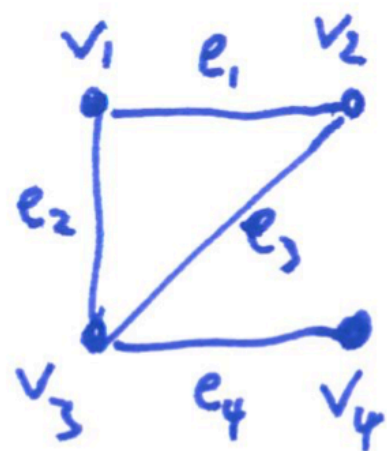


Kapitel 3: Suche in Graphen

Algorithmen und Datenstrukturen
WS 2025/26

Prof. Dr. Sándor Fekete

(4) Adjazenzliste



$V_1: V_2, V_3;$

$V_2: V_1, V_3;$

$V_3: V_1, V_2, V_4;$

$V_4: V_3;$

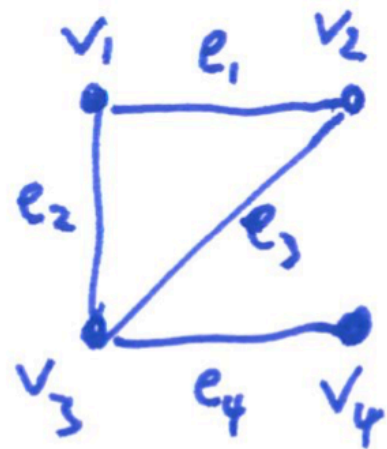
↓ ↓ ↓ ↓
 $V_2, V_3; \quad V_1, V_3; \quad V_1, V_2, V_4; \quad V_3$

$$\log_2 \left\lceil \left(2n + 4m + n(\lfloor \log_2 n \rfloor + 1) + 2m(\lfloor \log_2 n \rfloor + 1) \right) \right\rceil + 1$$

$$\left\lceil \log_{10} \left(2n + 4m + n \left(\lfloor \log_{10} n \rfloor + 1 \right) + 2m \left(\lfloor \log_{10} n \rfloor + 1 \right) \right) \right\rceil + 1$$

$$\leq \log_2 \left(9m \left(n + 1 \right) + 1 \right)$$

(4) Adjazenzliste



$V_1: V_2, V_3;$

$V_2: V_1, V_3;$

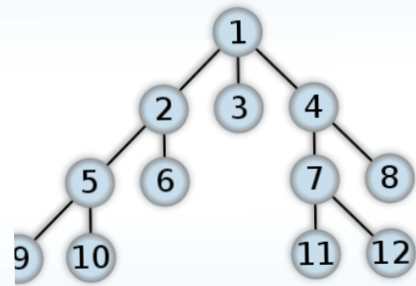
$V_3: V_1, V_2, V_4;$

$V_4: V_3;$

↓ ↓ ↓ ↓
 $V_2, V_3;$ $V_1, V_3;$ $V_1, V_2, V_4;$ V_3

$$\log_2 \left\lceil \left(2n + 4m + n(\lfloor \log_2 n \rfloor + 1) + 2m(\lfloor \log_2 n \rfloor + 1) \right) \right\rceil + 1$$

$$\Theta(n \log m + m \log n) = \Theta(m \log n)$$



Kapitel 3.8:
Laufzeit von DFS und BFS
Algorithmen und Datenstrukturen
WS 2025/26

Prof. Dr. Sándor Fekete

Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$
2. WHILE ($R \neq \emptyset$) DO {
 - 2.1. Wähle $v \in R$
 - 2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN
 - 2.2.1. $R := R \setminus \{v\}$
 - 2.3. ELSE {
 - 2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$
 - 2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$}}
3. STOP

Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

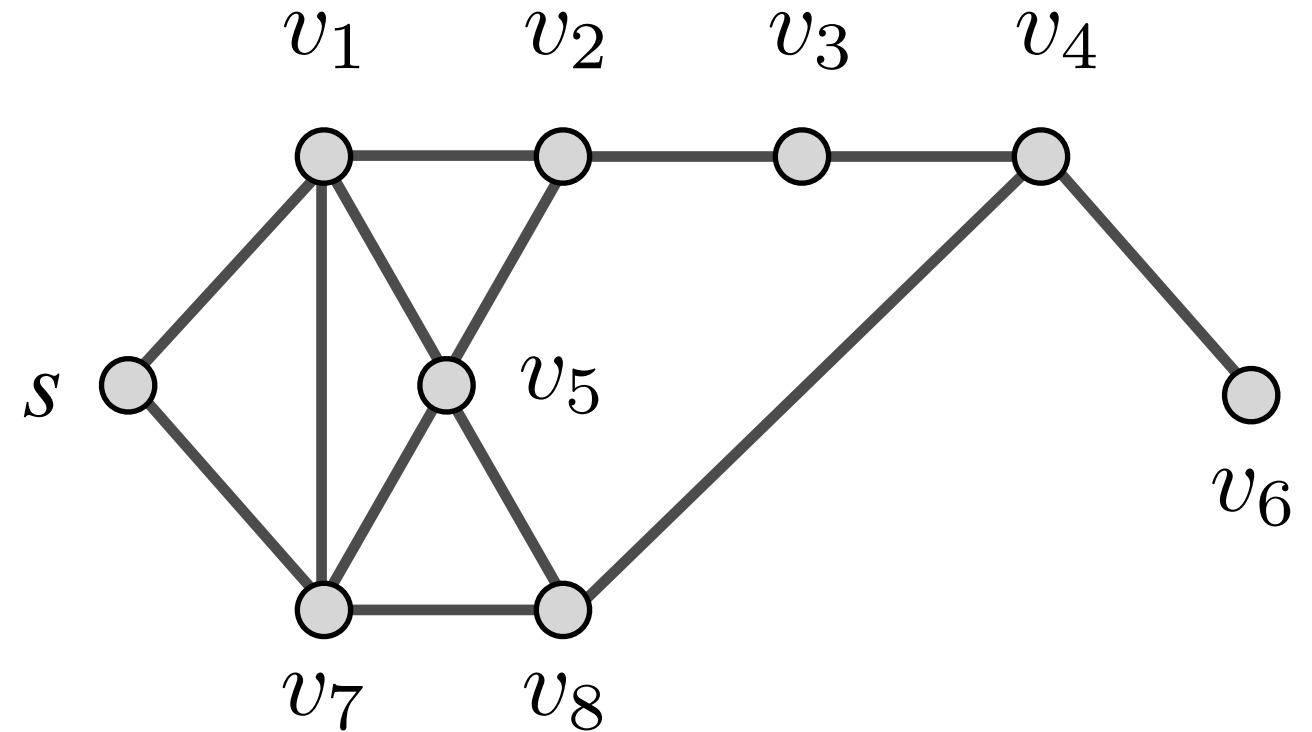
OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

```

1. Sei  $R := \{s\}$ ,  $Y := \{s\}$ ,  $T := \emptyset$ 
2. WHILE ( $R \neq \emptyset$ ) DO {
    2.1. Wähle  $v \in R$ 
    2.2. IF (es gibt kein  $w \in V \setminus Y$  mit  $e = \{v, w\} \in E$ ) THEN
        2.2.1.  $R := R \setminus \{v\}$ 
    2.3. ELSE {
        2.3.1. Wähle ein  $w \in V \setminus Y$  mit  $e = \{v, w\} \in E$ 
        2.3.2. Setze  $R := R \cup \{w\}$ ,  $Y := Y \cup \{w\}$ ,  $T := T \cup \{e\}$ 
    }
}
3. STOP

```

2



s v_1 v_2 v_3 v_4 v_5 $v_6 v_7$ v_8
 $\downarrow$ $\downarrow$ $\downarrow$ $\downarrow$ $\downarrow$ $\downarrow$ $\downarrow$ $\downarrow$
 $v_1 v_7 s v_7 v_5 v_2 v_1 v_5 v_3 v_2 v_4 v_3 v_8 v_6 v_1 v_2 v_7 v_8 v_4 s v_1 v_5 v_8 v_7 v_5 v_4 X$

Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

2. WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

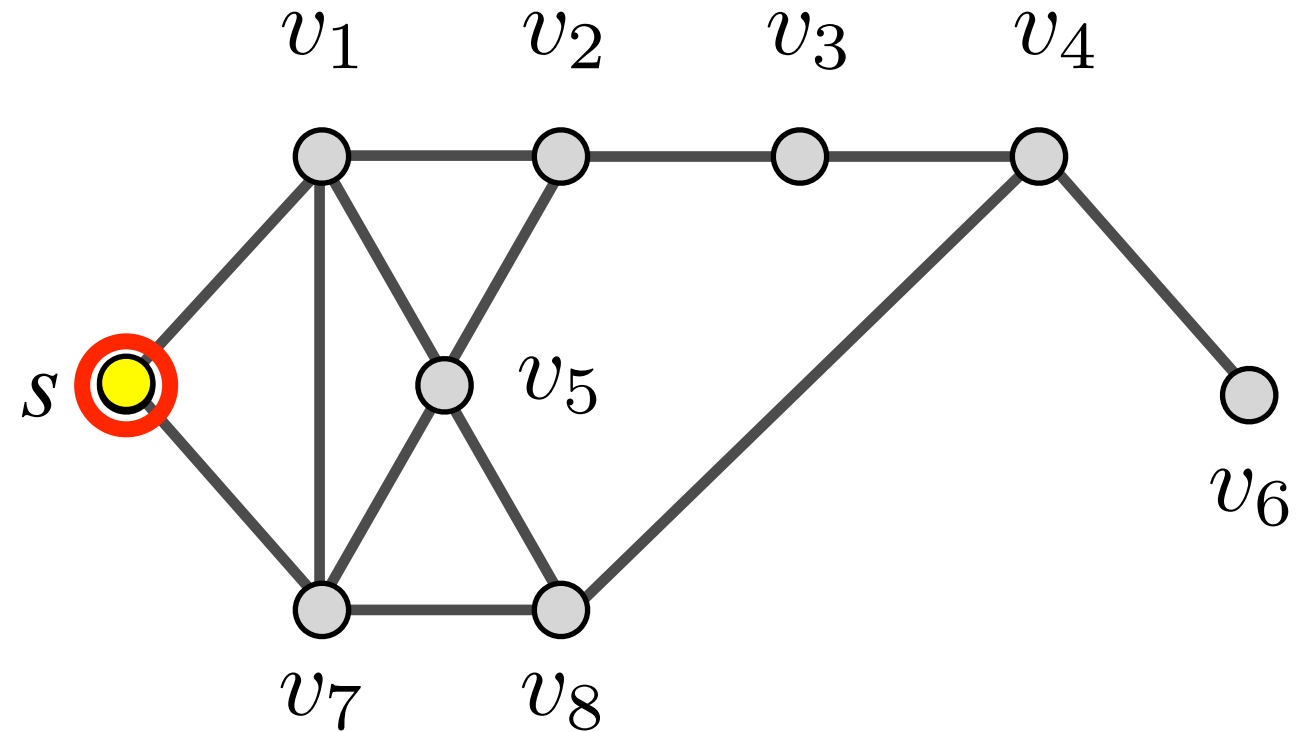
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



2

s

v_1

v_2

v_3

v_4

v_5

$v_6 v_7$

v_8

$v_1 v_7 s v_7 v_5 v_2 v_1 v_5 v_3 v_2 v_4 v_3 v_8 v_6 v_1 v_2 v_7 v_8 v_4 s v_1 v_5 v_8 v_7 v_5 v_4 X$

Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

2. WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

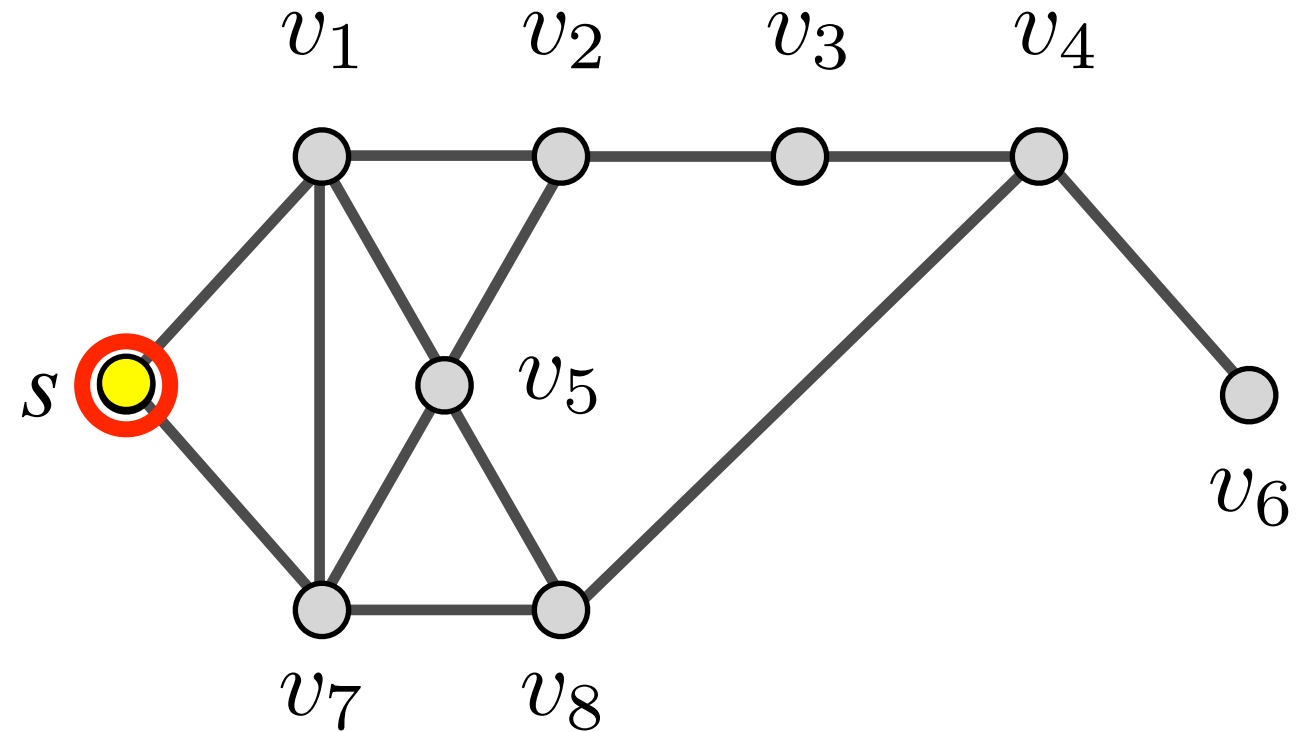
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

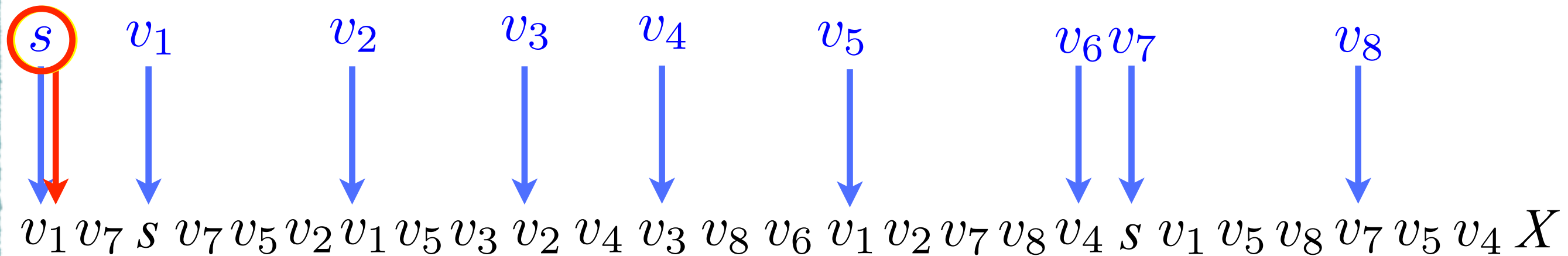
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

