 faz o papel de ∞
 3       π(i) ← NIL
 4   y(s) ← 0
 5   Q ← N      ▷ Q func. como uma fila com prioridades
 6   enquanto Q ≠ ⟨⟩ faça
 7       retire de Q um nó i com y(i) mínimo
 8       para cada ij em A(i) faça
 9           se y(j) > y(i) + c(ij) então
10               y(j) ← y(i) + c(ij)
11               π(j) ← i
12   devolva π e y
```

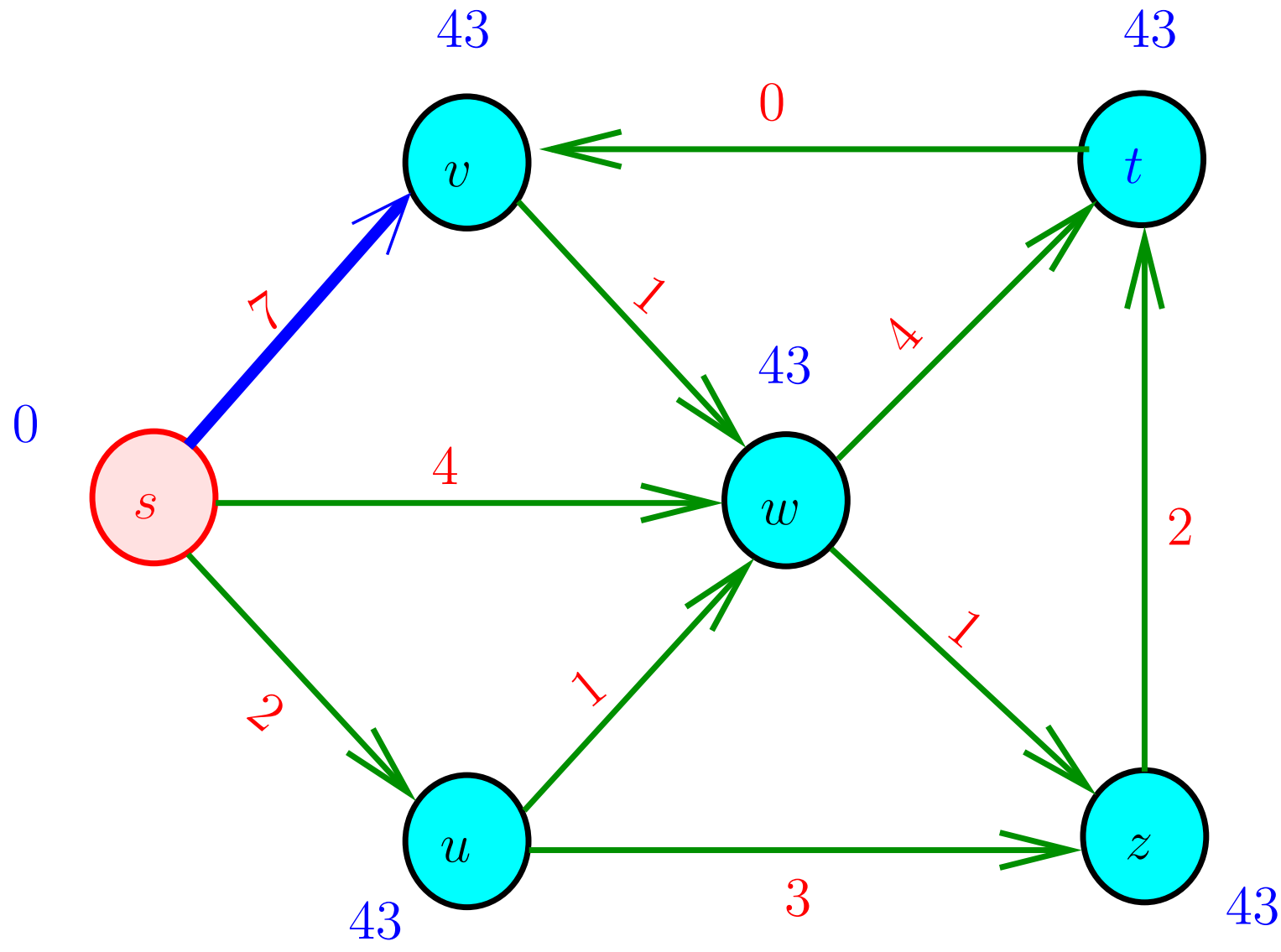
Simulação



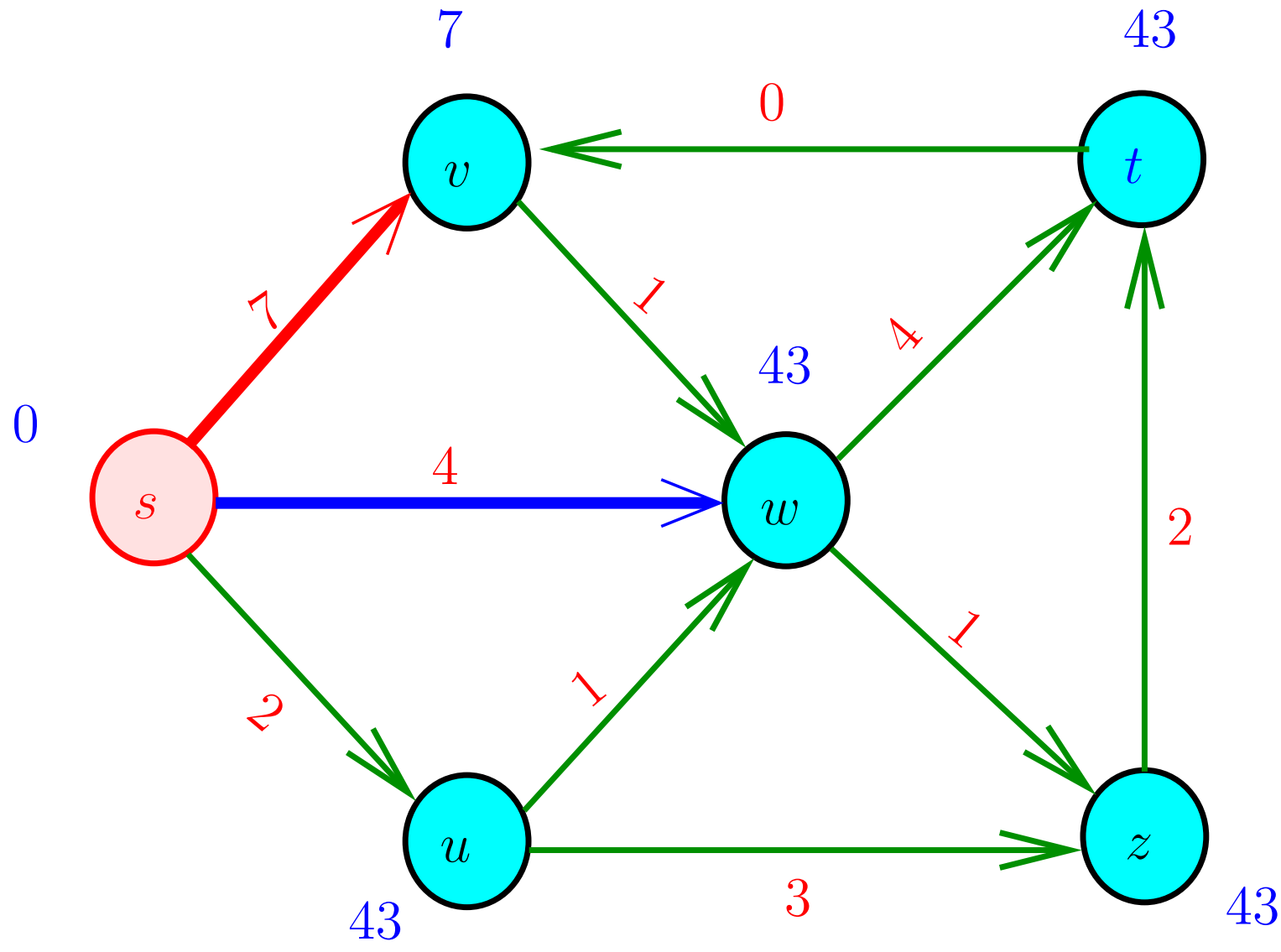