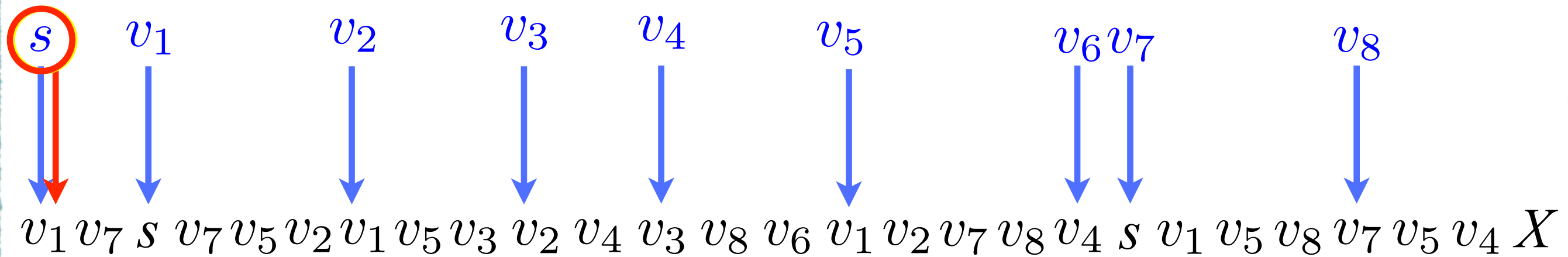
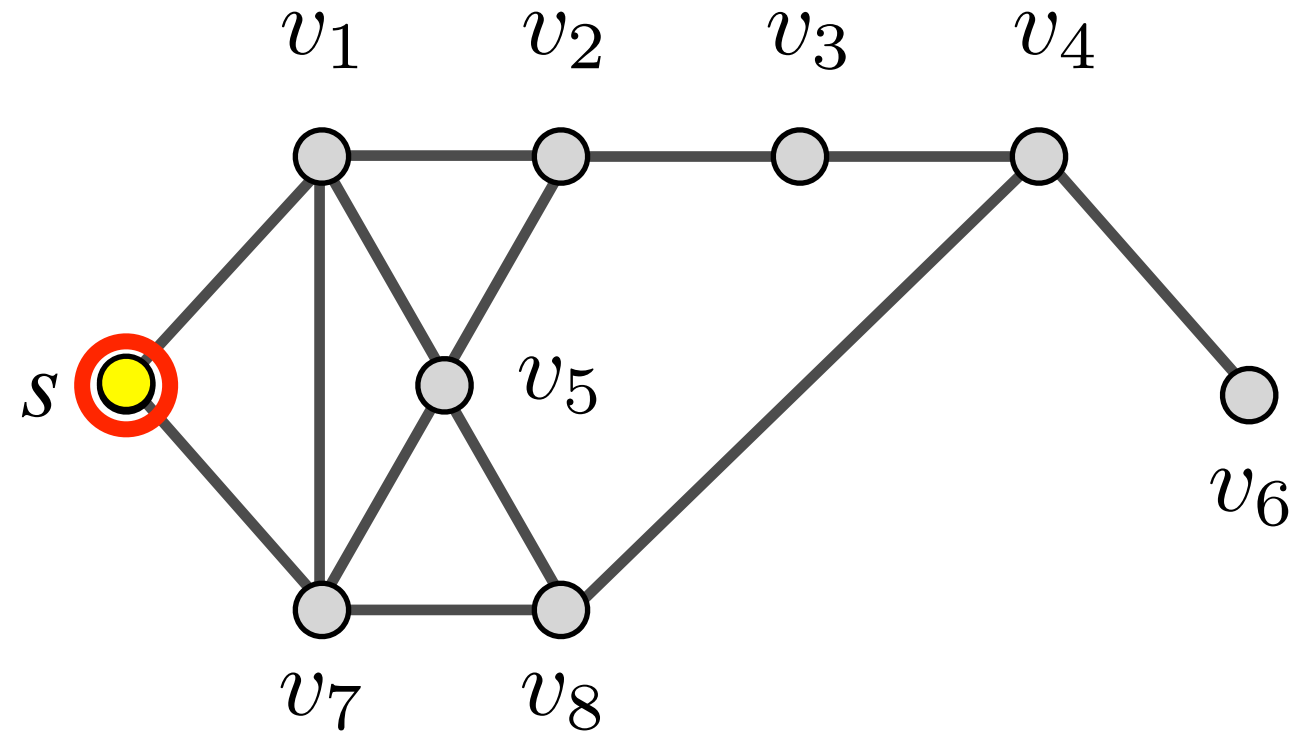
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

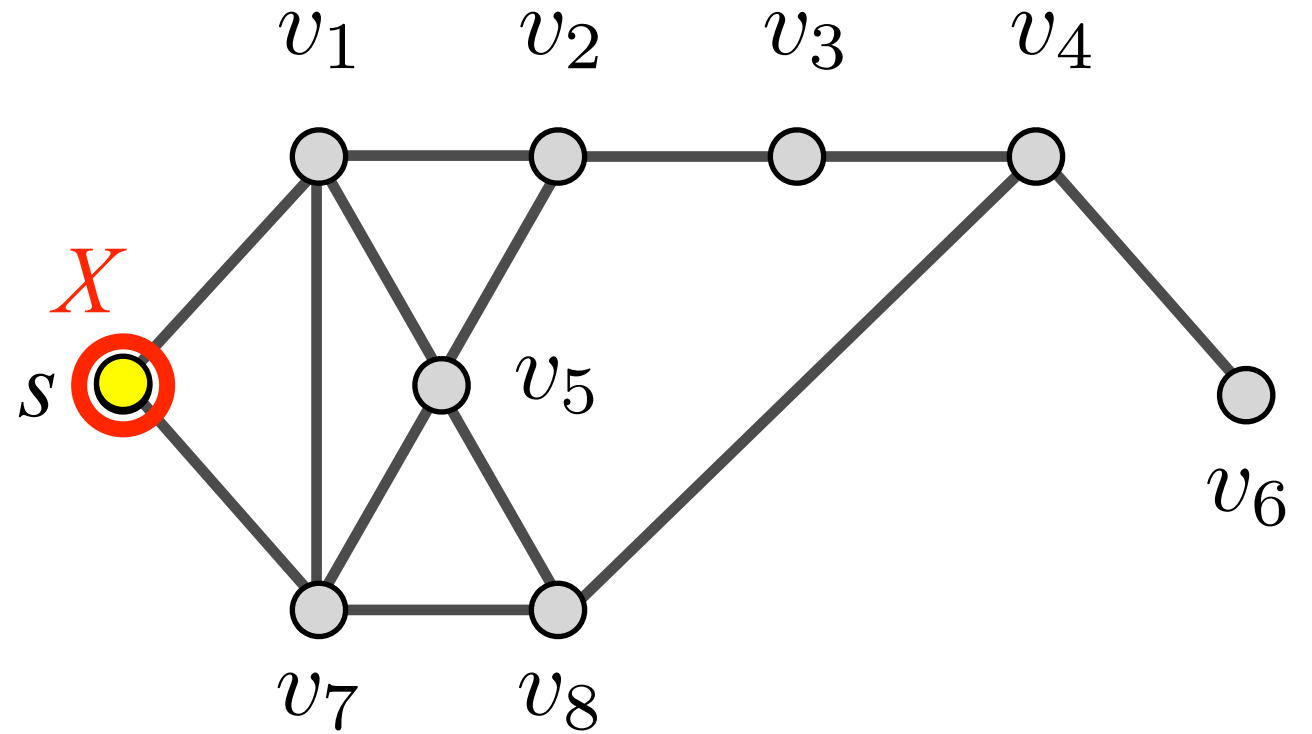
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

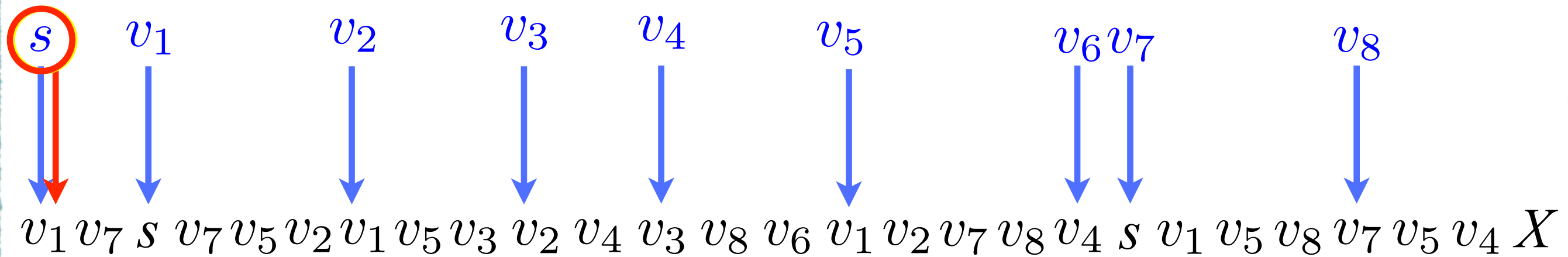
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