Simulação



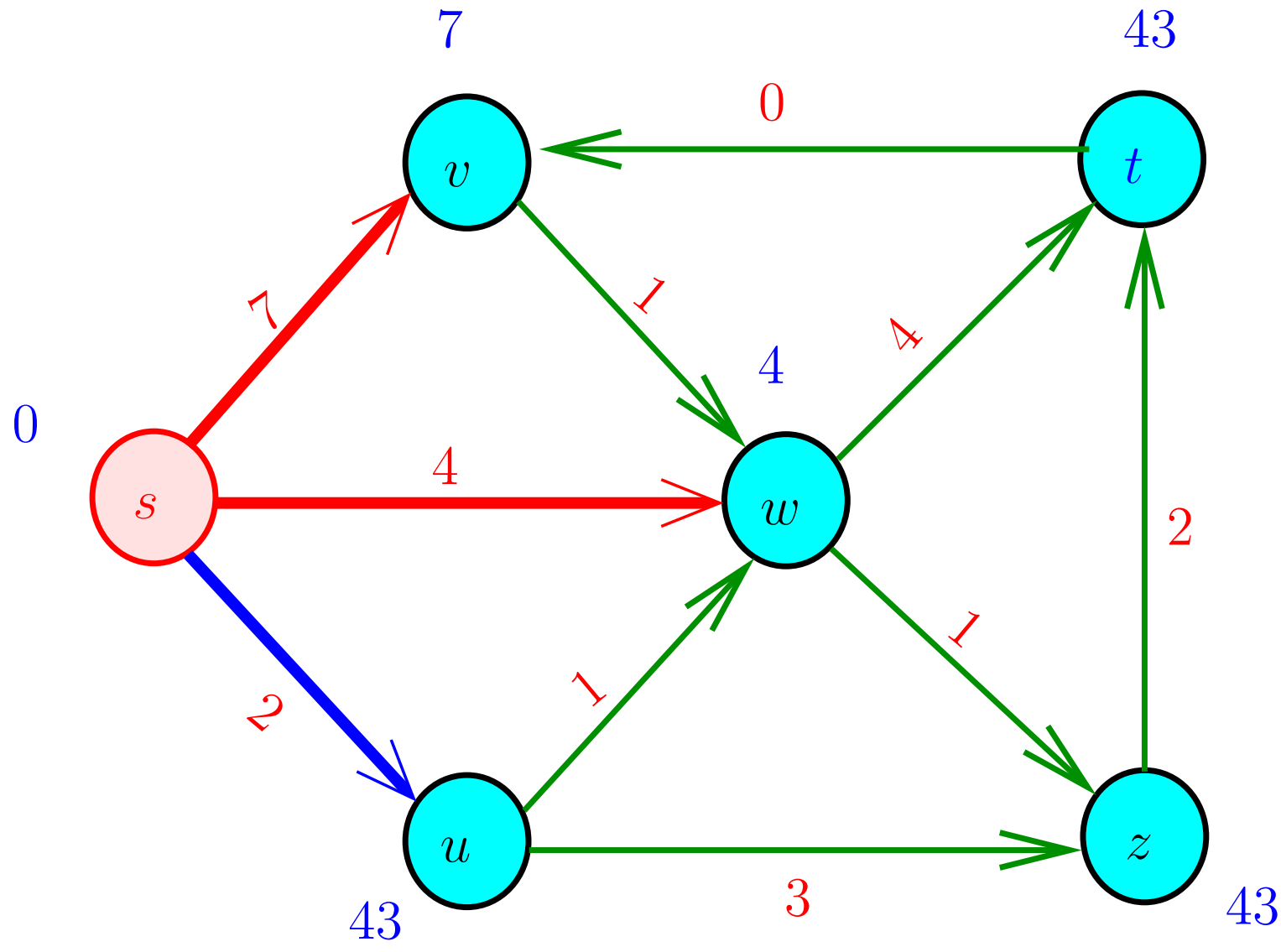
Simulação



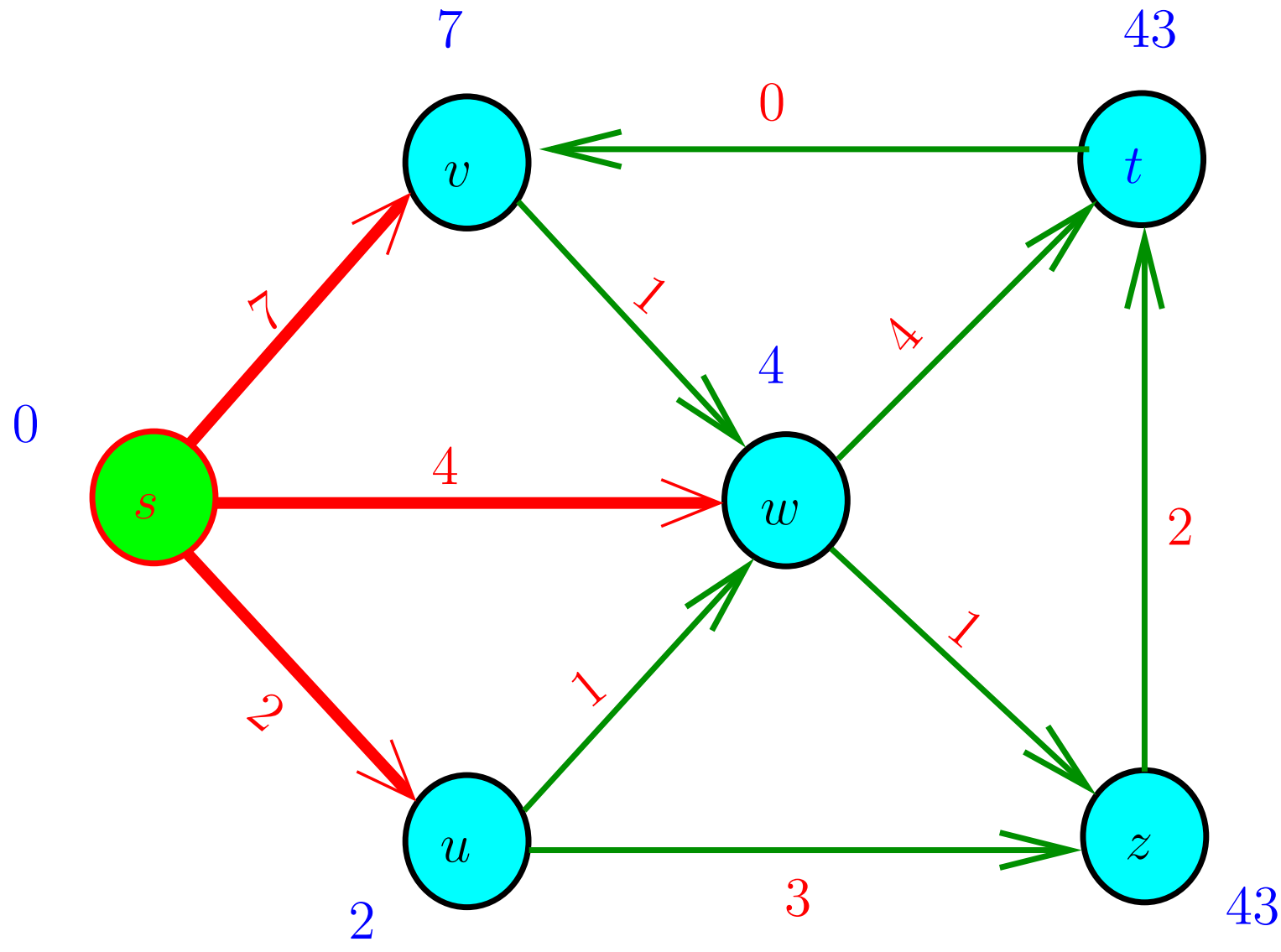
Simulação



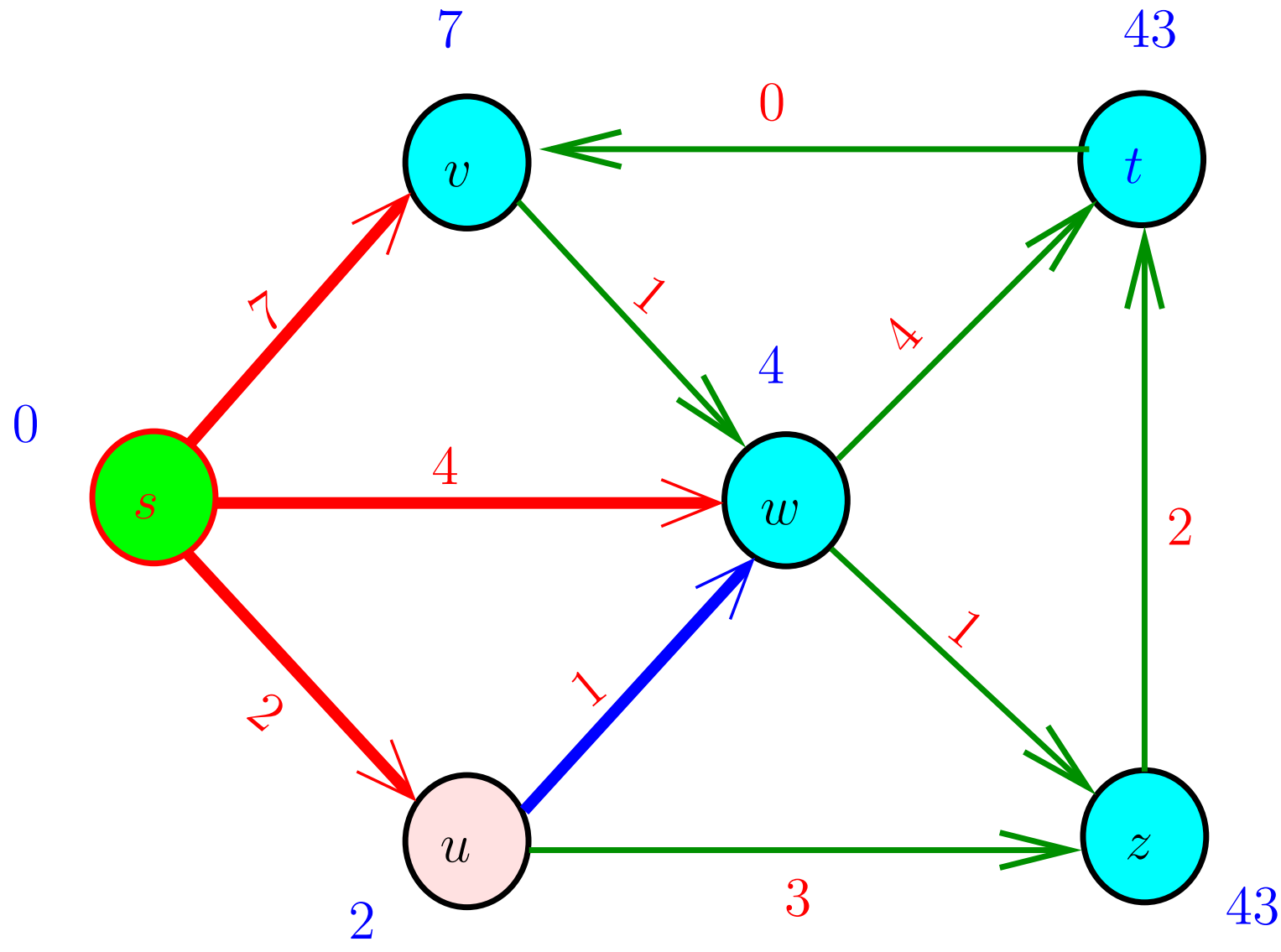
Simulação



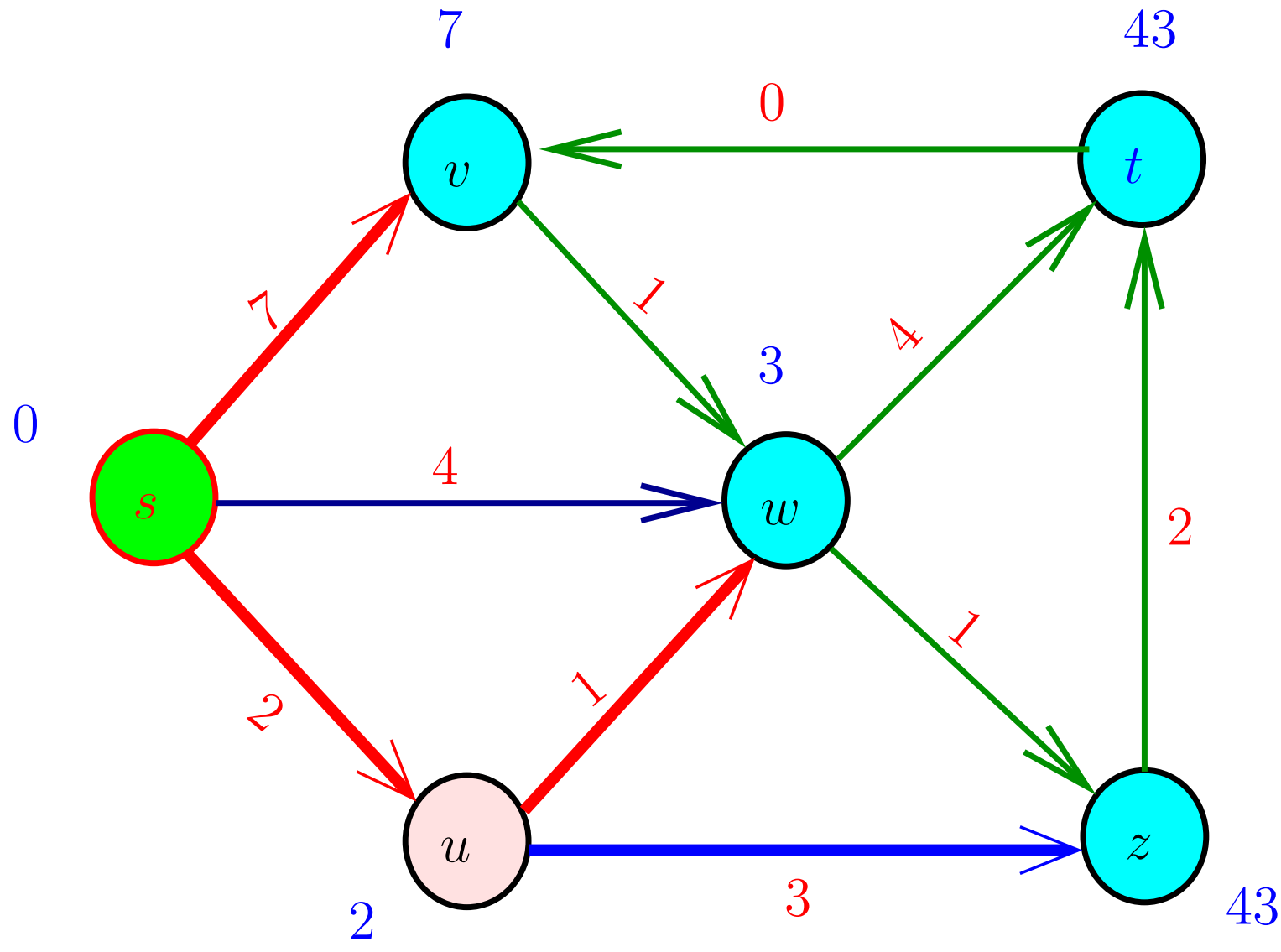
Simulação



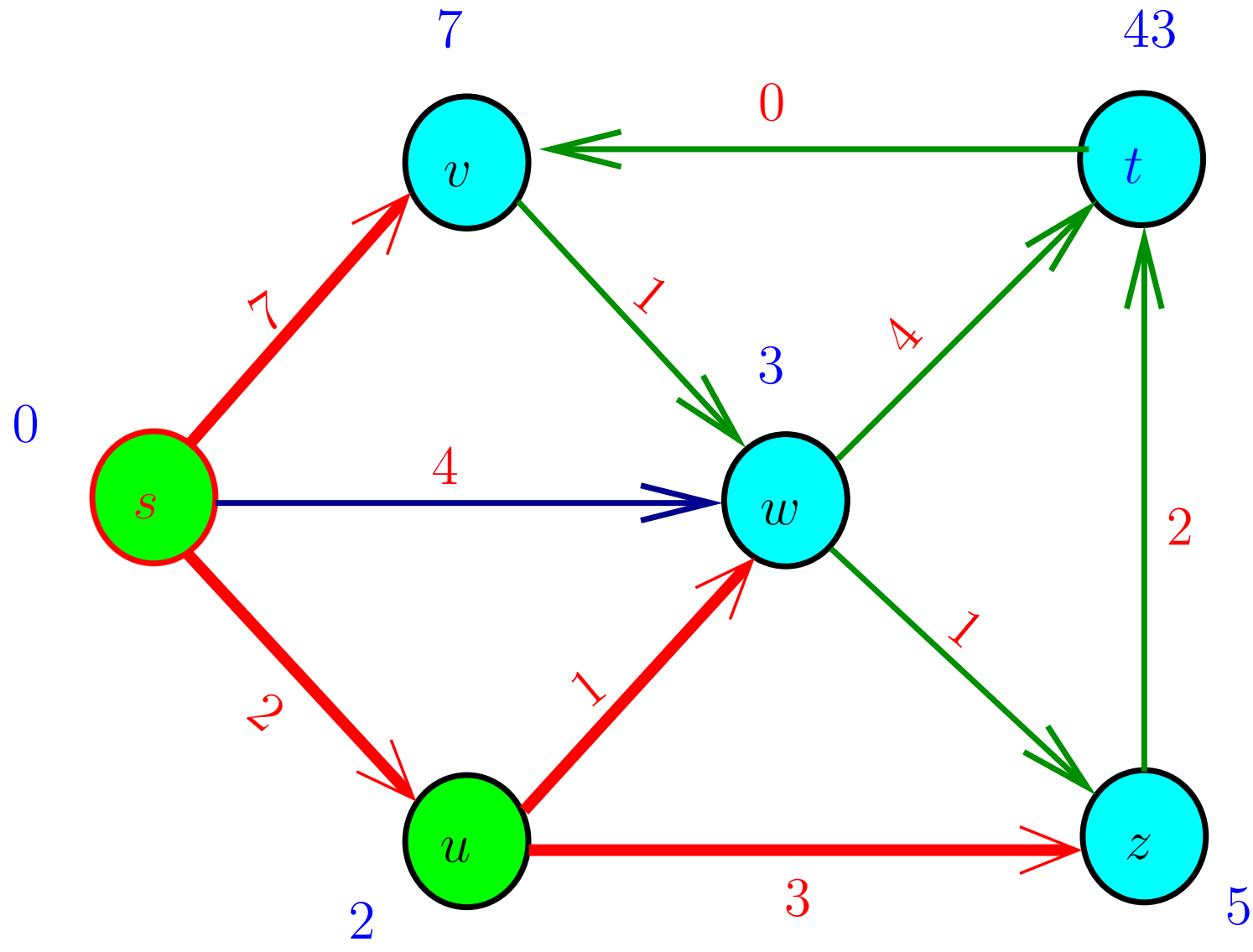
Simulação



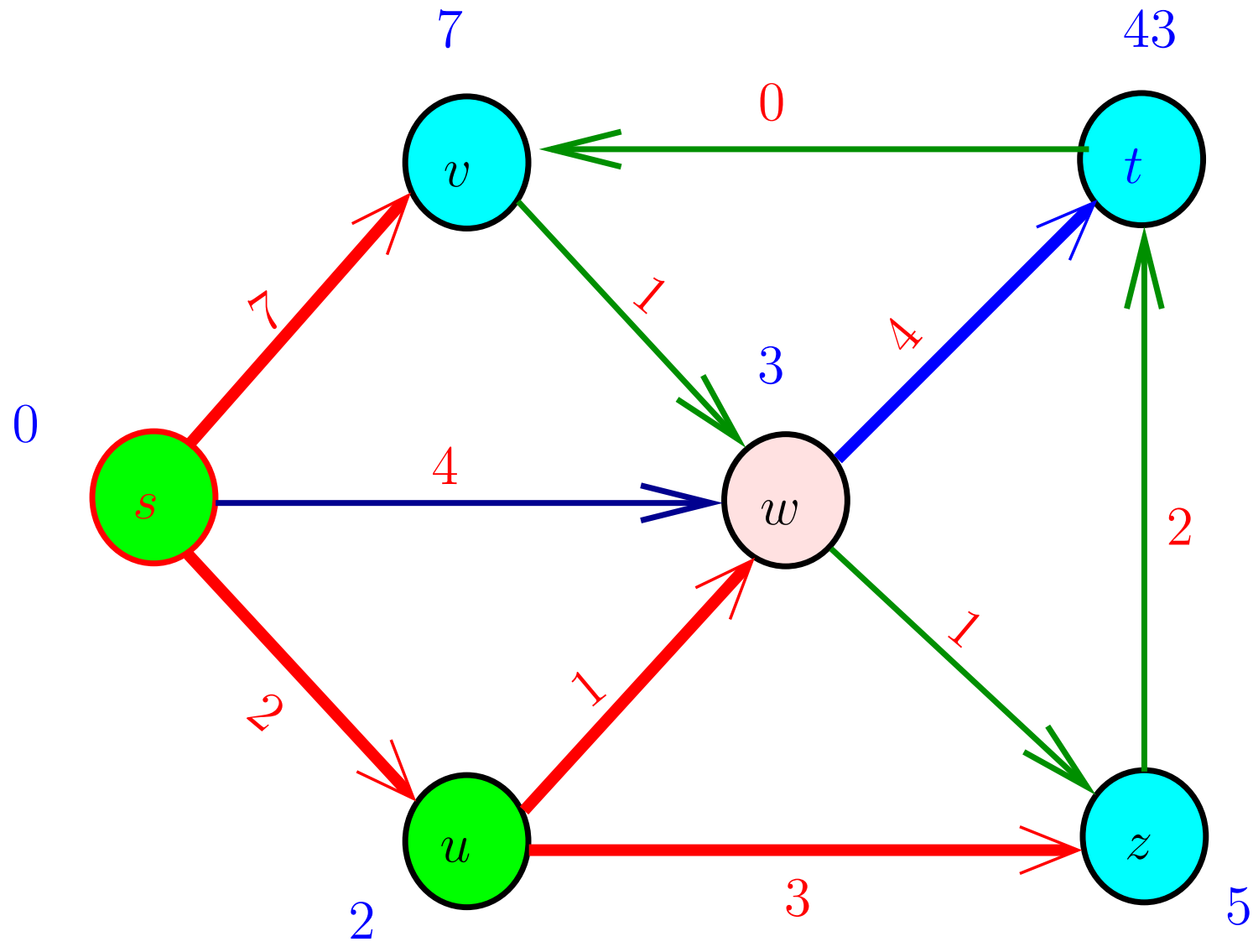
Simulação



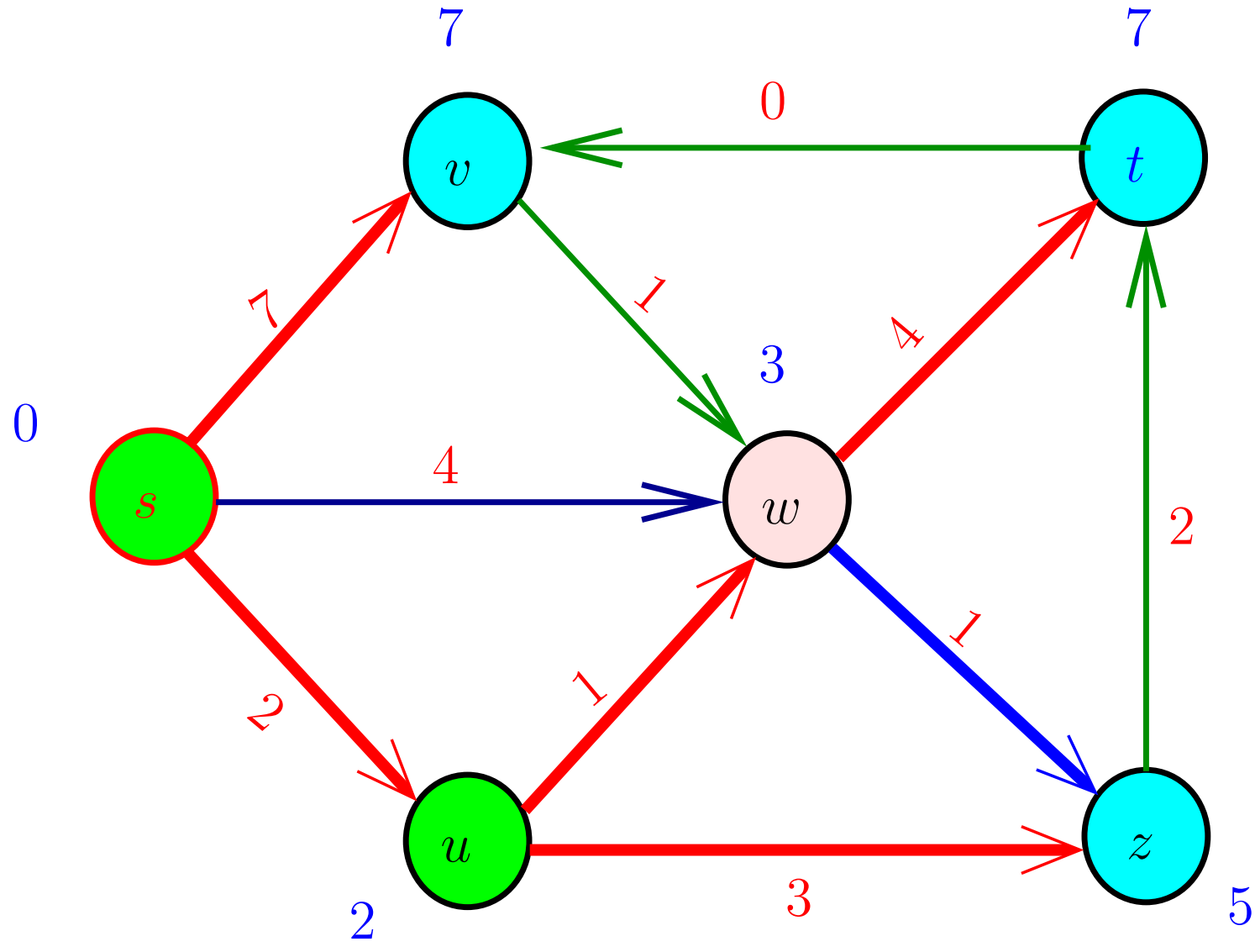
Simulação



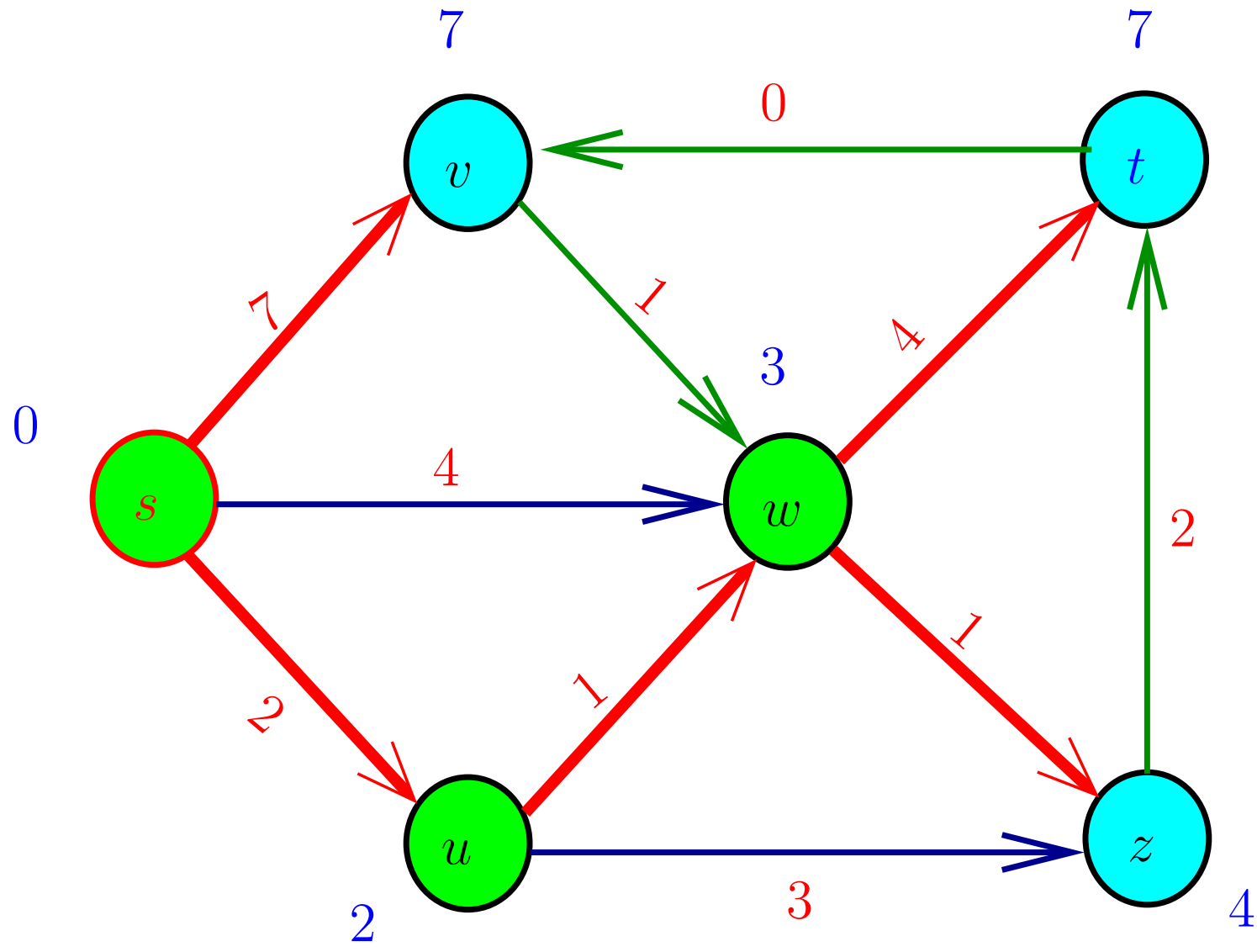