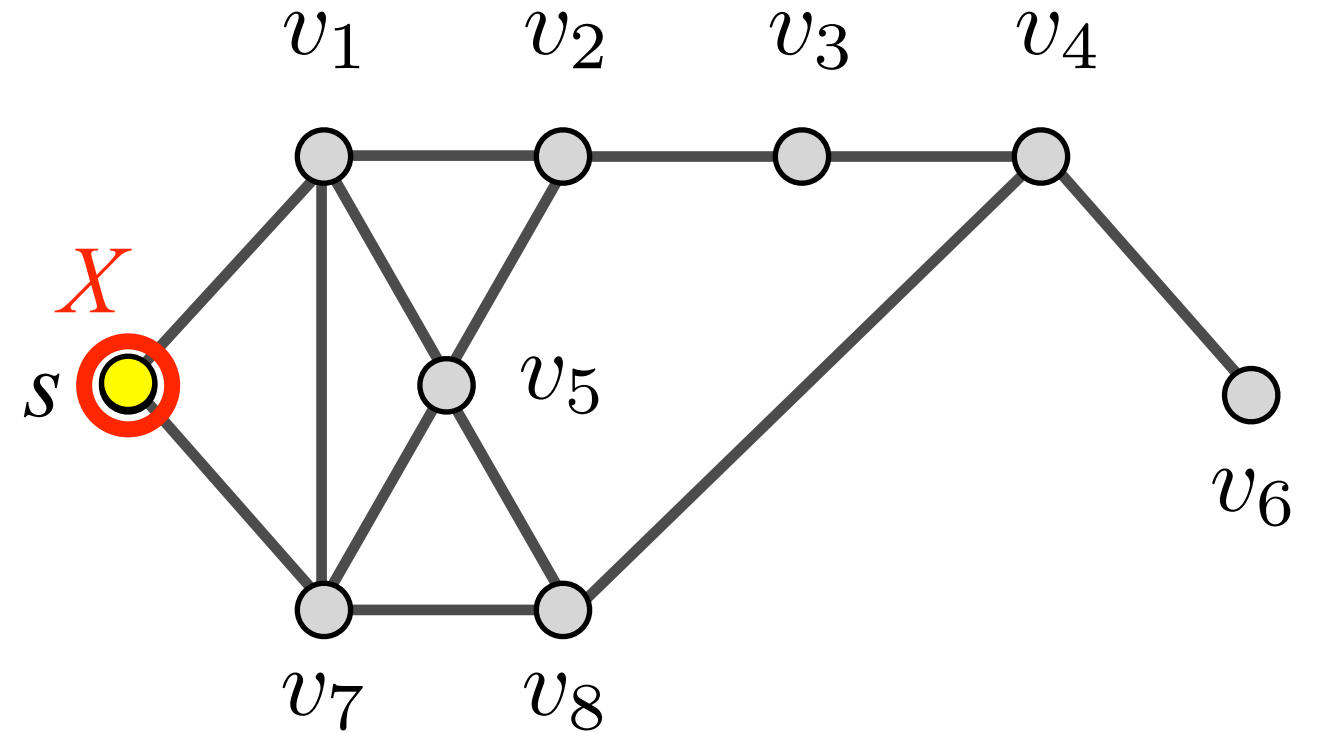
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

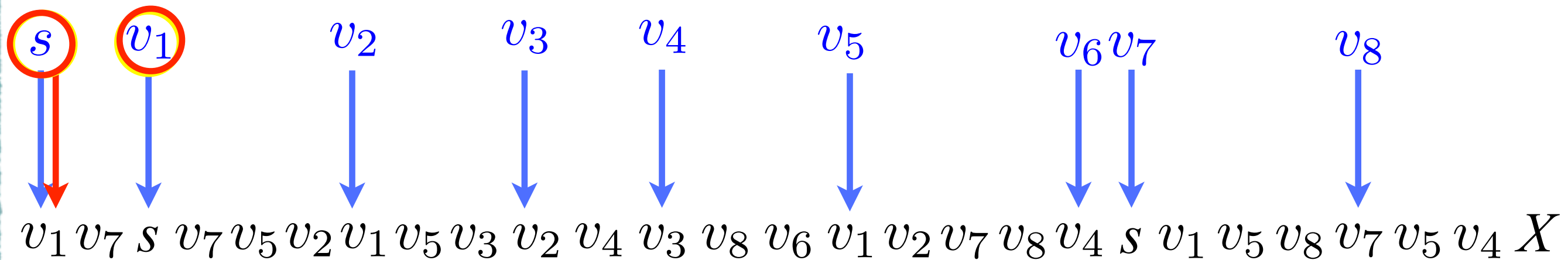
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

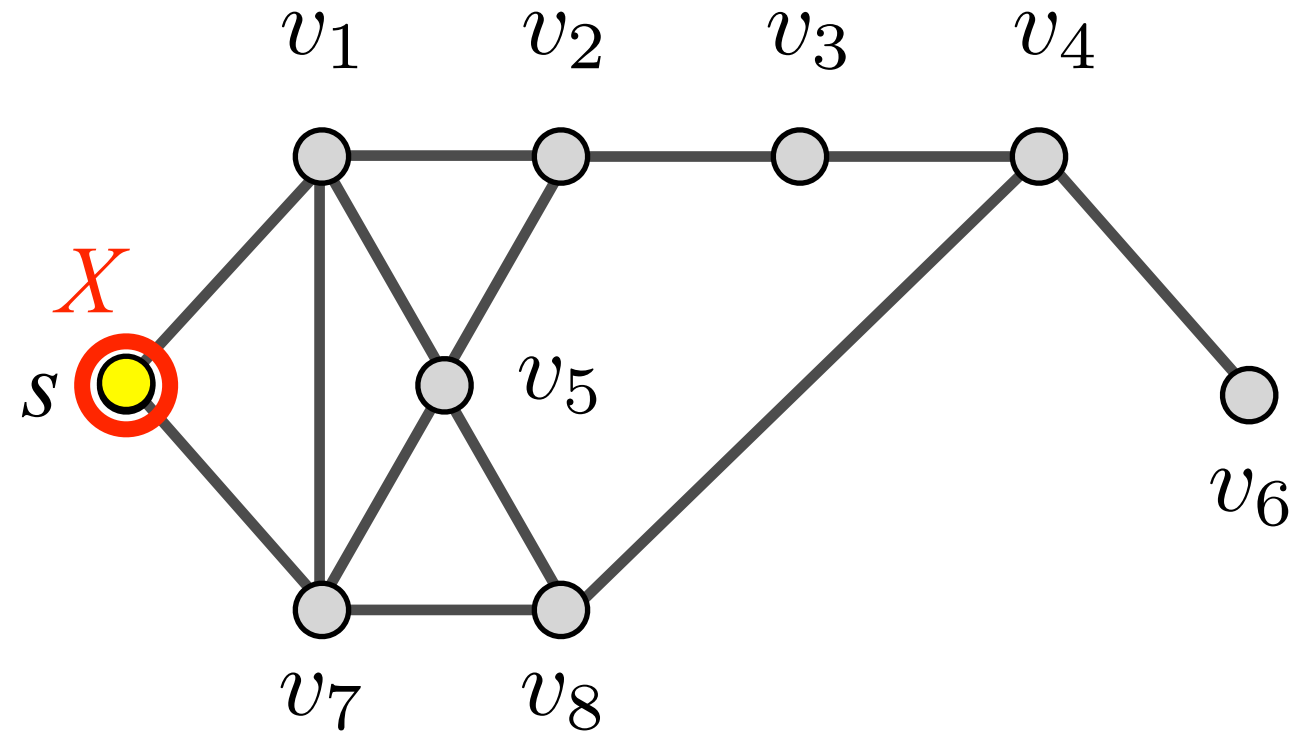
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

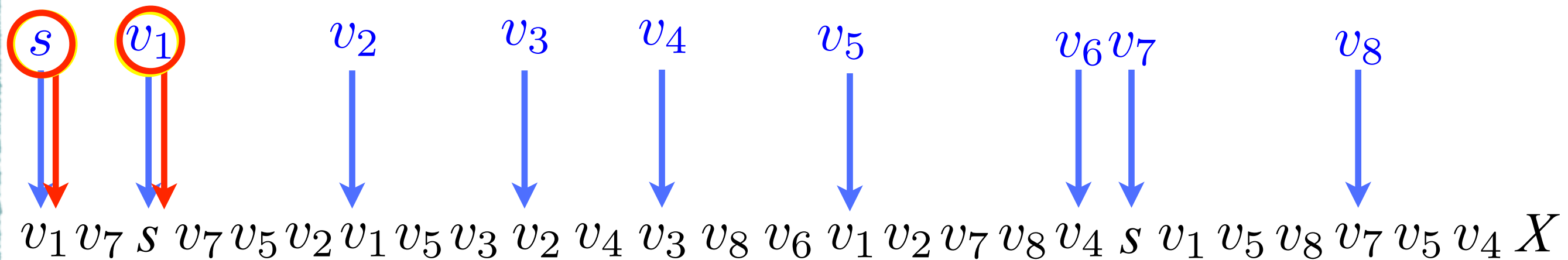
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

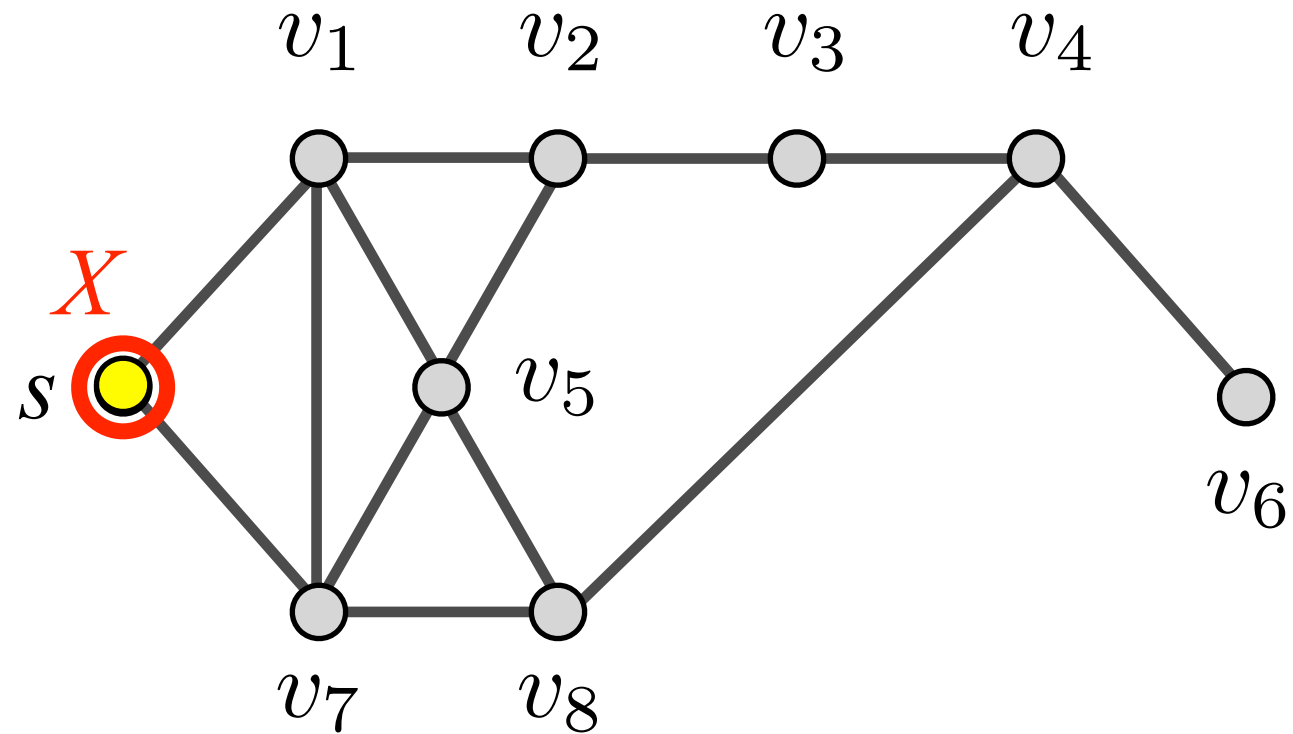
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

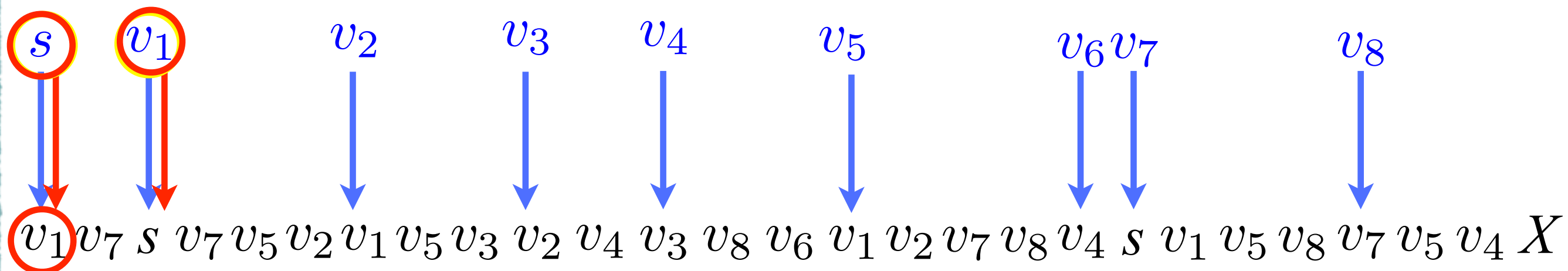
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

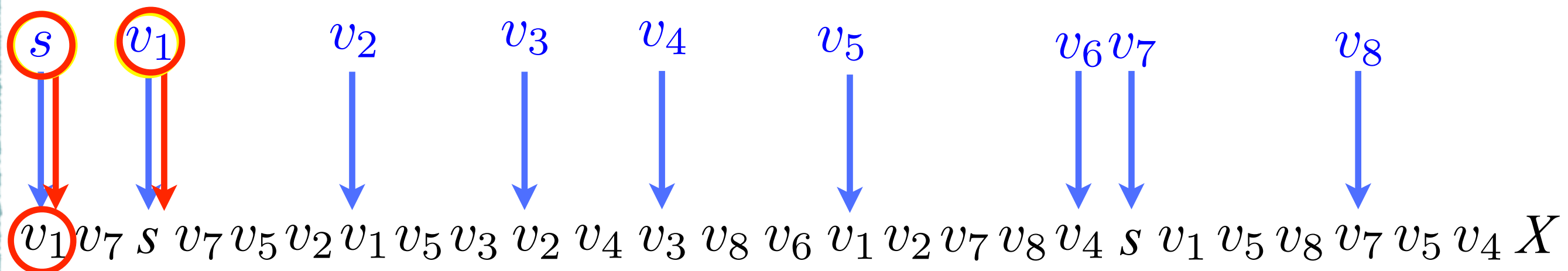
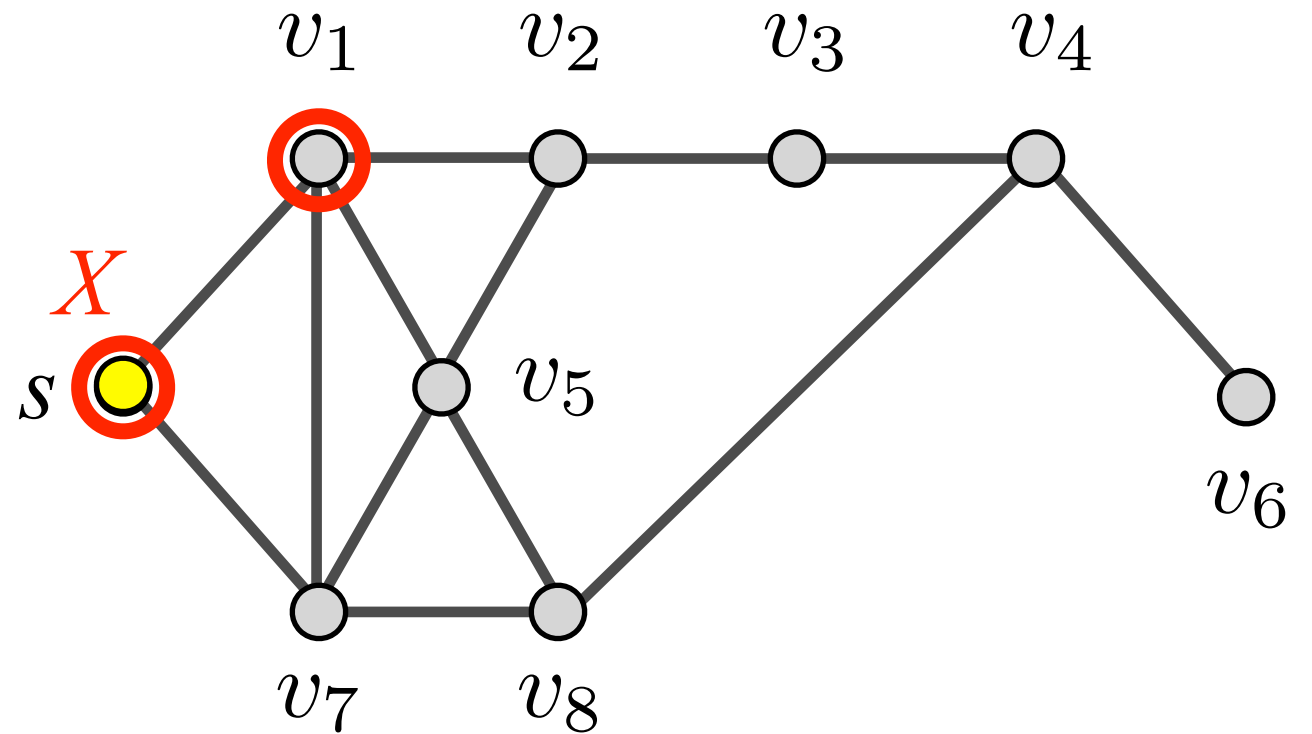
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