Simulação



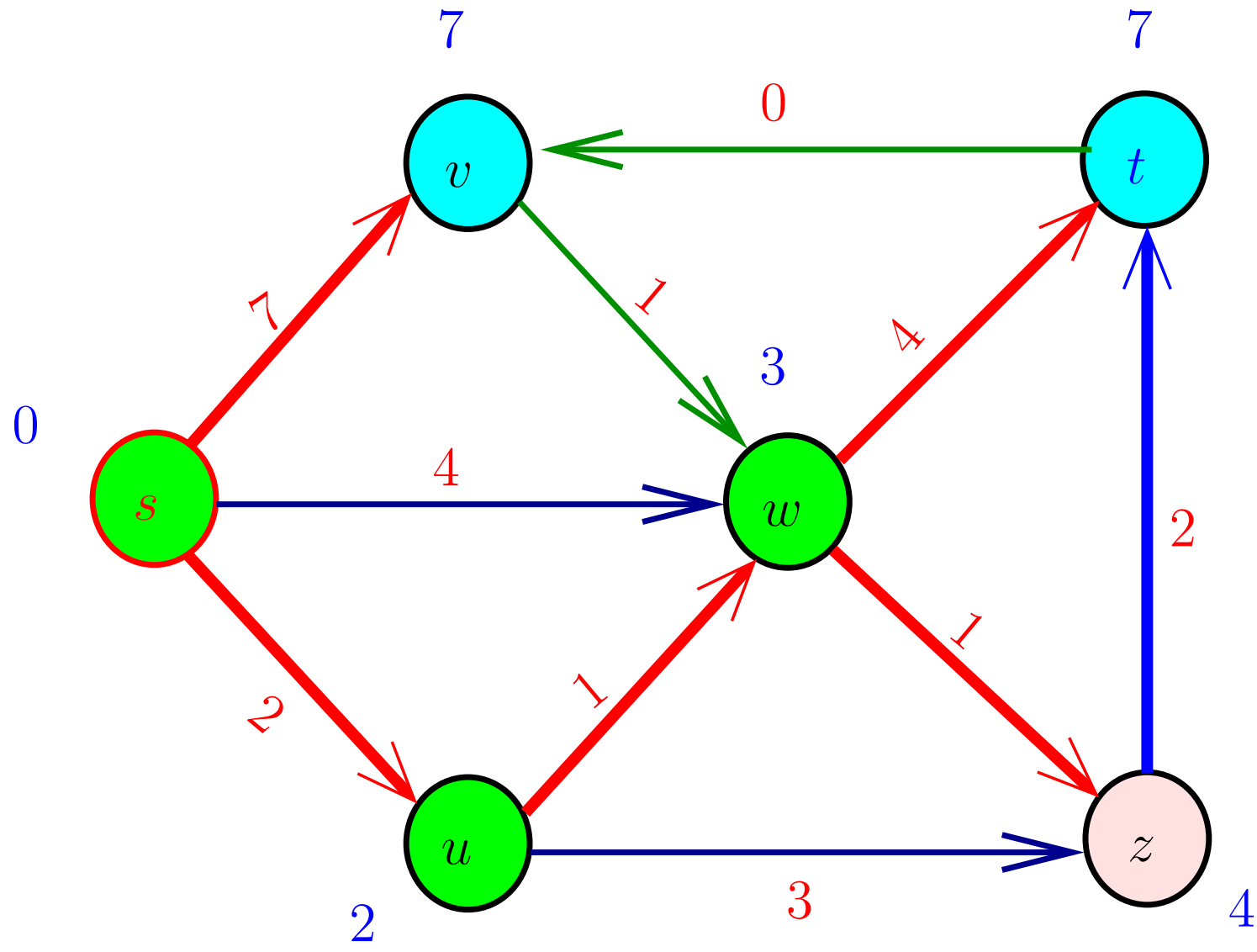
Simulação



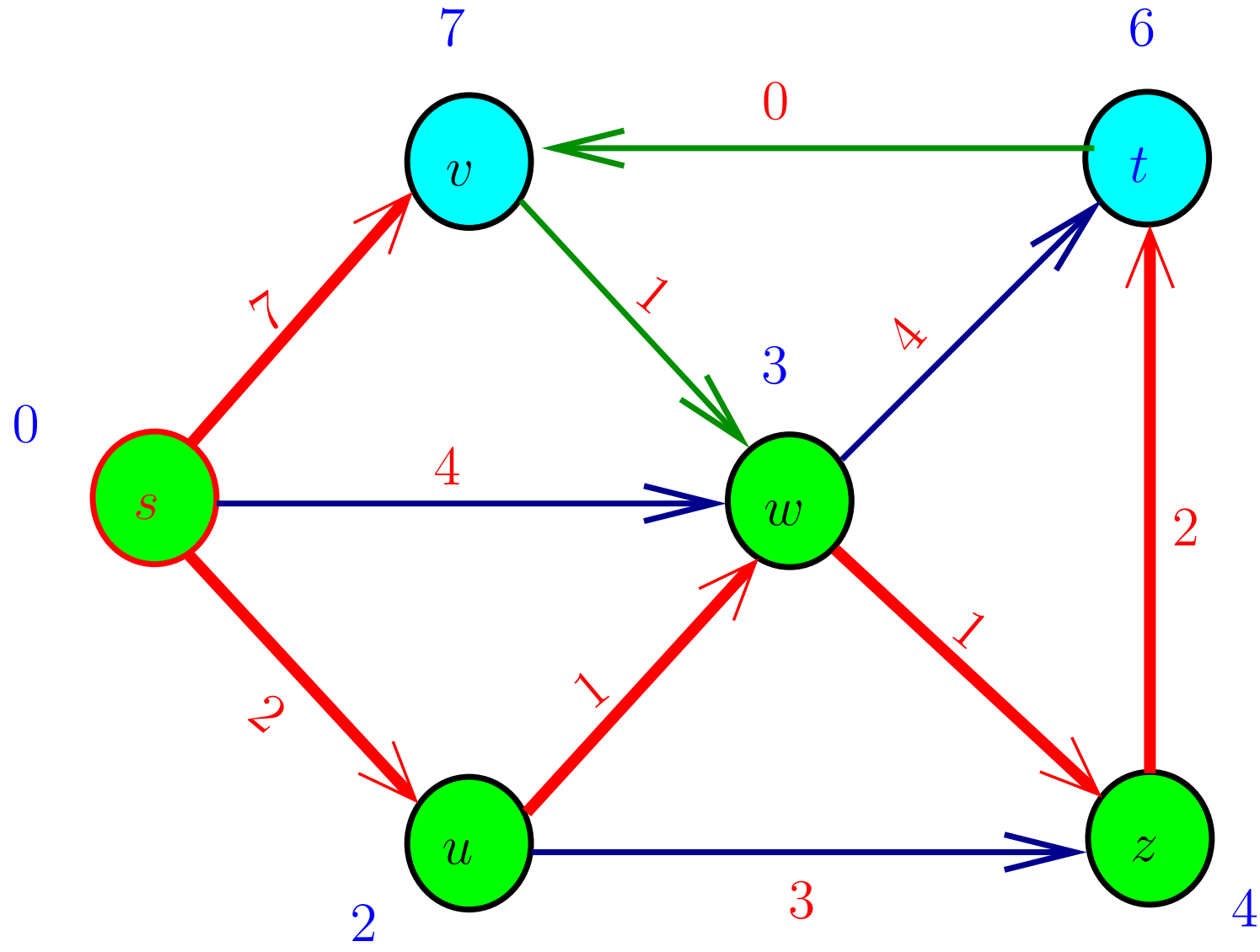
Simulação



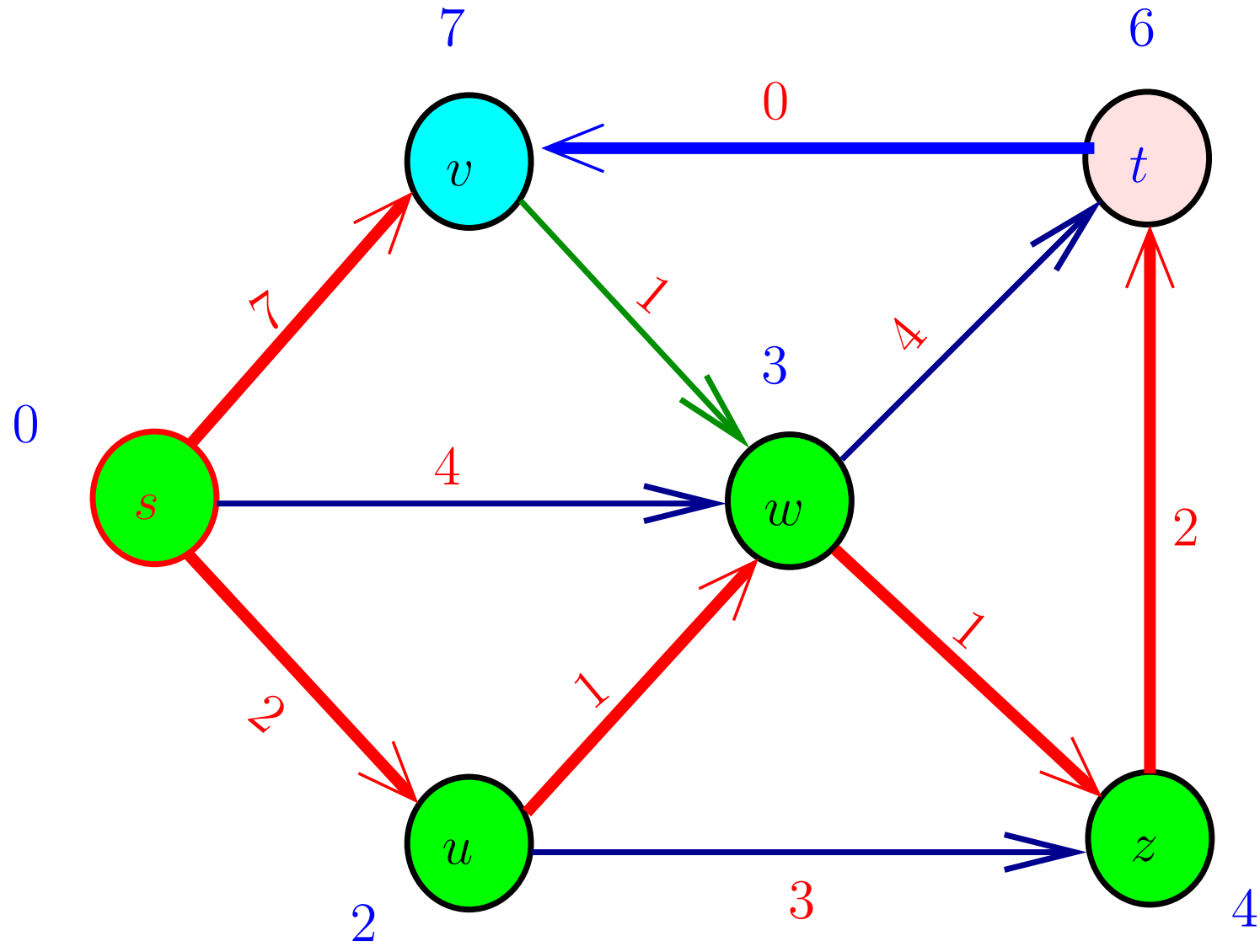
Simulação



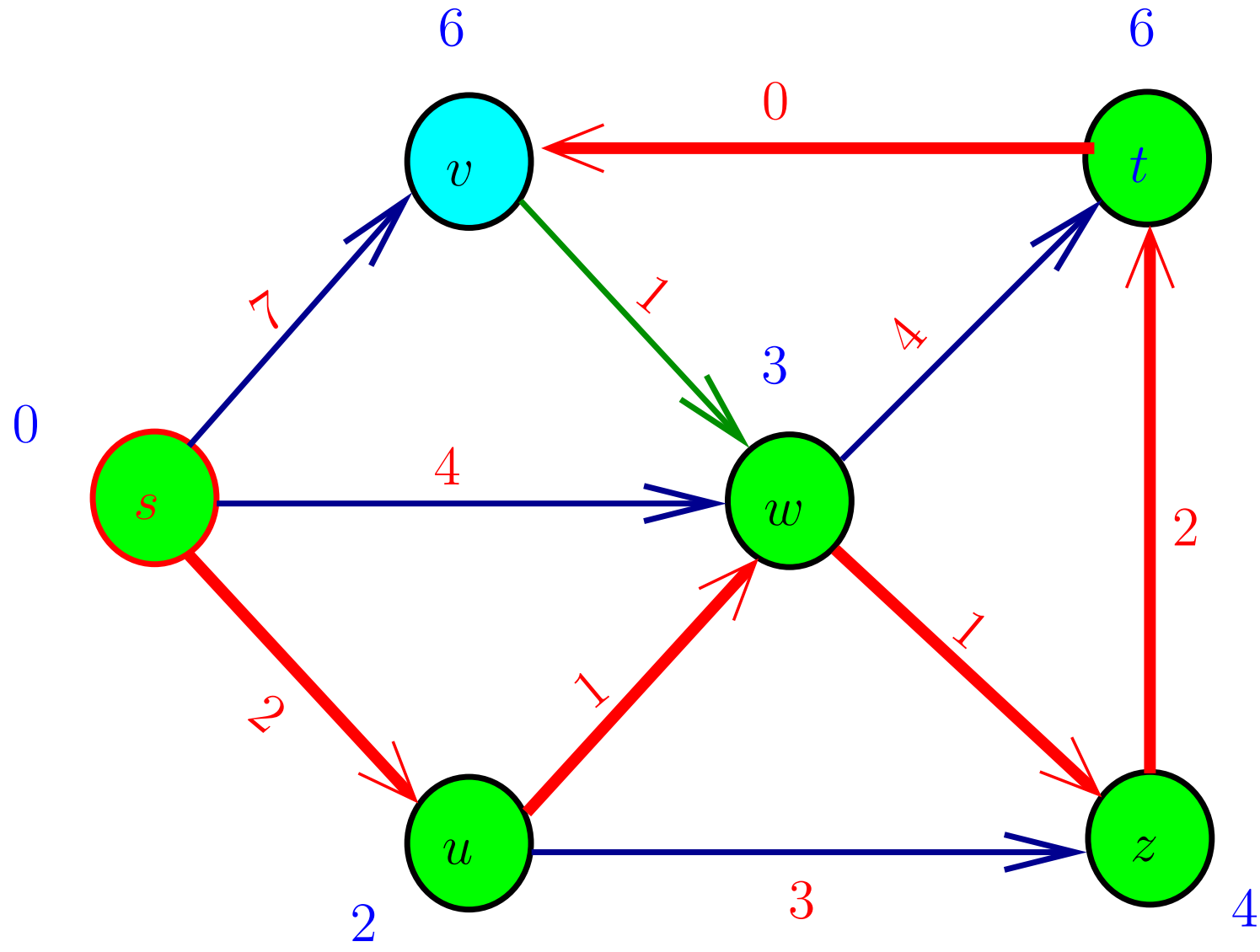
Simulação



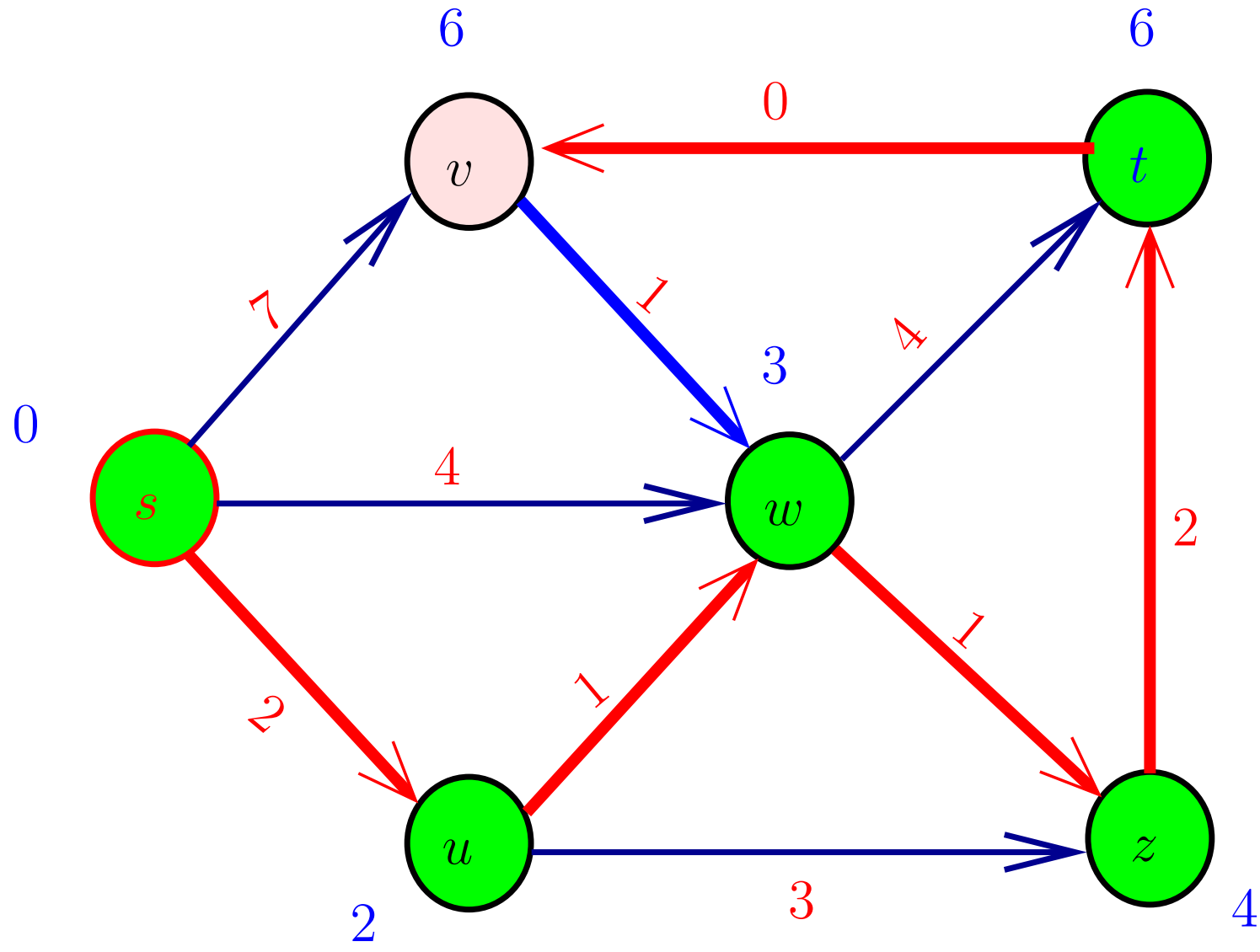
Simulação



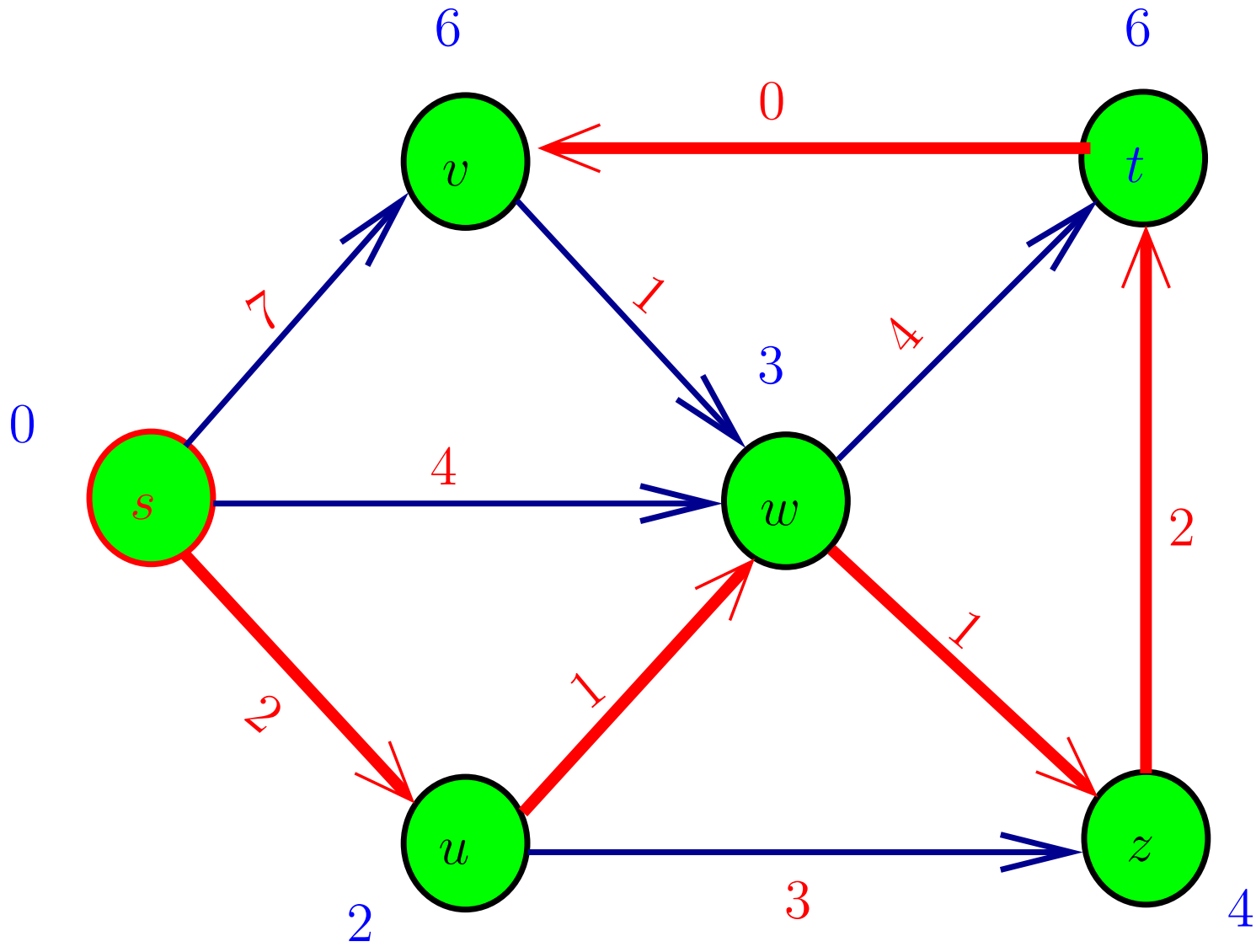
Simulação



Simulação



Simulação



Consumo de tempo

O consumo de tempo do algoritmo DIJKSTRA é $\mathrm{O}(n^2)$.