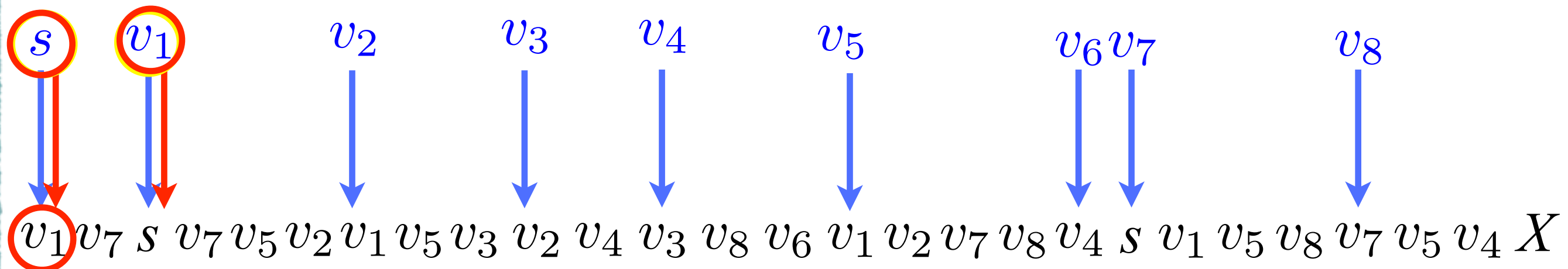
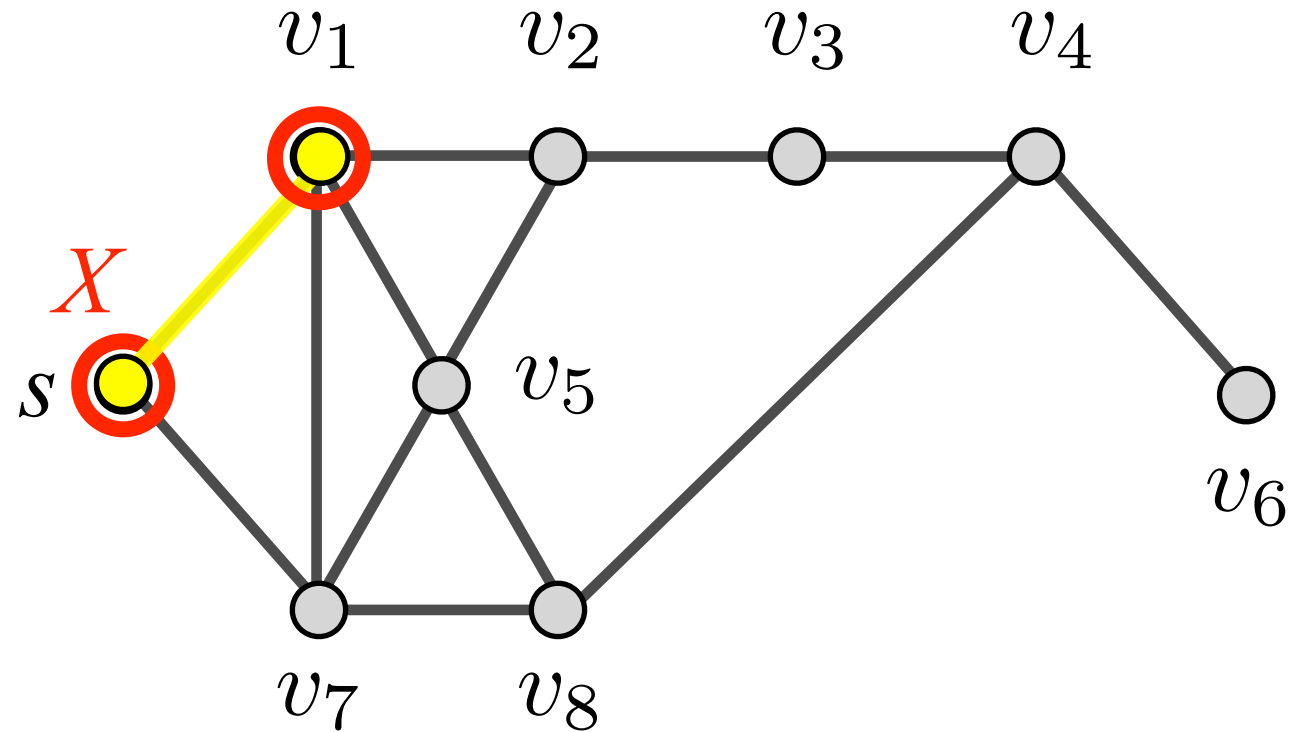
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

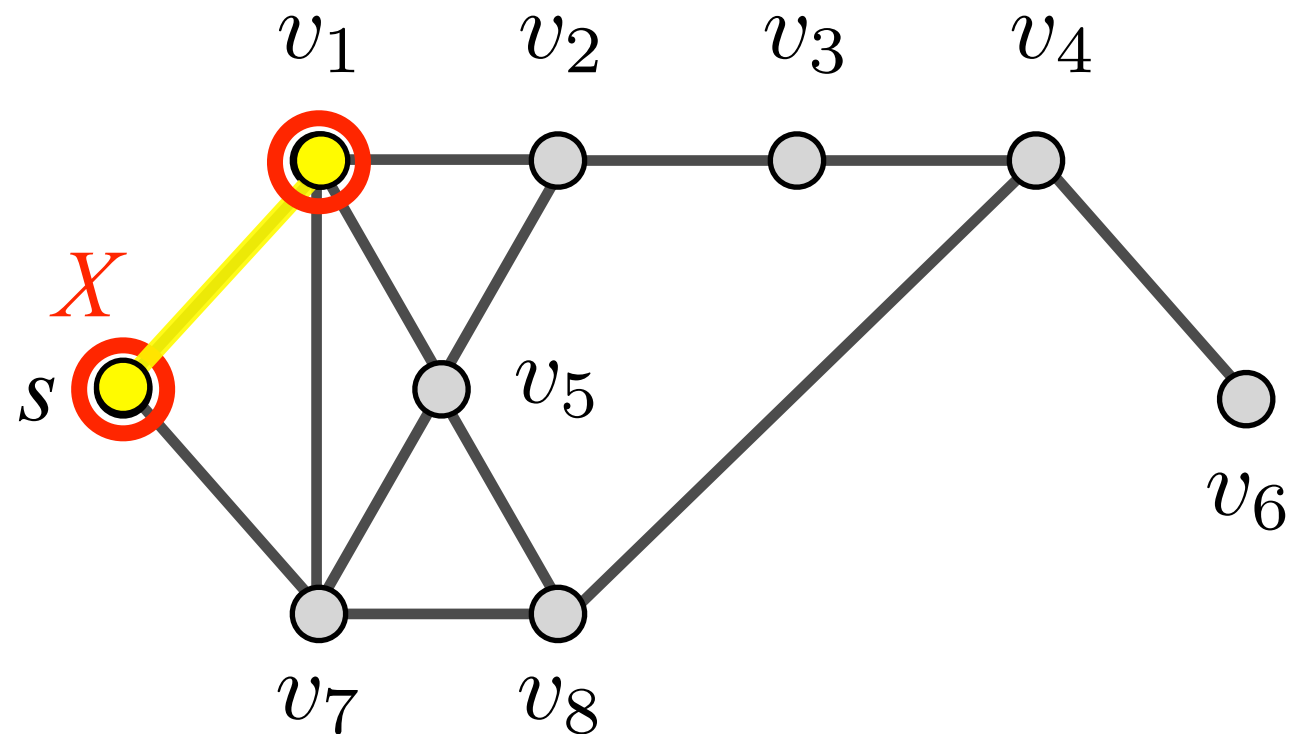
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

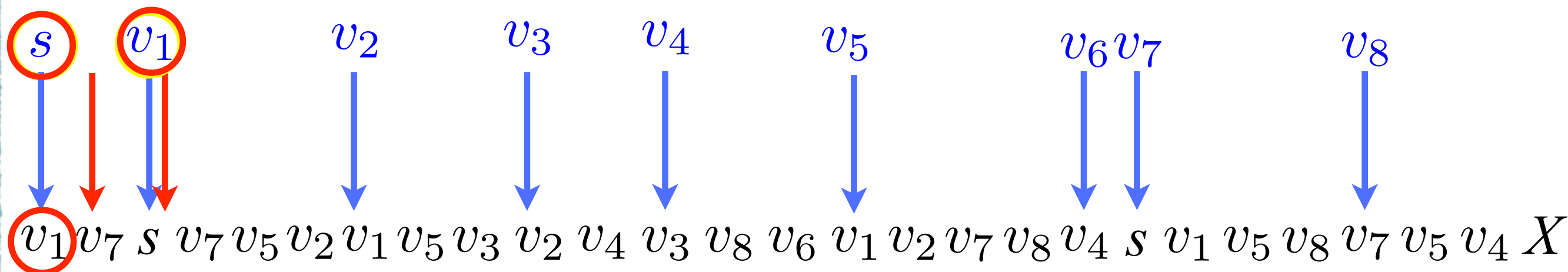
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

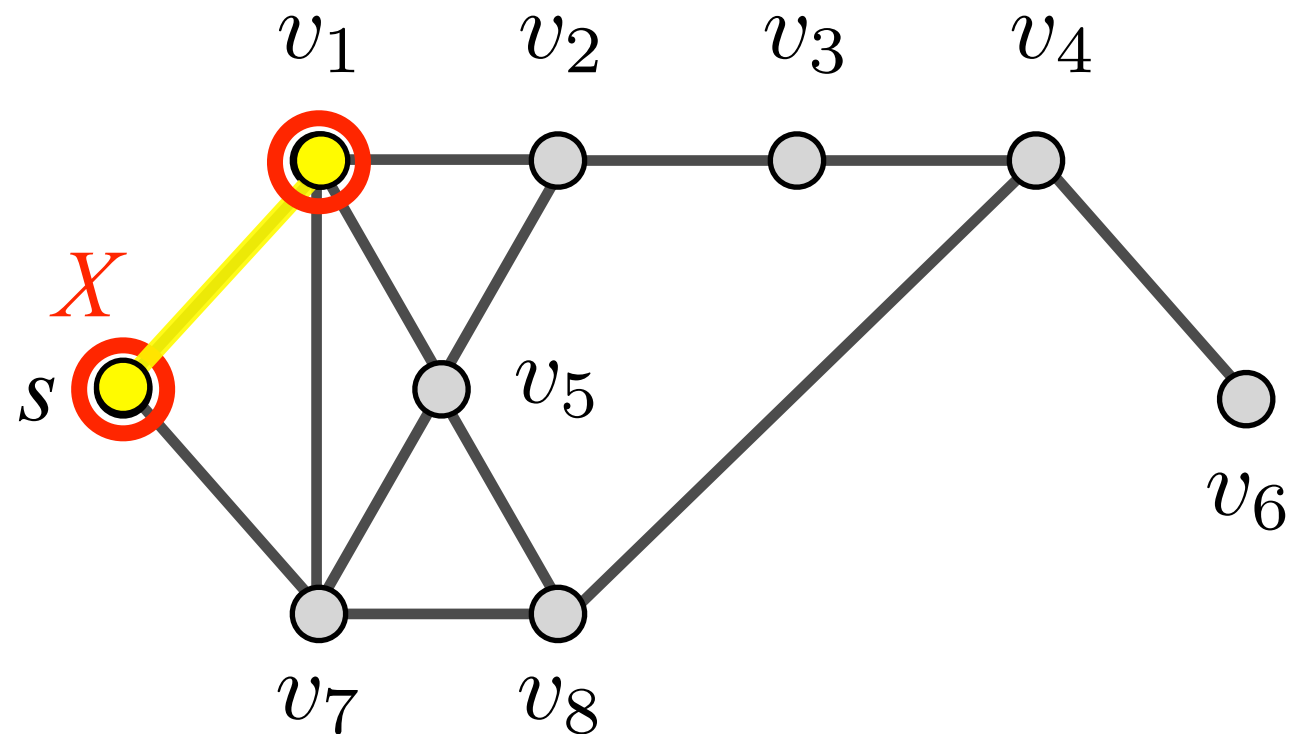
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

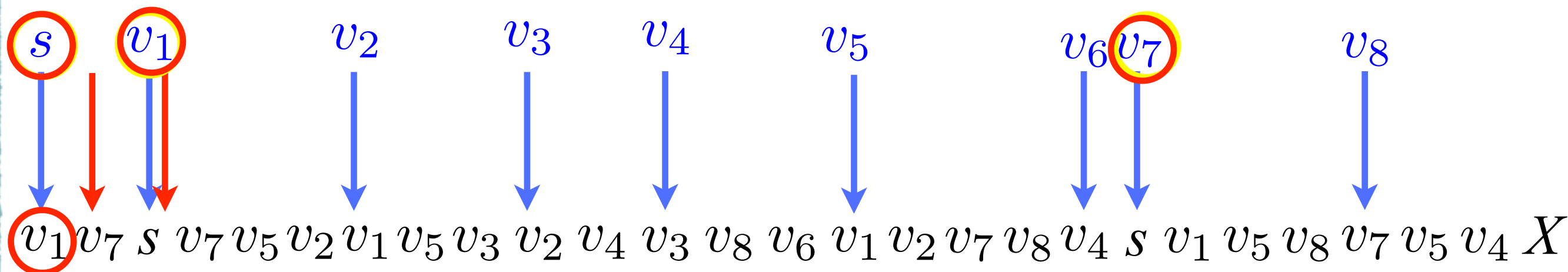
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

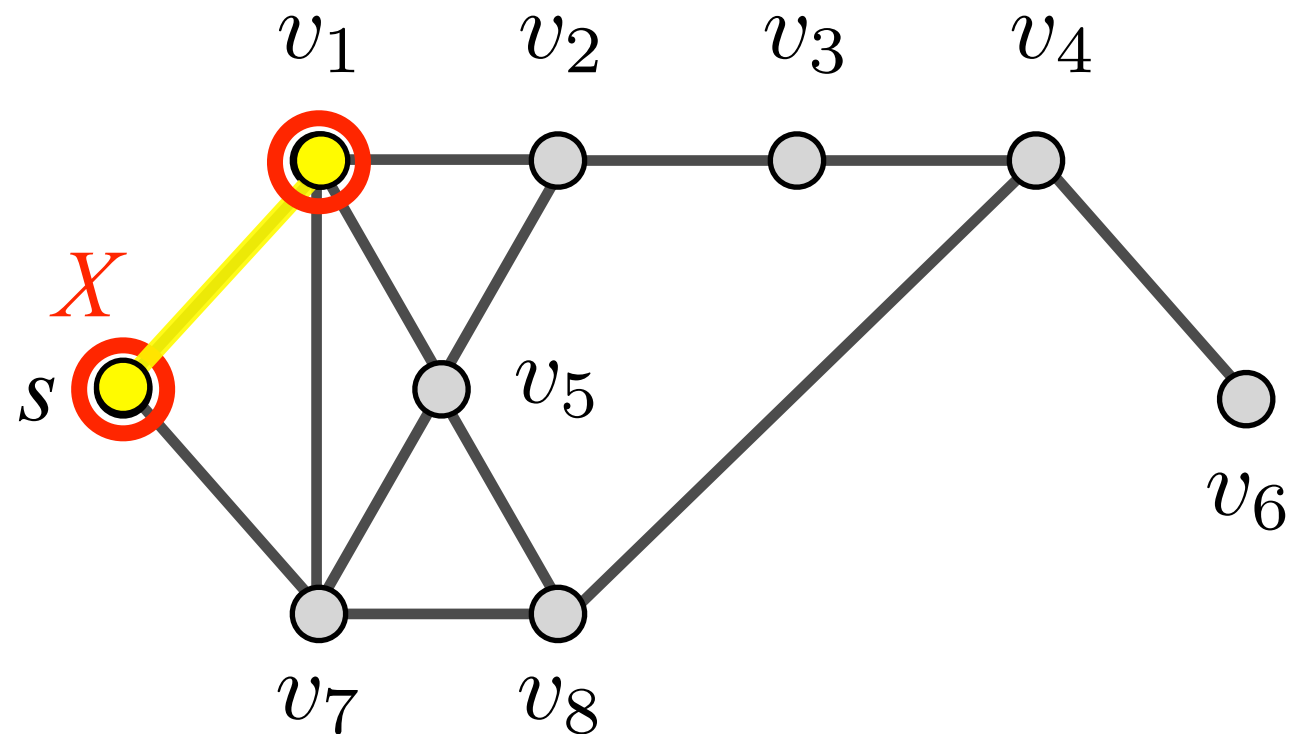
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