AULA 9

Caminhos de custo mínimo

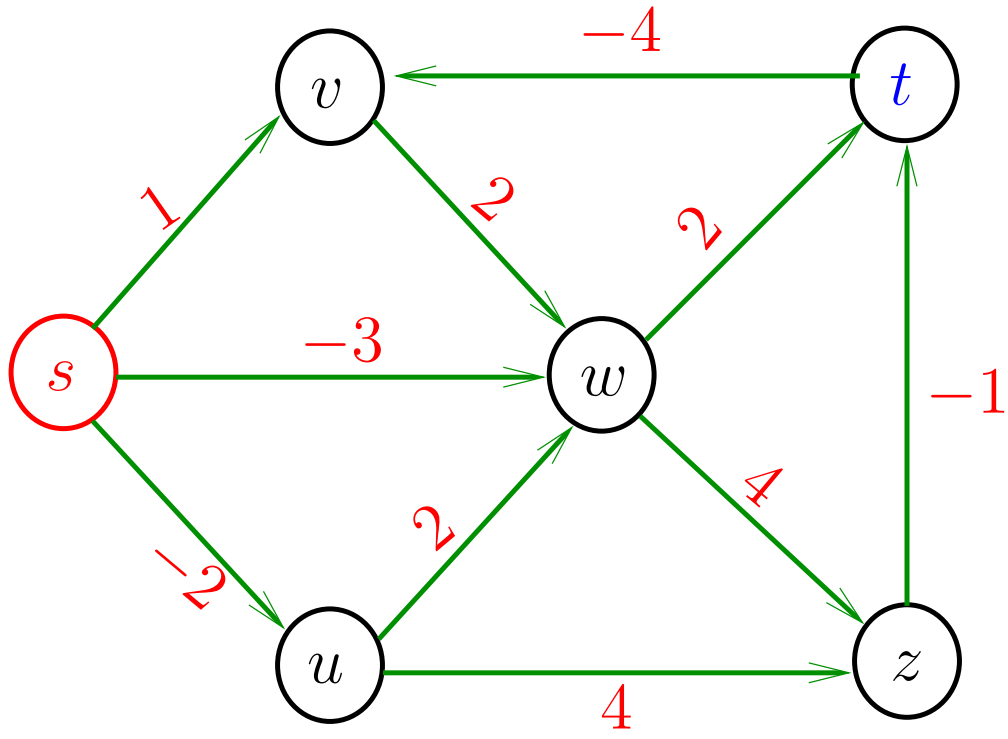
PF 6.1, 6.2, 6.3, 6.4, 6.5

Problema

Problema do caminho de custo mínimo: Dada uma rede (N, A, c) com função-custo $c : A \to \mathbb{Z}$ e um nó s, encontrar, para cada nó t, um caminho de custo mínimo de s a t.

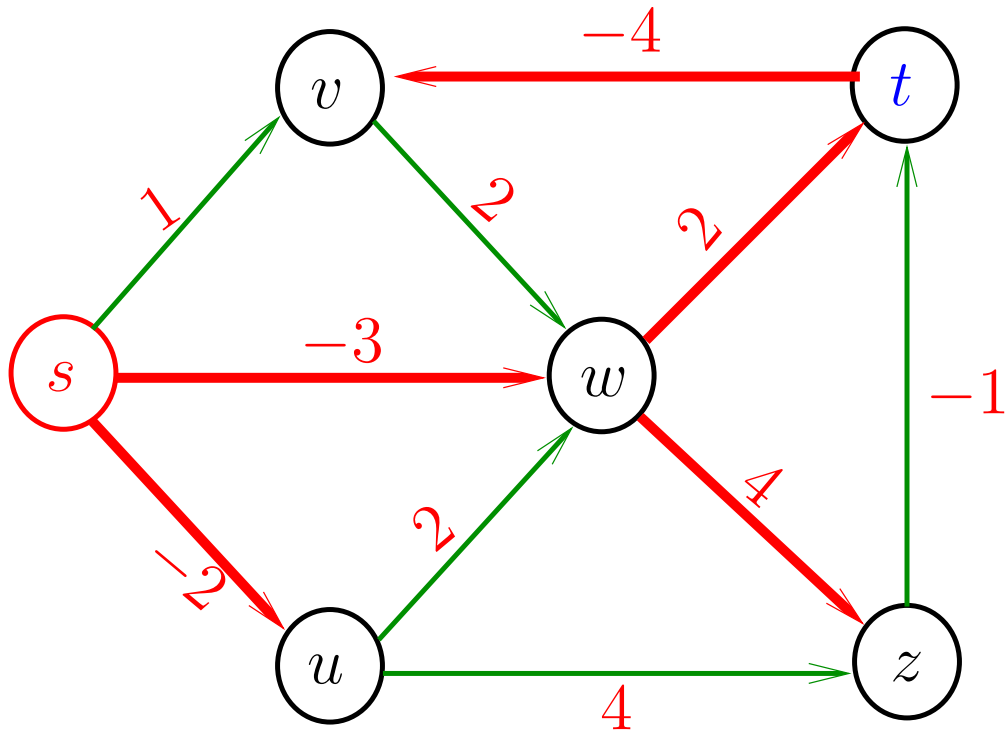
Problema

Problema do caminho de custo mínimo: Dada uma rede (N, A, c) com função-custo $c : A \to \mathbb{Z}$ e um nó s, encontrar, para cada nó t, um caminho de custo mínimo de s a t.
Entra:



Problema

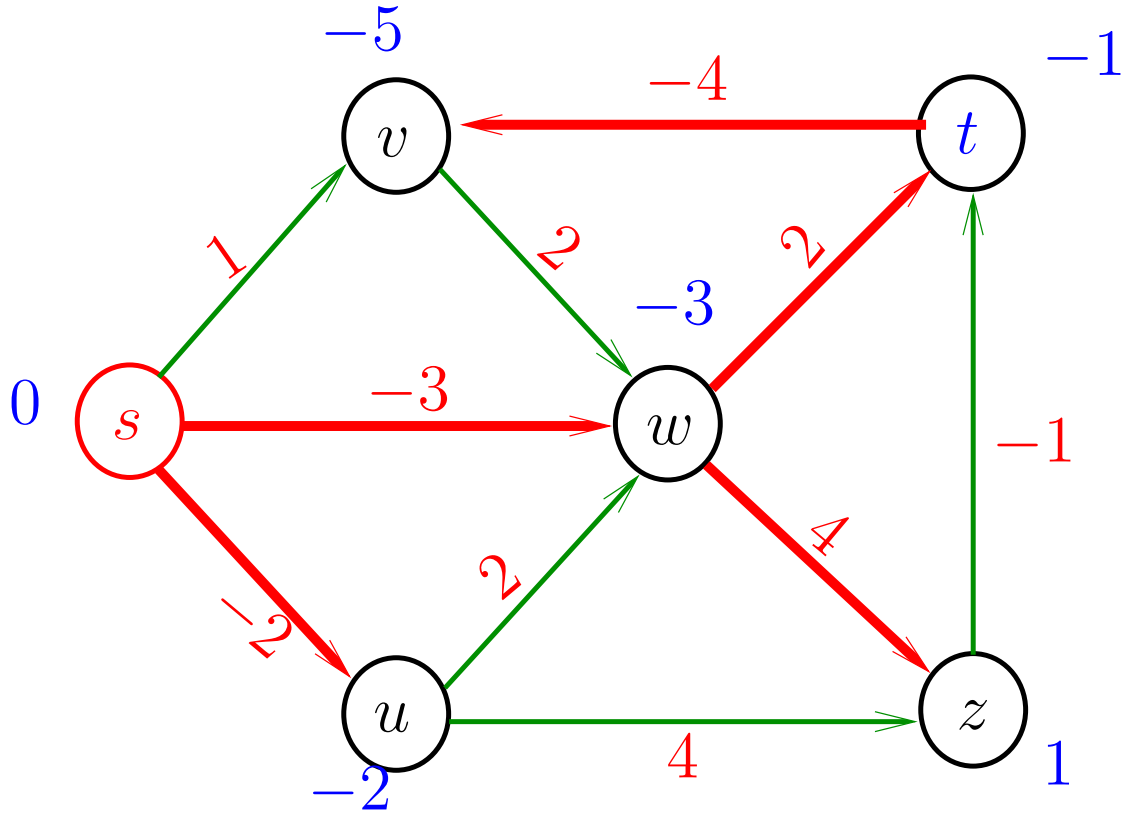
Problema do caminho de custo mínimo: Dada uma rede (N, A, c) com função-custo $c : A \to \mathbb{Z}$ e um nó s, encontrar, para cada nó t, um caminho de custo mínimo de s a t.
Sai:



Propriedade de c-Potenciais

(Lema da dualidade.) Se P é um passeio de s a t e y é um c-potencial então

$$c(P) \geq y(t) - y(s).$$

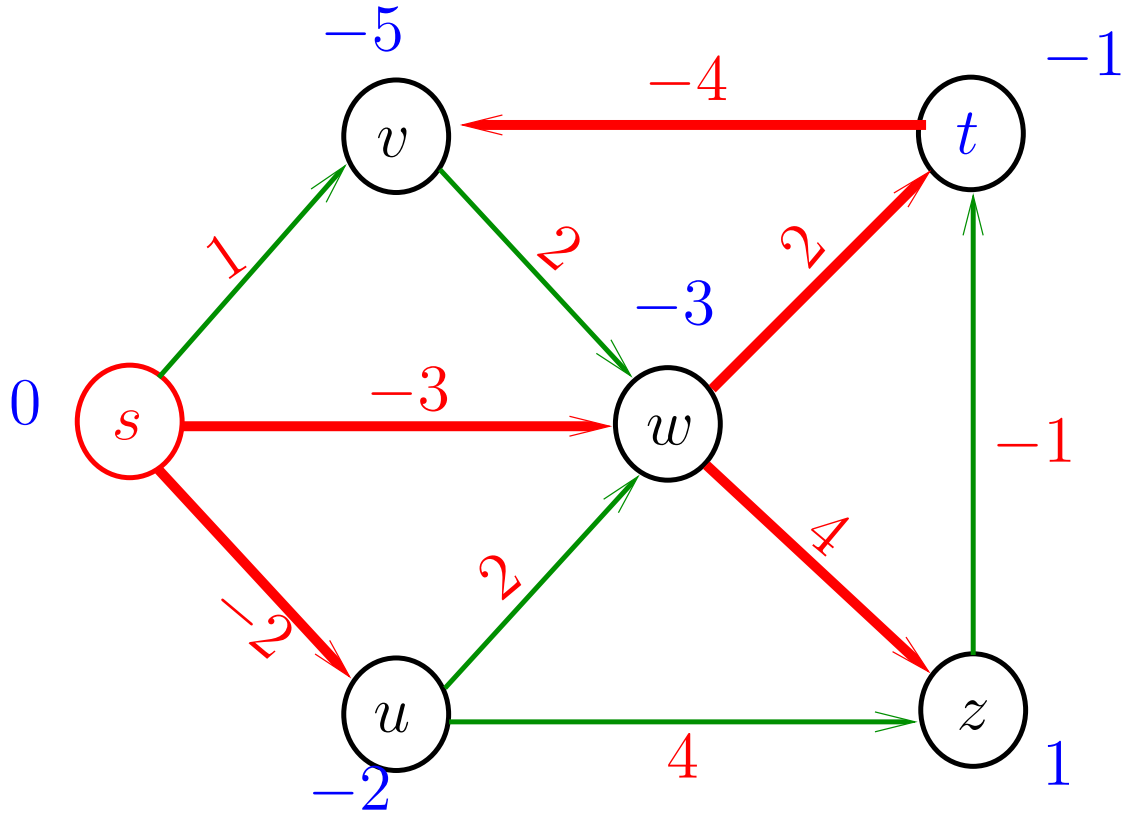


Conseqüência

Se P é um caminho de s a t e y é um c-potencial tais que

$$c(P) = y(t) - y(s),$$

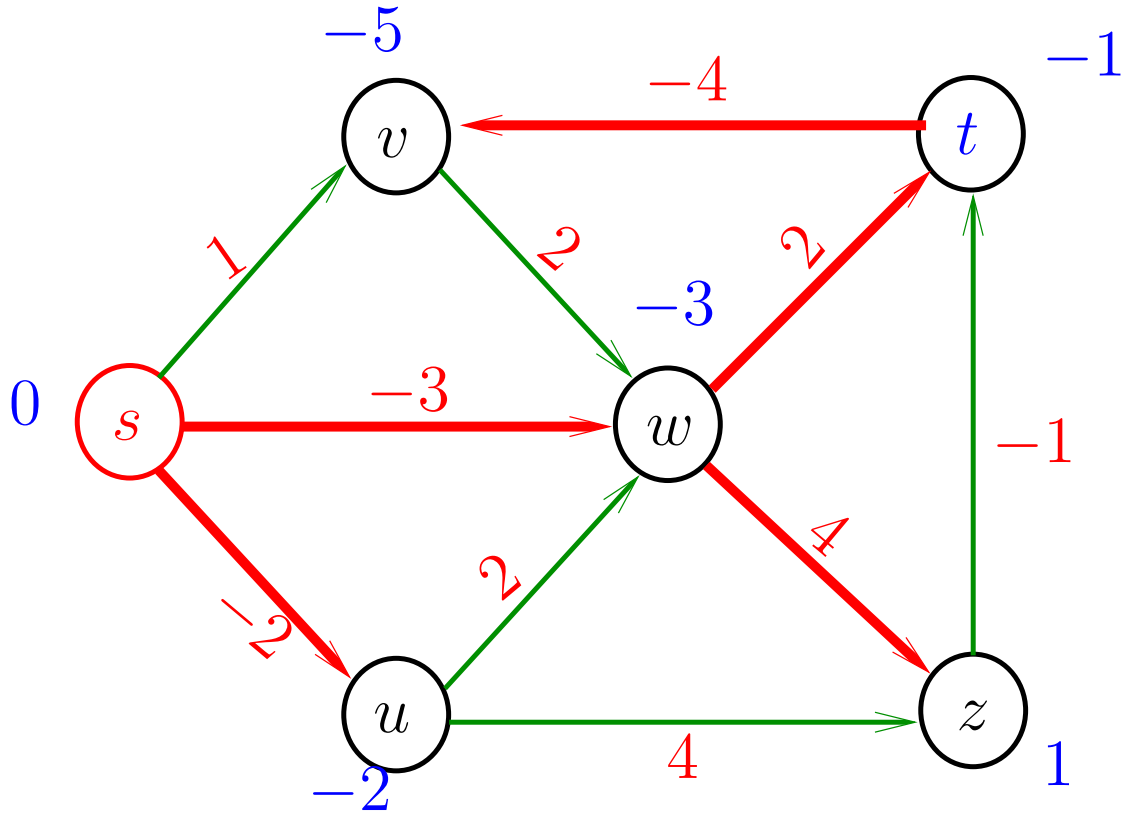
então P é uma caminho mínimo e y é um c-potencial tal que o valor de $y(t) - y(s)$ é máximo.



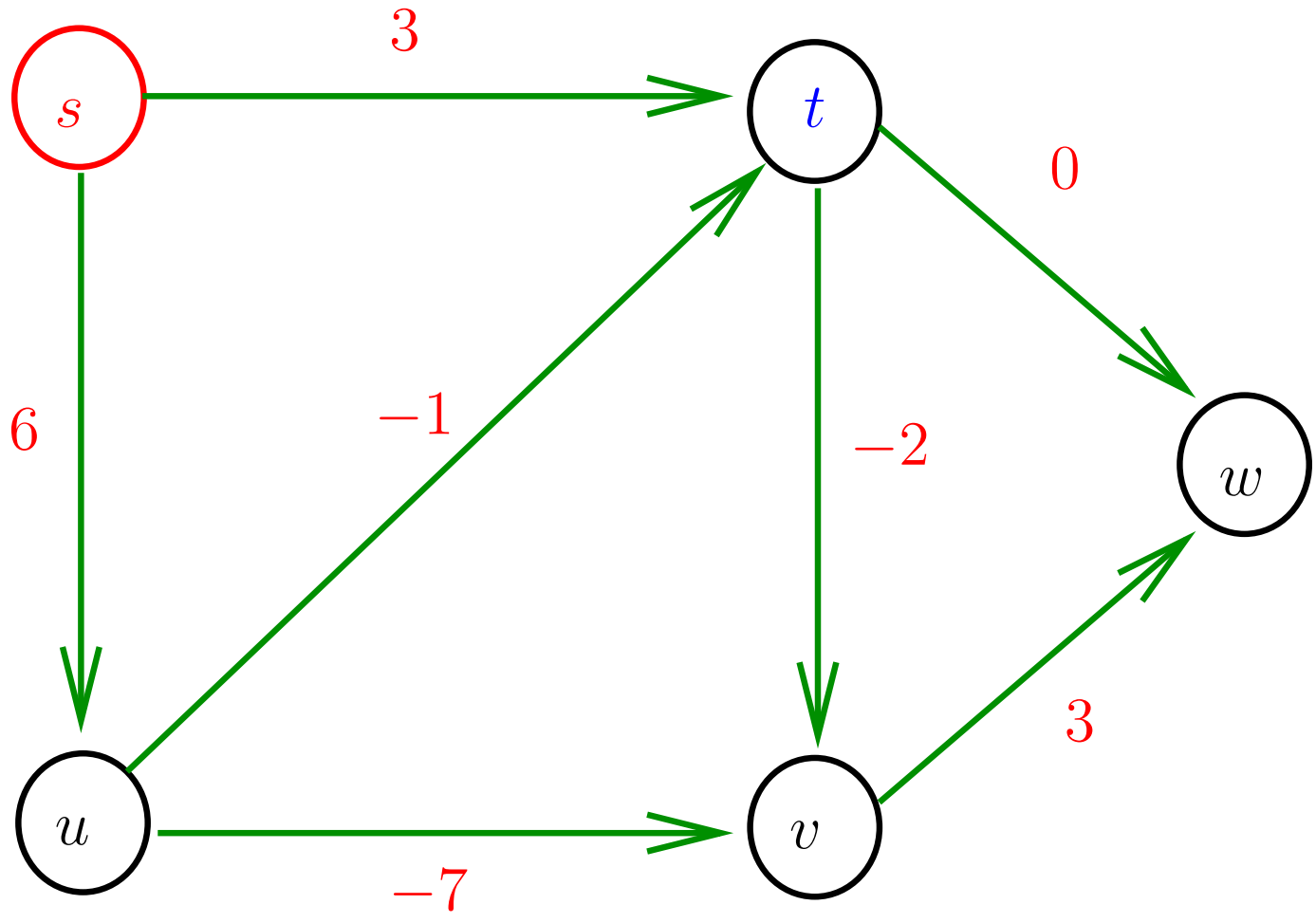
Potenciais ótimos

Um c-potencial y é $(s, *)$-ótimo se, para todo nó t, existe um caminho P de s a t tal que

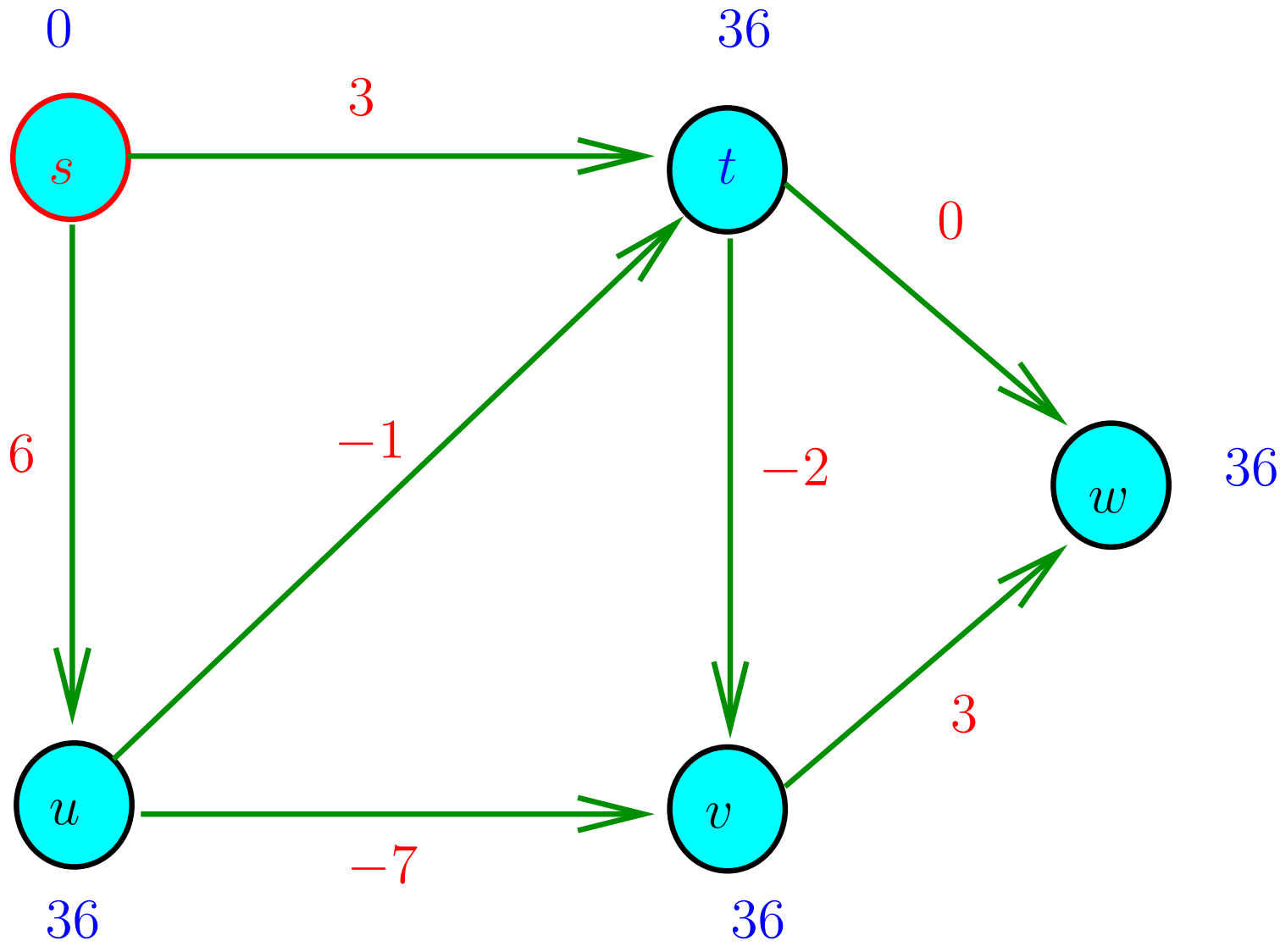
$$c(P) = y(t) - y(s),$$



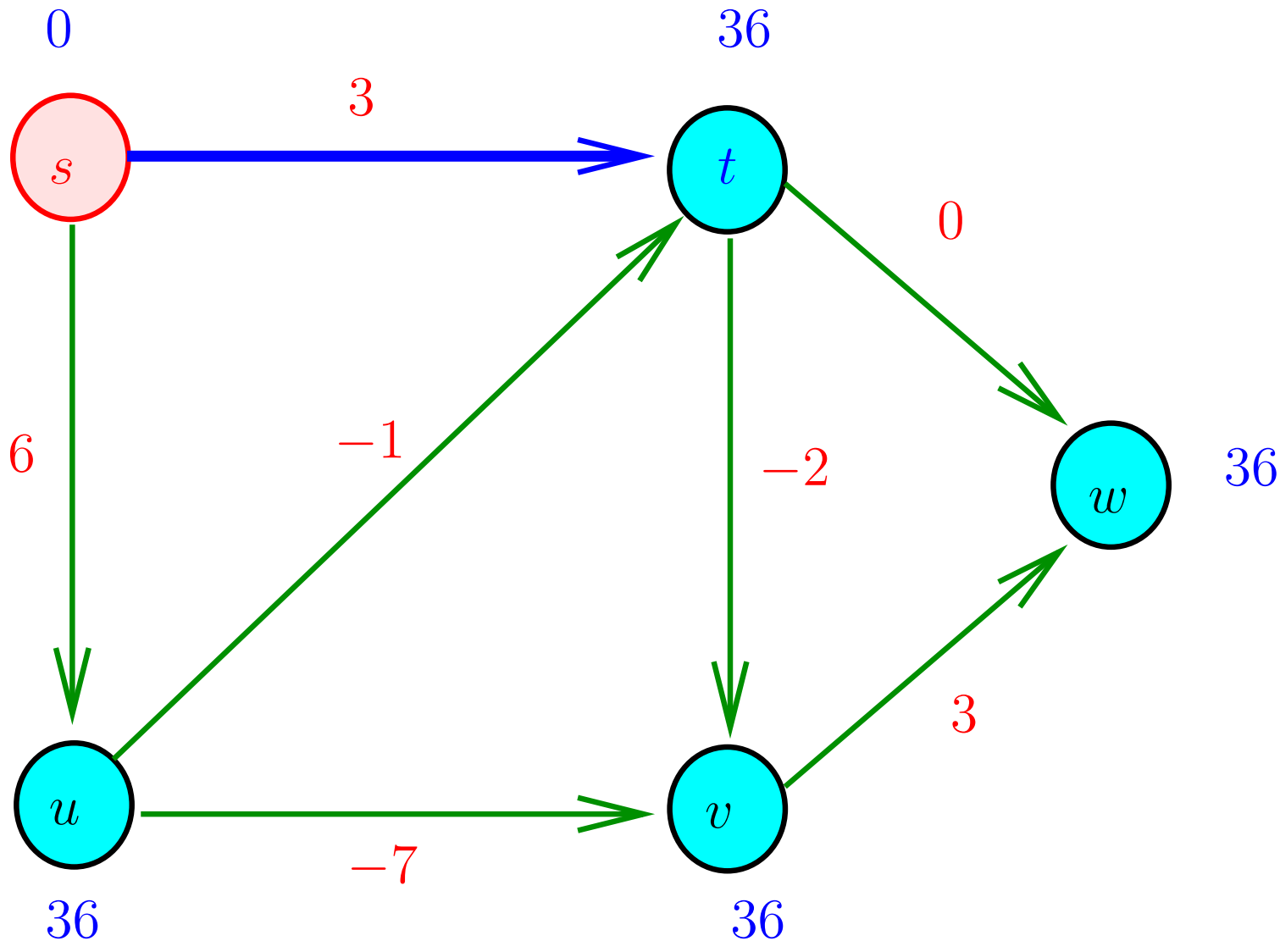
Apelemos para Dijkstra



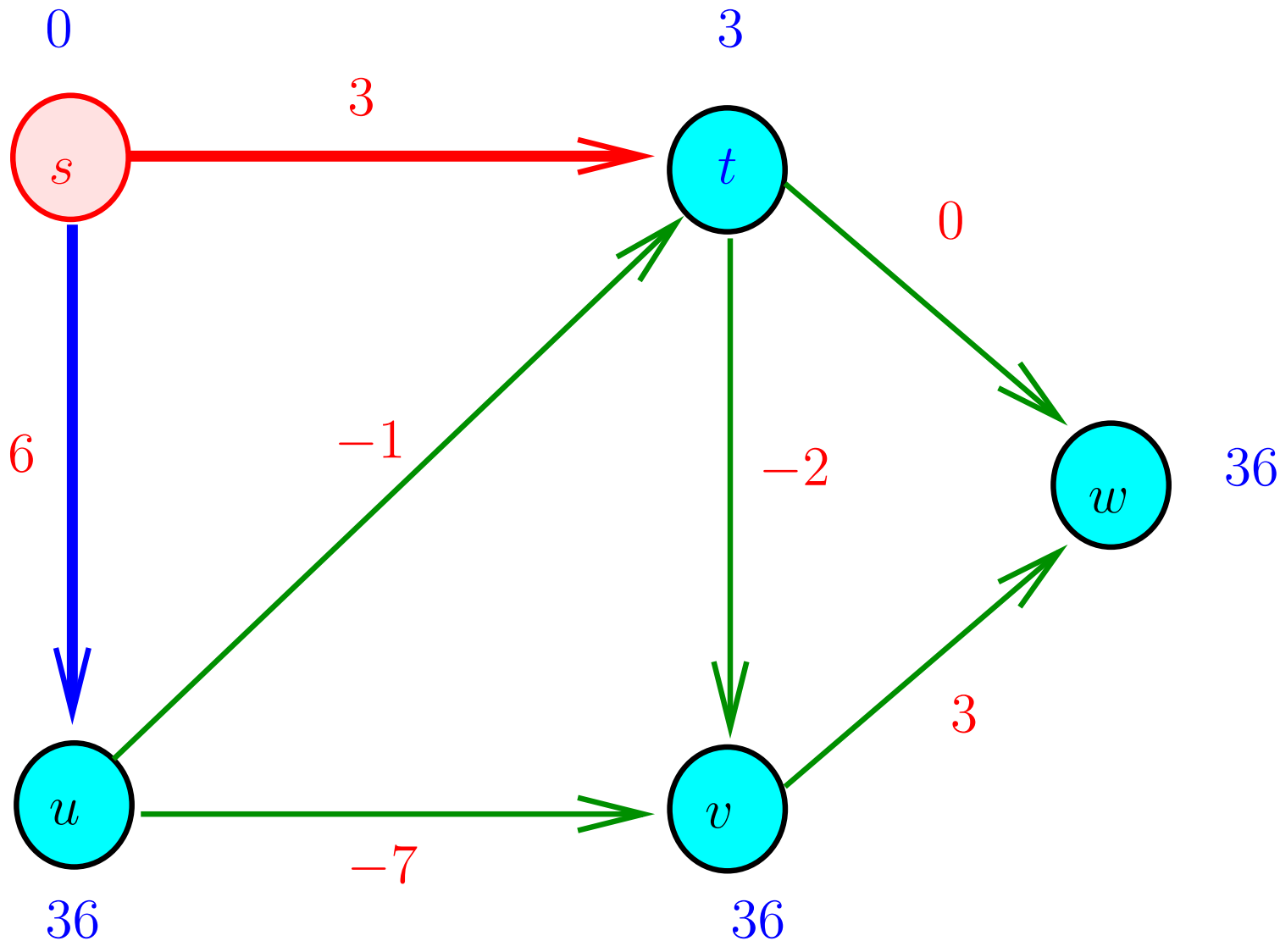
Apelemos para Dijkstra



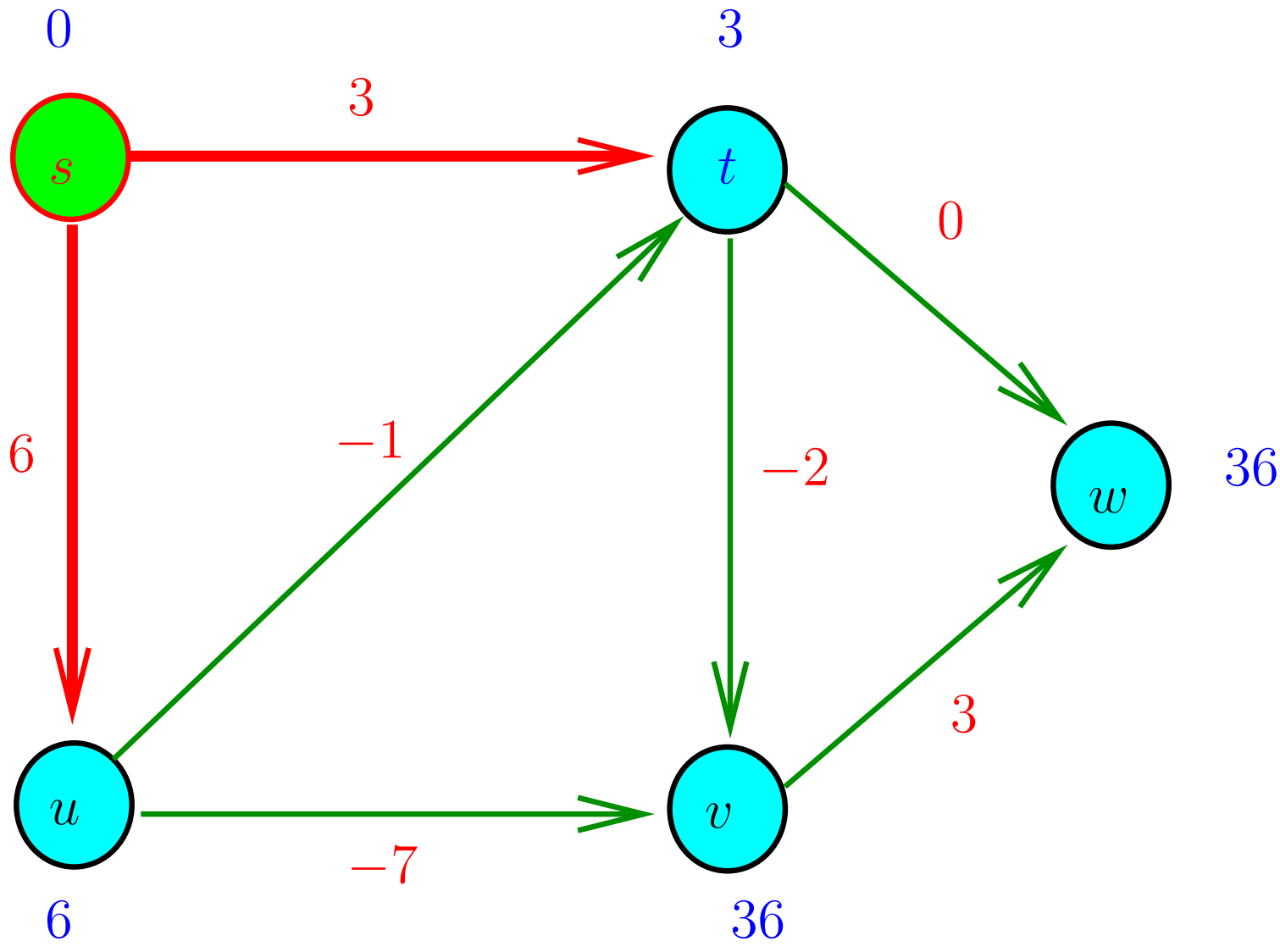
Apelemos para Dijkstra



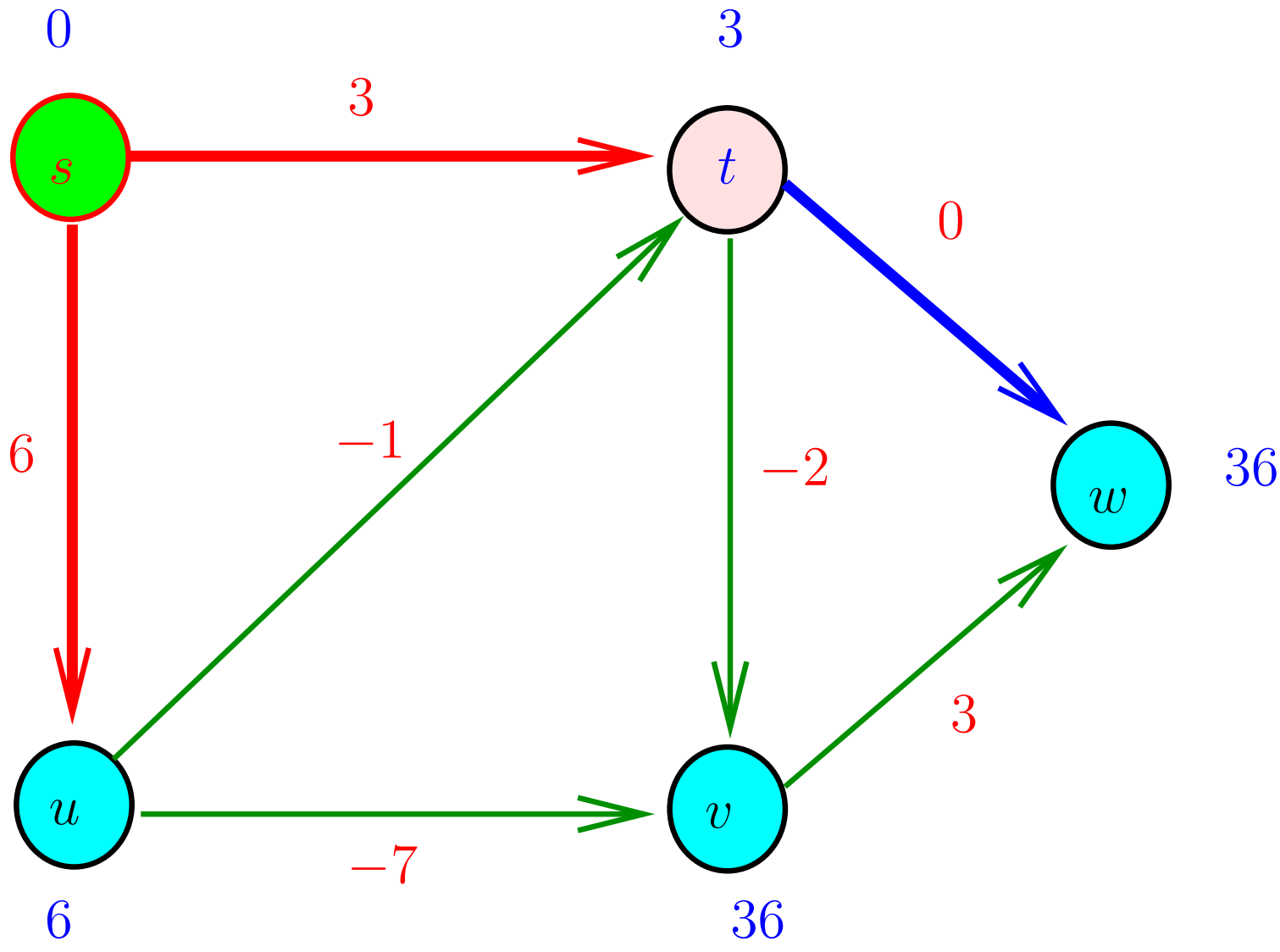
Apelemos para Dijkstra



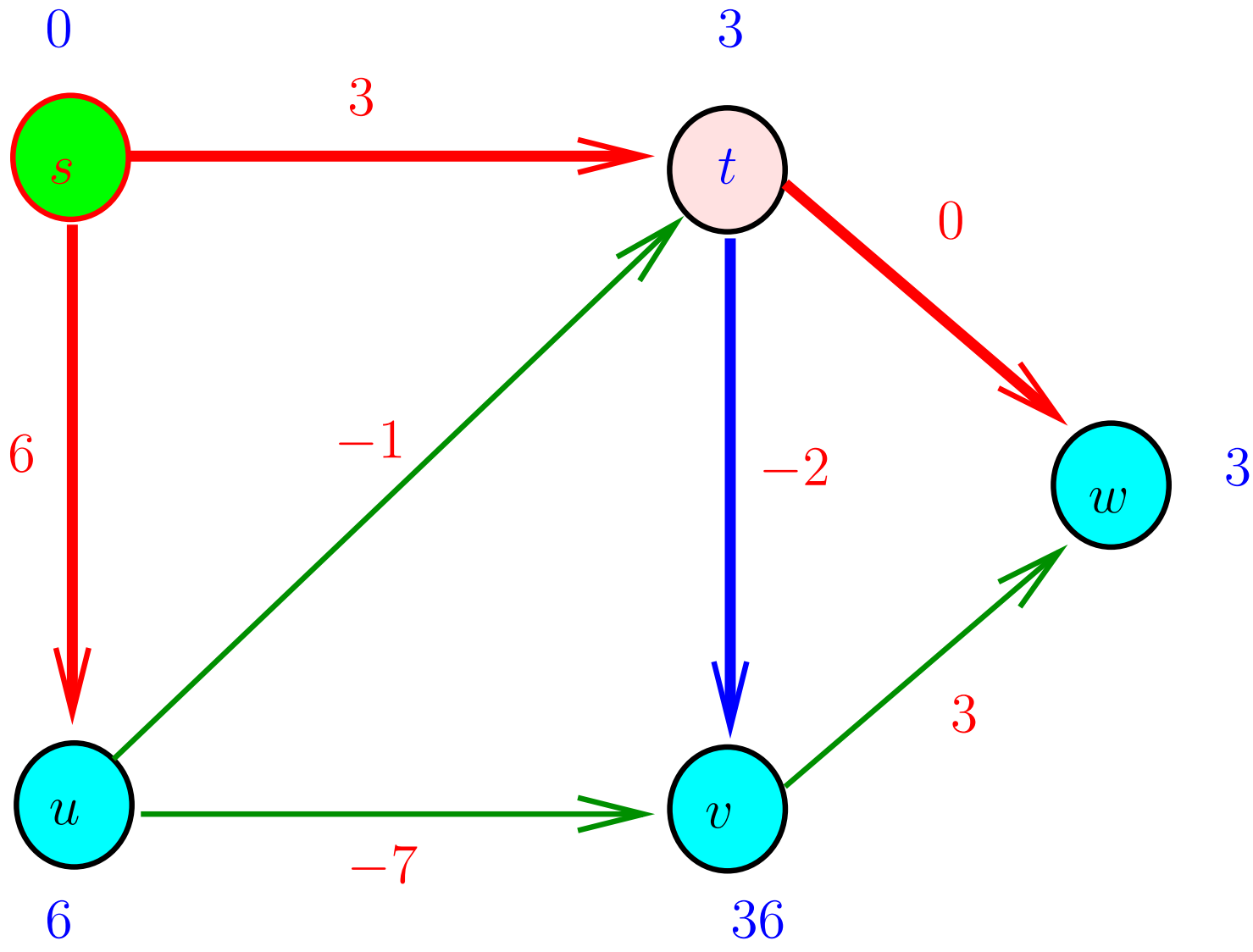
Apelemos para Dijkstra



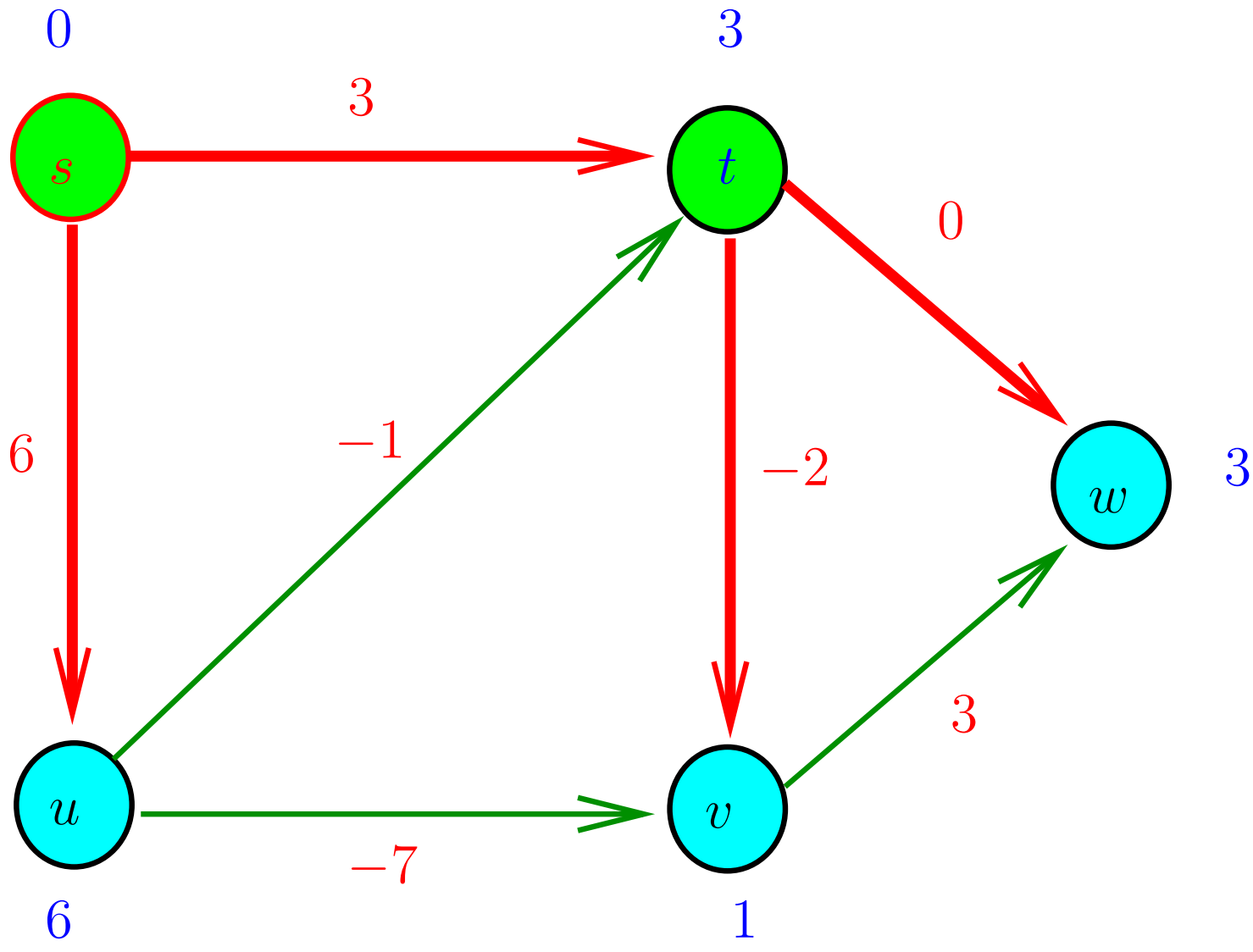
Apelemos para Dijkstra



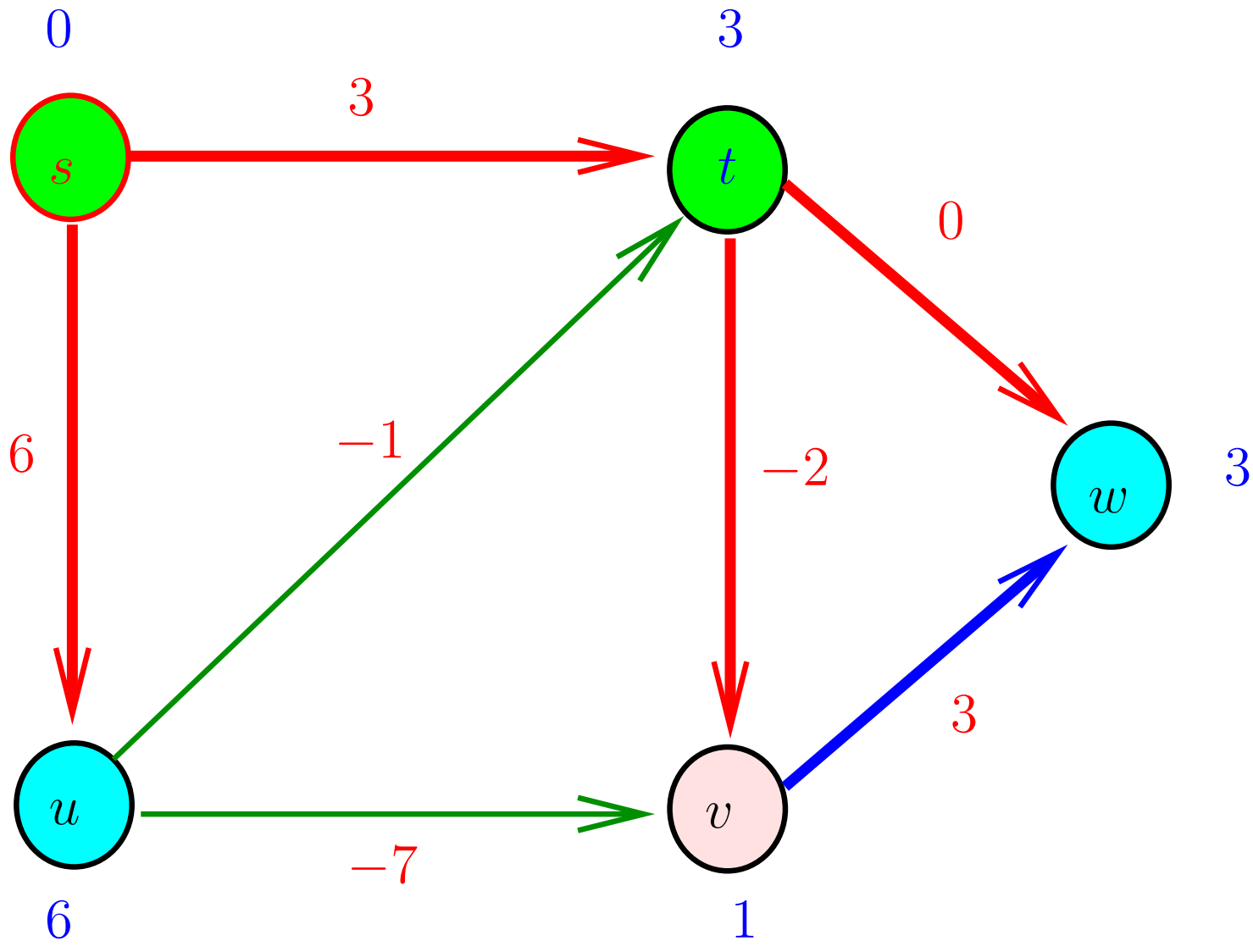
Apelemos para Dijkstra



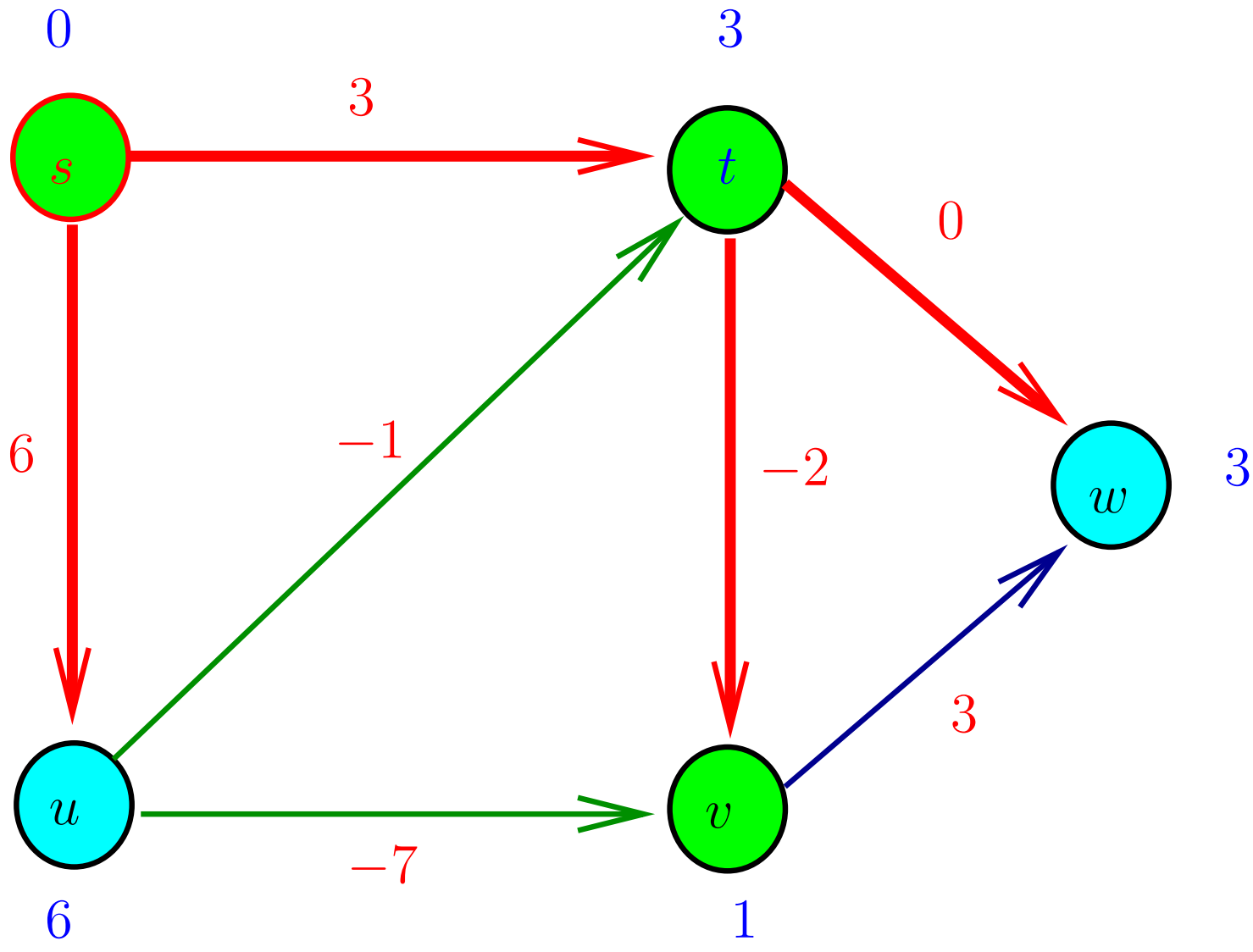
Apelemos para Dijkstra



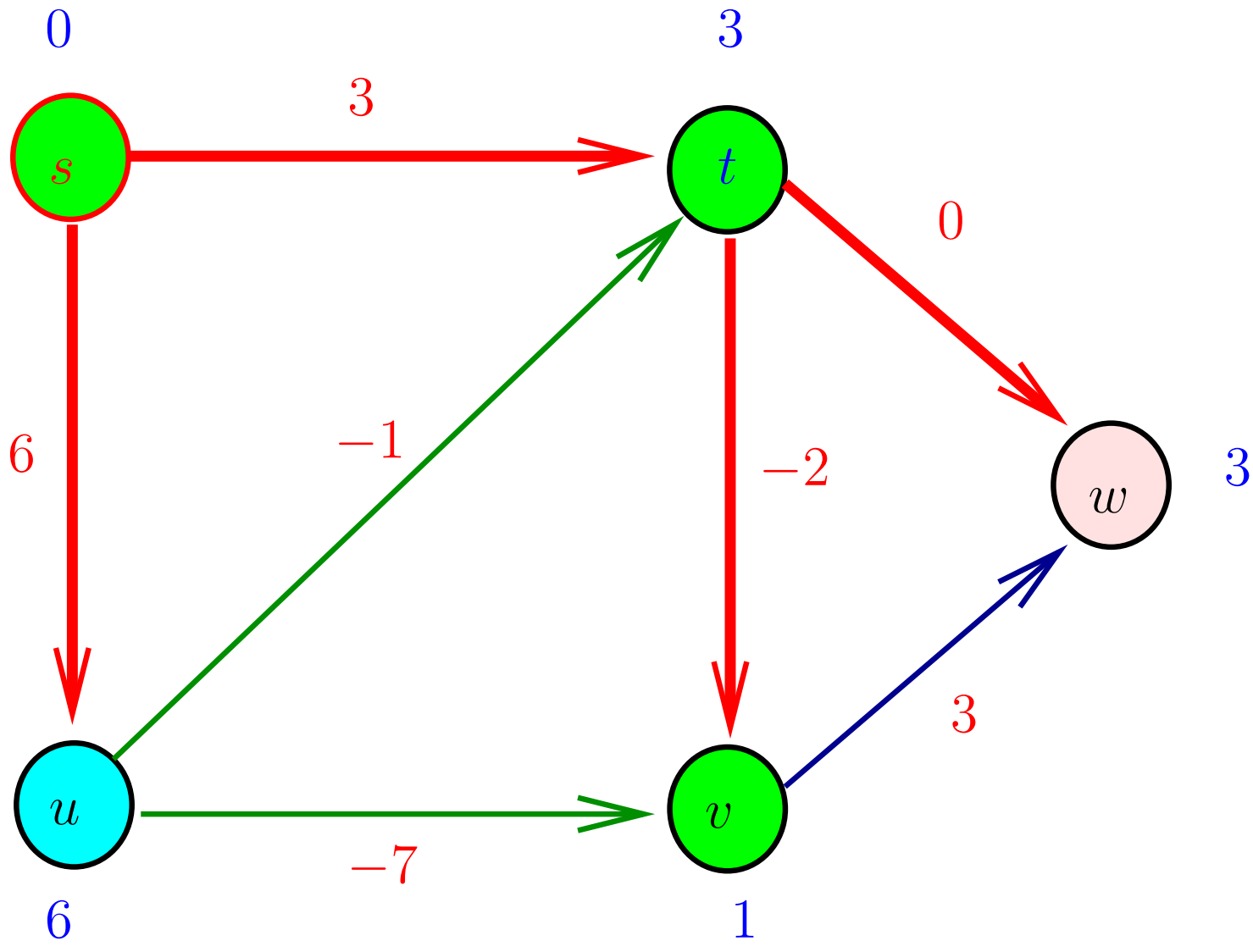
Apelemos para Dijkstra



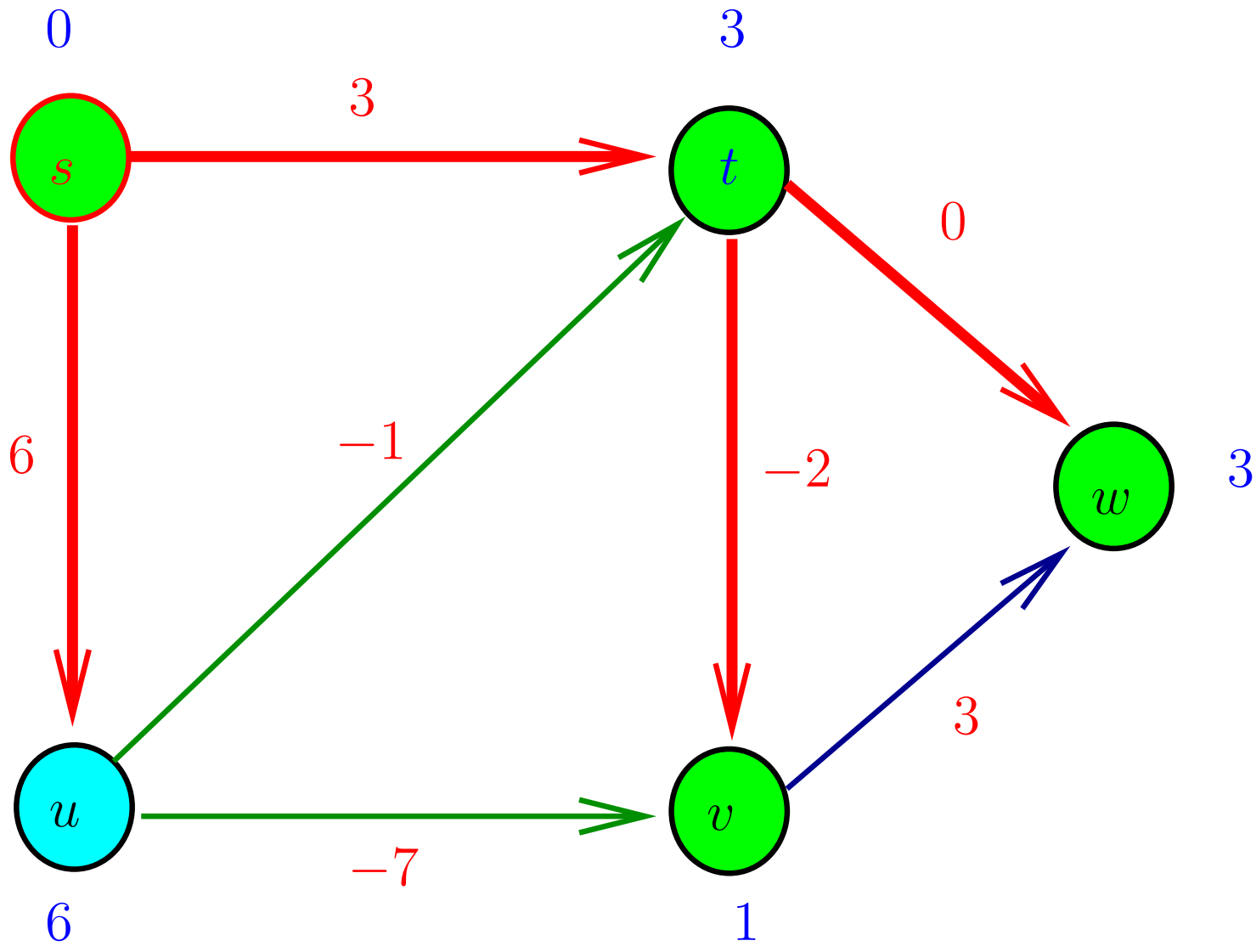
Apelemos para Dijkstra



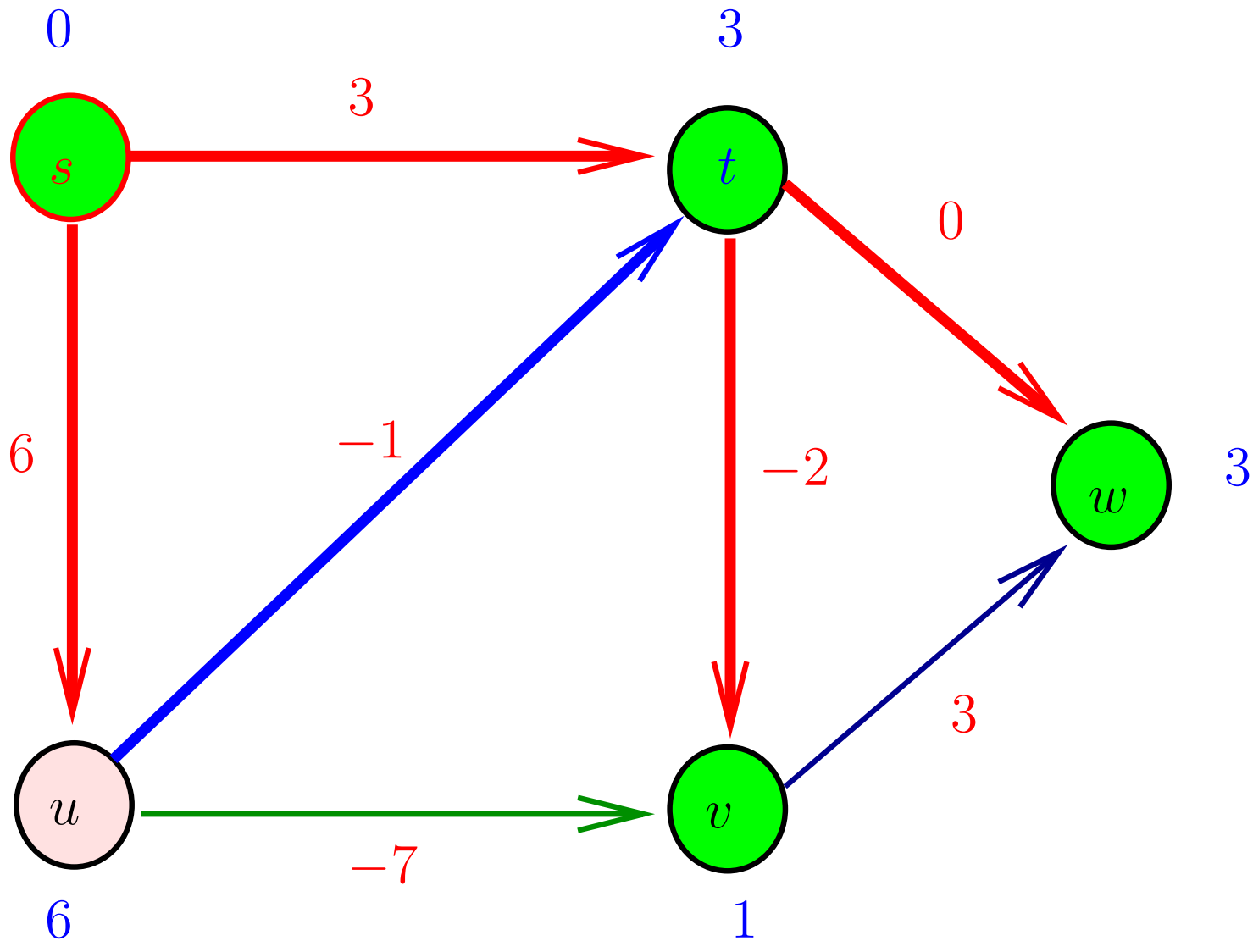
Apelemos para Dijkstra



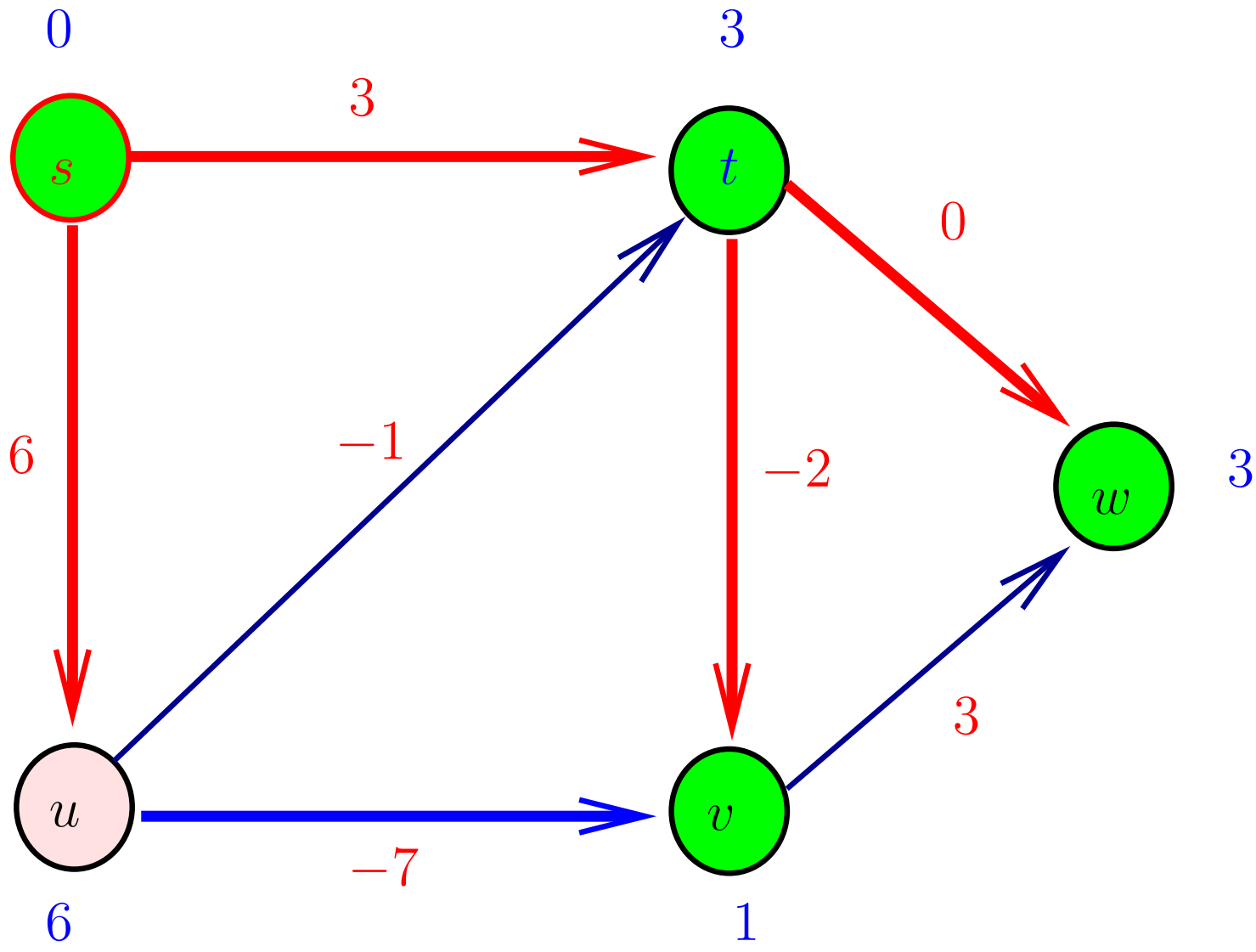
Apelemos para Dijkstra



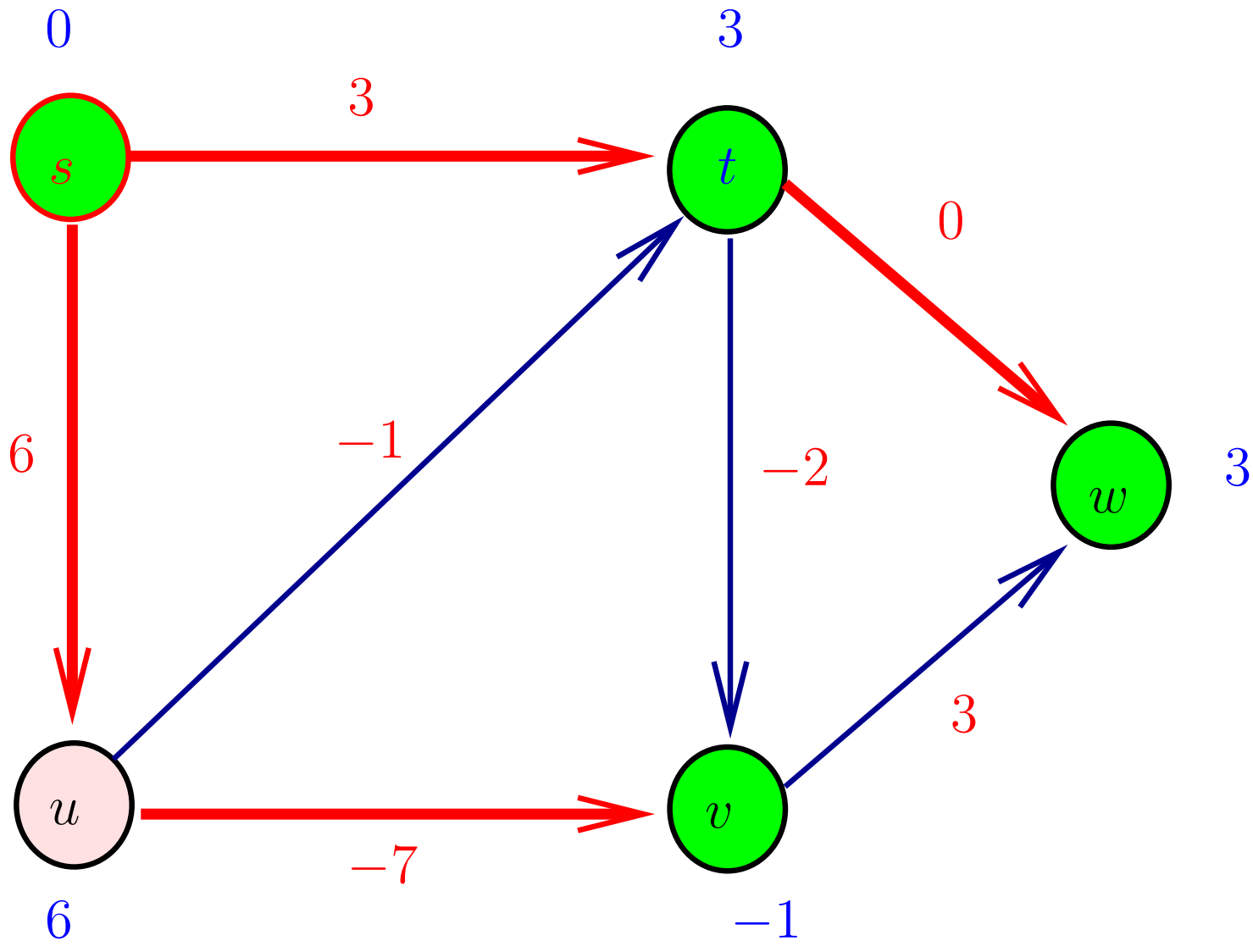
Apelemos para Dijkstra



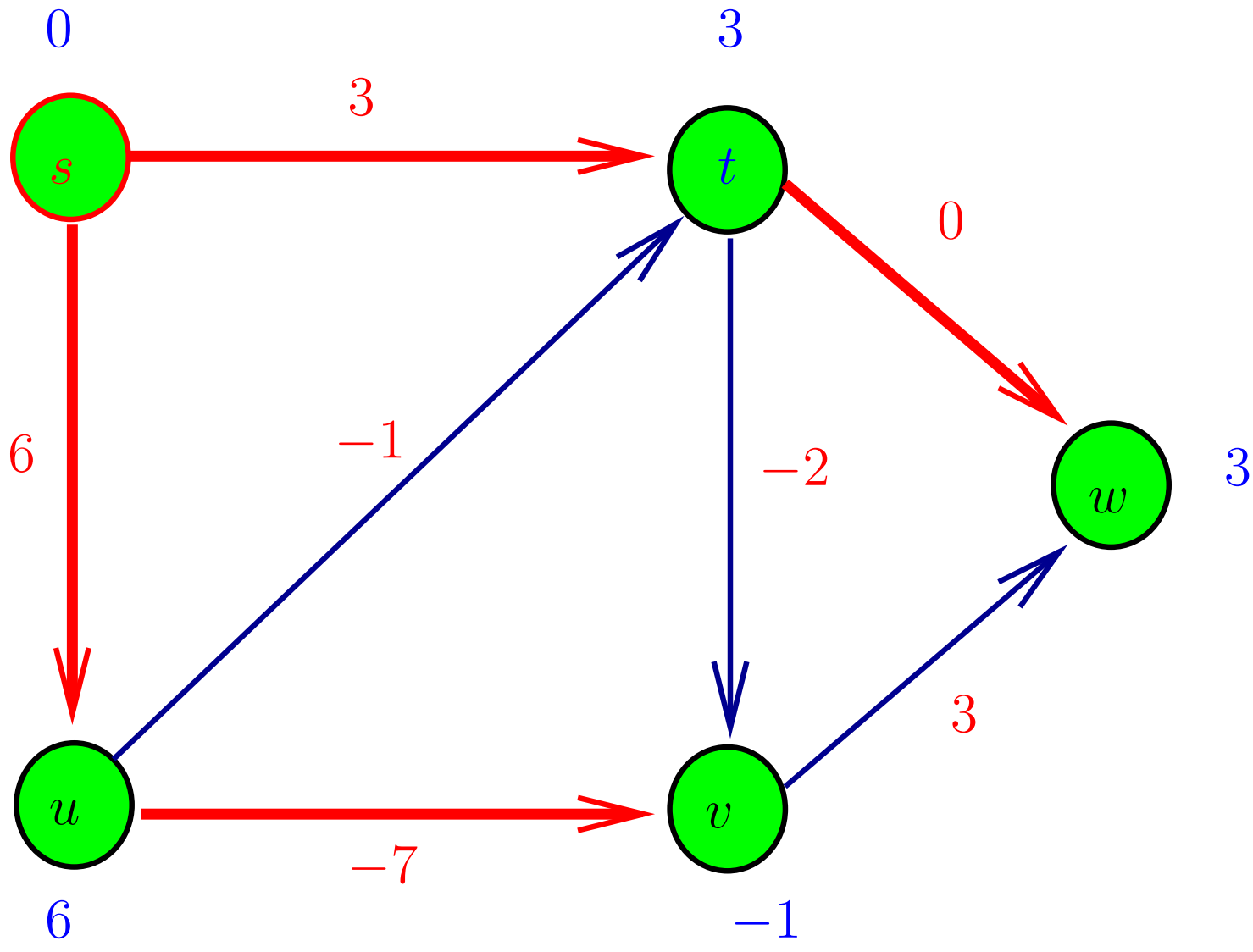
Apelemos para Dijkstra



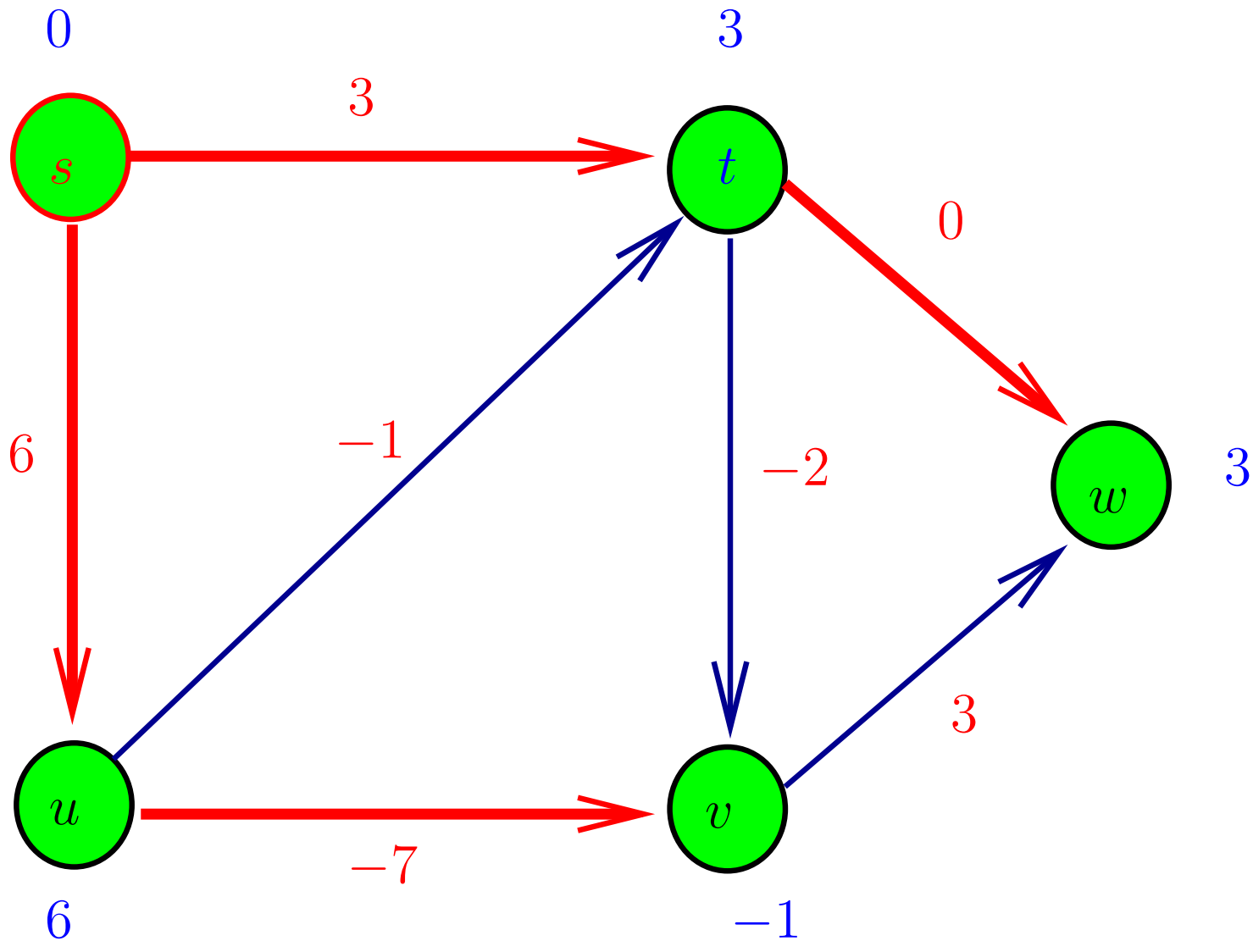
Apelemos para Dijkstra



Apelemos para Dijkstra



Apelemos para Dijkstra



O sw-caminho mínimo tem custo 2 e **não** 3...

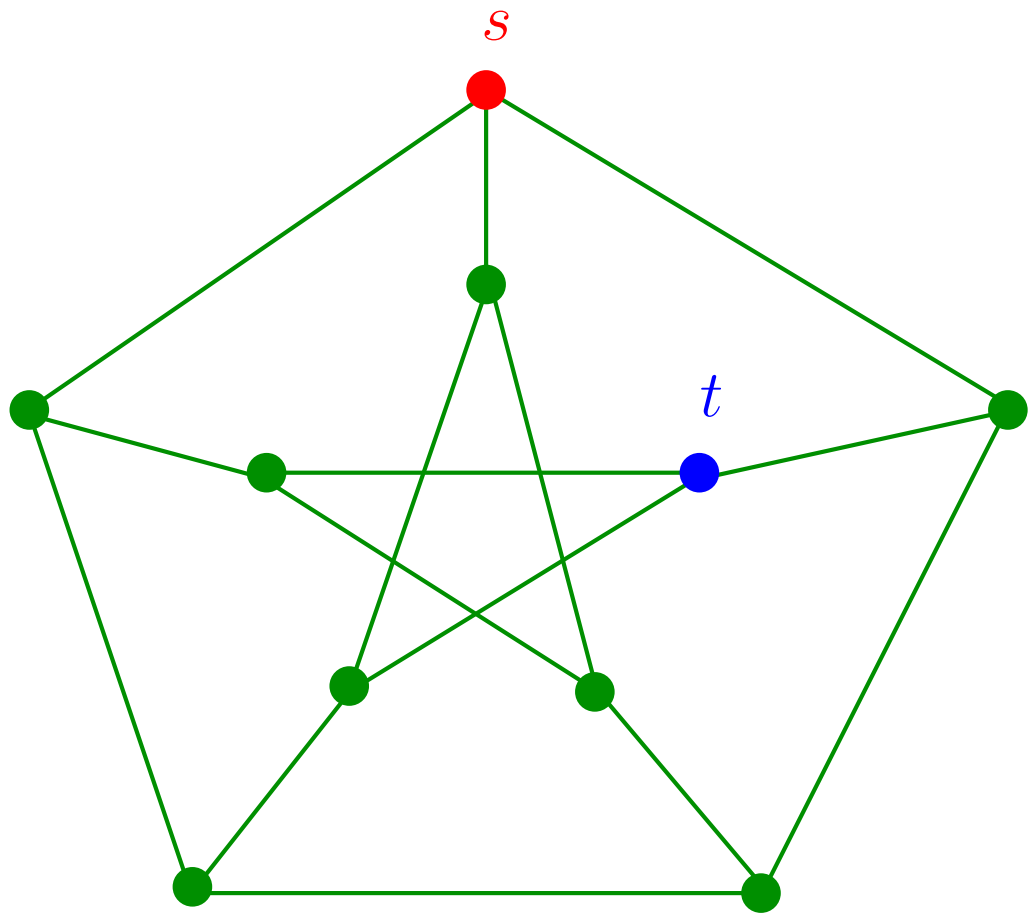
Caminhos hamiltonianos

Problema: Dados nós s e t de grafo não-orientado (N, E) encontrar um caminho hamiltoniano entre s e t.

Caminhos hamiltonianos

Problema: Dados nós s e t de grafo não-orientado (N, E) encontrar um caminho hamiltoniano entre s e t.

G

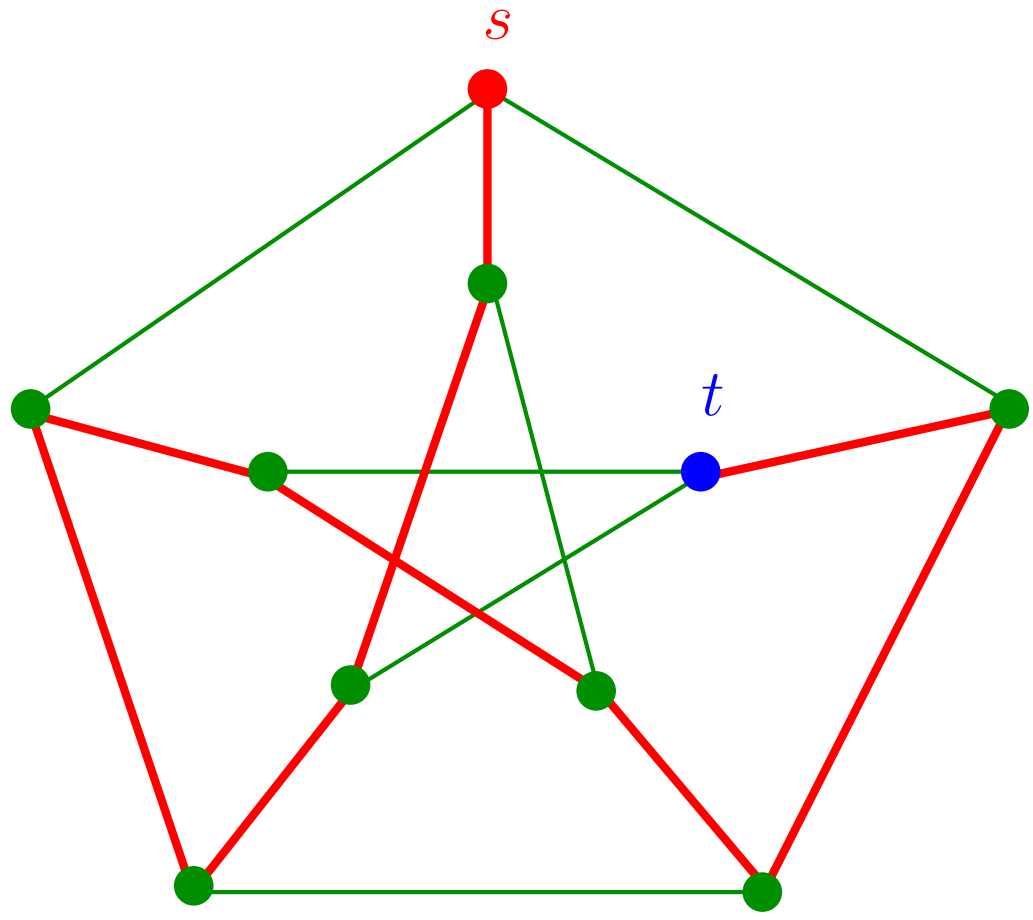


Este problema é NP-difícil.

Caminhos hamiltonianos

Problema: Dados nós s e t de grafo não-orientado (N, E) encontrar um caminho hamiltoniano entre s e t.

G

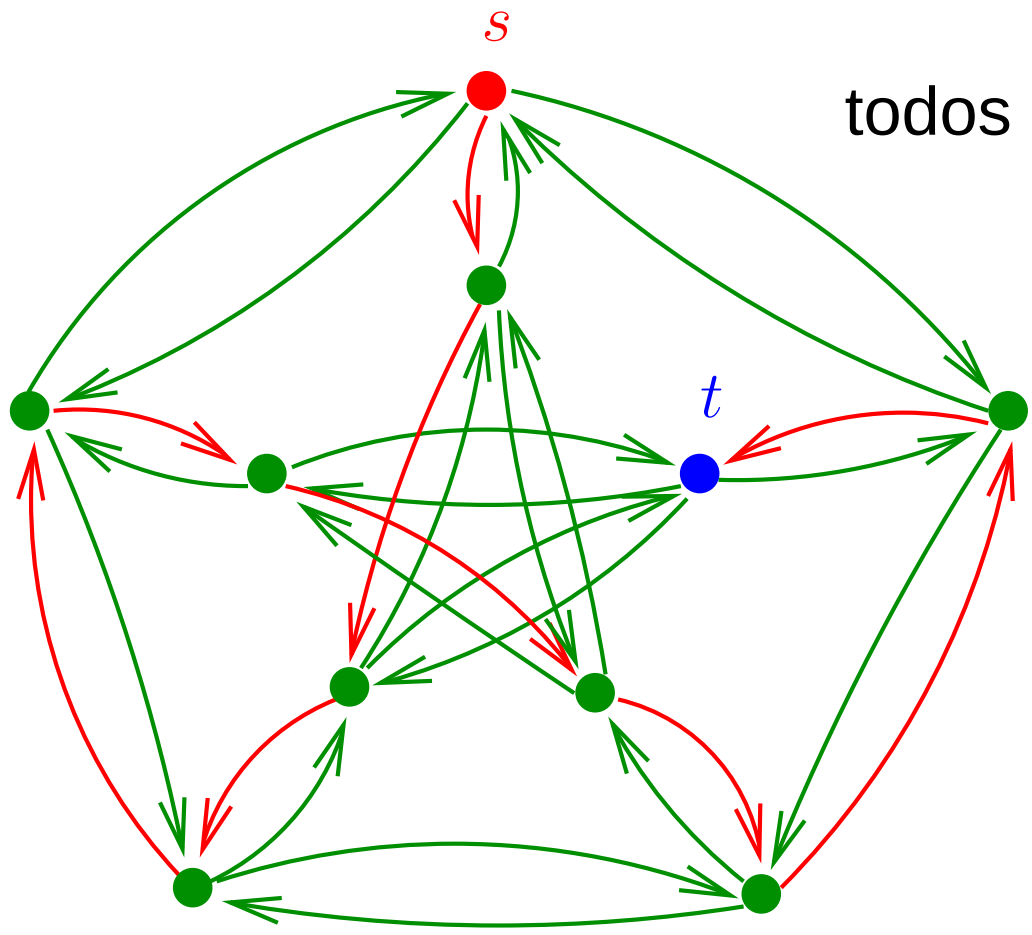


Este problema é NP-difícil.

Redução polinomial

todos custos = -1

G



G possui um st-caminho hamiltoniano

$\Leftrightarrow$

(N, A) possui um st-caminho de custo $-(n-1)$.

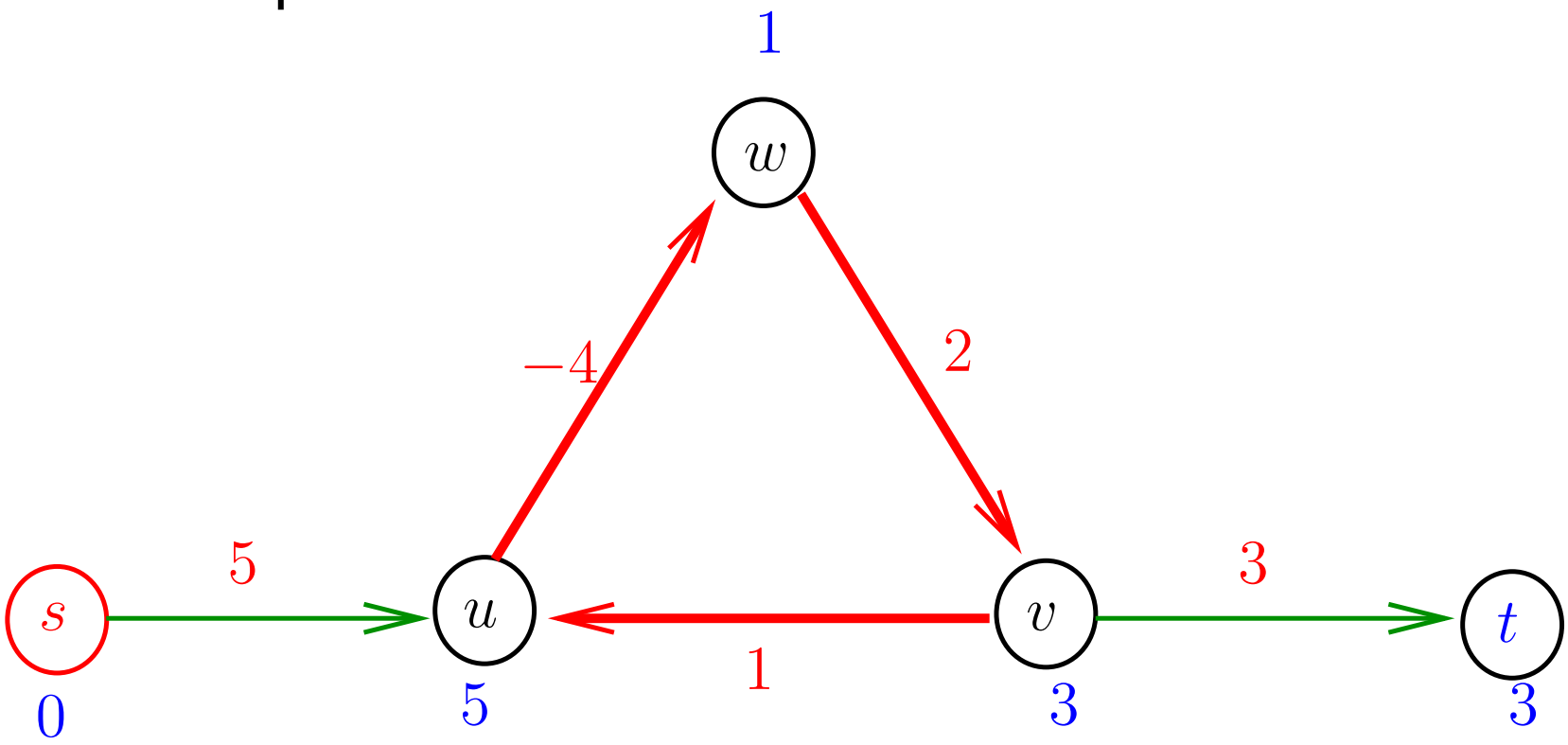
Conclusão

O problema do caminho mínimo é tão difícil quanto o problema do caminho hamiltoniano.

O problema do caminho de custo mínimo é NP-difícil.

Ciclos negativos

Se a rede possui um ciclo (de custo) negativo então não existe um c-potencial.



Vamos supor que...

Vamos supor que:

- a rede não tem ciclo negativo
- todo nó da rede está ao alcance de s
- $C := \max\{|c(ij)| : ij \in A\}$

Algoritmo genérico

Recebe uma rede (N, A, c) com função custo $c : A \to \mathbb{Z}$ sem ciclos negativos e um nó s e devolve uma função-predecessor π e um c-potencial y com a seguinte propriedade: para todo nó t, a função π determina um caminho P de s a t tal que $c(P) = y(t) - y(s)$.

FORD (N, A, c, s)

1 **para cada** i em N **faça**
2 $\quad y(i) \leftarrow nC + 1 \quad \triangleright\ C := \max\{|c(ij)| : ij \in A\}$
3 $\quad \pi(i) \leftarrow \text{NIL}$
4 $y(s) \leftarrow 0$
5 **enquanto** $y(j) > y(i) + c(ij)$ **para** algum $ij \in A$ **faça**
6 $\quad y(j) \leftarrow y(i) + c(ij)$
7 $\quad \pi(j) \leftarrow i$
8 **devolva** π e y

Invariantes

Na linha 5, antes da verificação da condição "$y(j) > y(i) + c(ij) \ldots$" valem as seguintes invariantes:

(i1) para cada arco pq no grafo de predecessores tem-se $y(q) - y(p) \geq c(pq)$;

(i2) se pq é um arco tal que $y(q) - y(p) \geq c(pq)$, então **não** há um qp-caminho no grafo de predecessores.

(i3) $\pi(s) = \text{NIL}$ e $y(s) = 0$;

(i4) se $y(t) < \infty$, então há um st-caminho no grafo de predecessores.

(i5) Se $S := \{v : y(v) < \infty\}$, então, para cada v em S temos que $y(v) < |S|C'$, onde

$$C' := \max\{|c(pq)| : pq \in A, p \in S, q \in S\}.$$

Correção

Início da última iteração:

- y é um c-potencial
- Se $y(t) < \infty$ então, por (i4), segue que existe um st-caminho P no grafo de predecessores. Logo, (i1) e (i3) implicam que

$$c(P) \geq y(t) - y(s) = y(t).$$

  Da propriedade dos c-potencias, concluímos que P é um st-caminho (de custo) mínimo.

Conclusão: o algoritmo faz o que promete.

Conclusão

Da propriedade dos c-potenciais (lema da dualidade) e da correção do algoritmo FORD concluímos o seguinte:

(Teorema dualidade) Se s e t são nós de uma rede (N, A, c) com função custo $c : A \to \mathbb{Z}$, sem ciclos negativos, e t está ao alcance de s então

$$\min\{c(P) : P \text{ é um } st\text{-caminho}\}$$
$$= \max\{y(t) - y(s) : y \text{ é um } c\text{-potencial}\}.$$

Consumo de tempo

O número de execuções do bloco de linhas 5–7 é

$$< n(2\,nC + 1) = 2\,n^2C + n.$$

linha	consumo de todas as execuções da linha
1-3	$\mathrm{O}(n)$
4	$\mathrm{O}(1)$
5	$(2\,n^2C + n)\mathrm{O}(m) = \mathrm{O}(n^2m(C+1))$
6-7	$(2\,n^2C + n)\mathrm{O}(1) = \mathrm{O}(n^2(C+1))$
8	$\mathrm{O}(n)$
total	$2\,\mathrm{O}(n) + \mathrm{O}(1) + \mathrm{O}(n^2m(C+1)) + \mathrm{O}(n^2(C+1))$ $= \mathrm{O}(n^2m(C+1))$

Conclusão

O consumo de tempo do algoritmo CAMINHO-CURTO-GENÉRICO é $\mathrm{O}(n^2 m(C+1))$.

Este consumo de tempo não é polinomial.