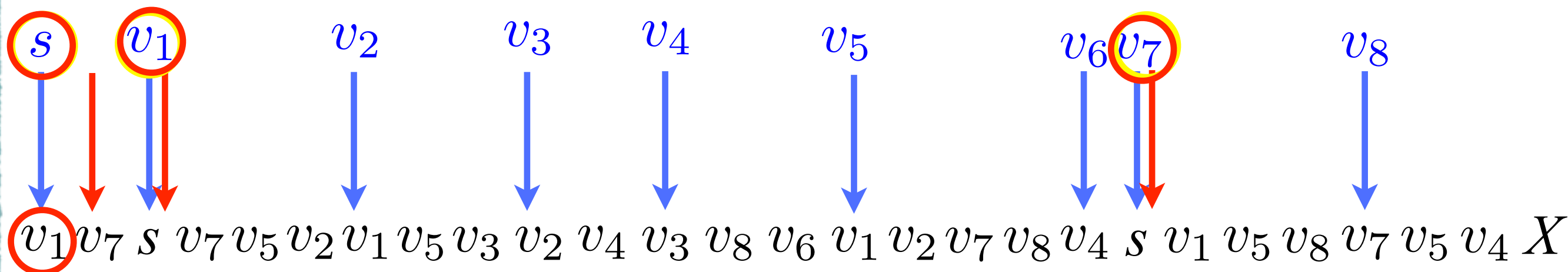
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

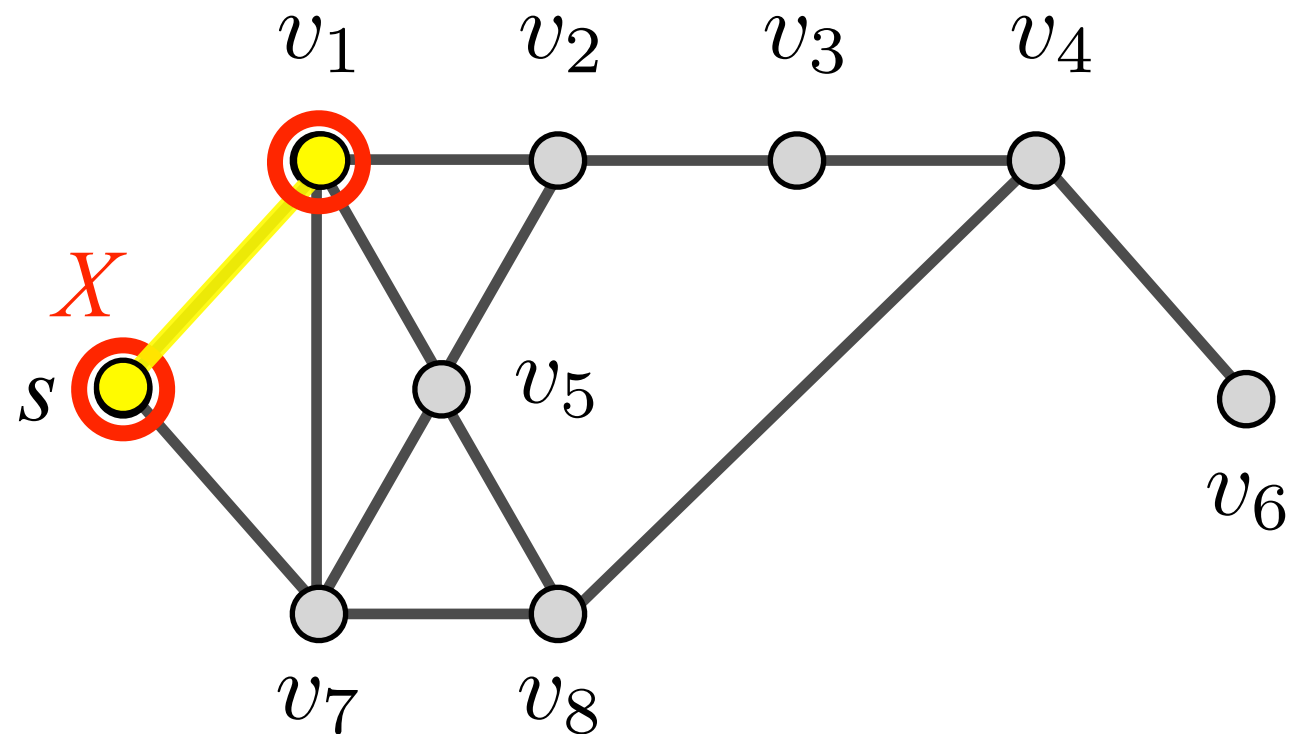
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

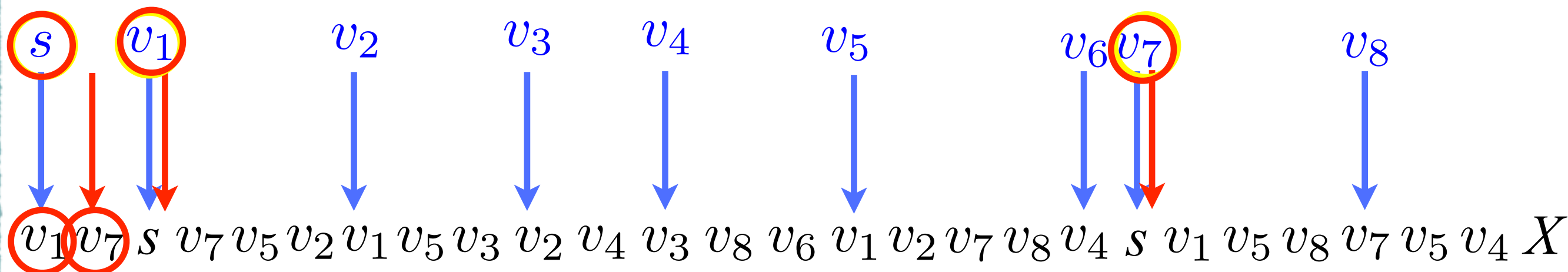
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

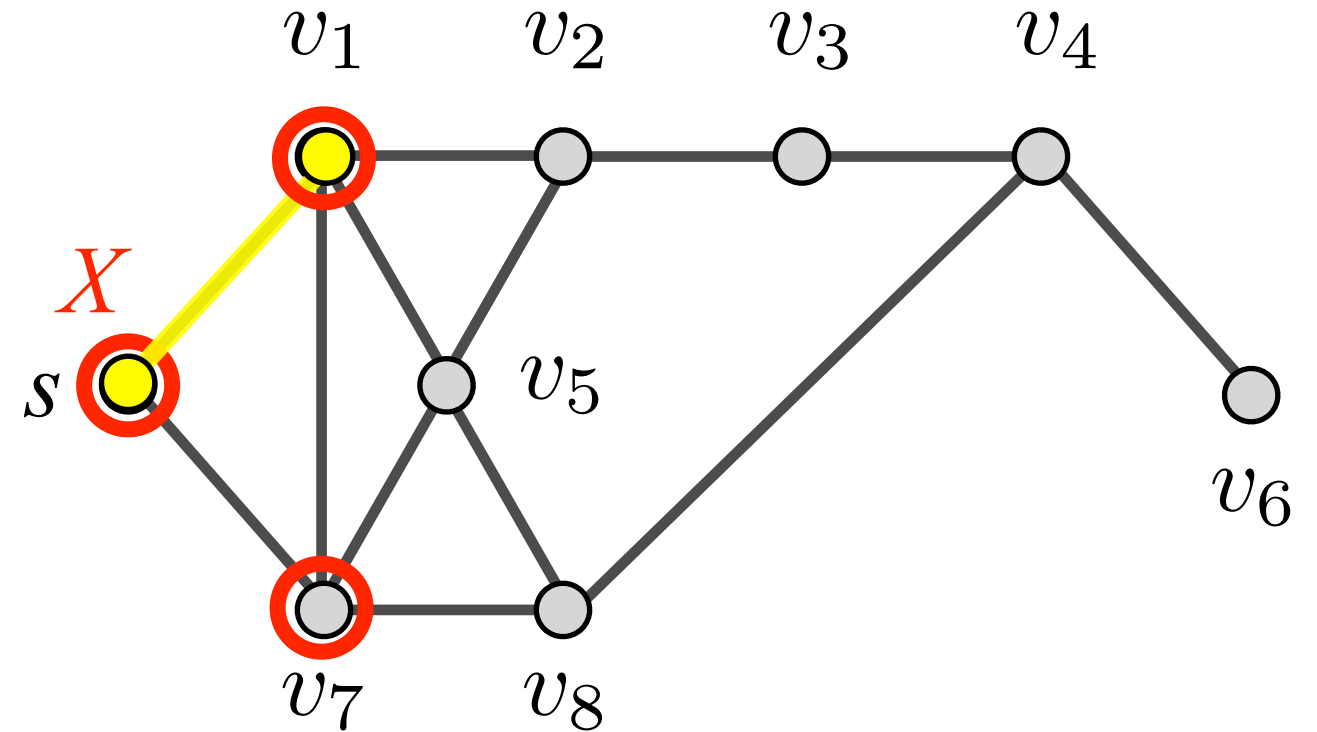
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

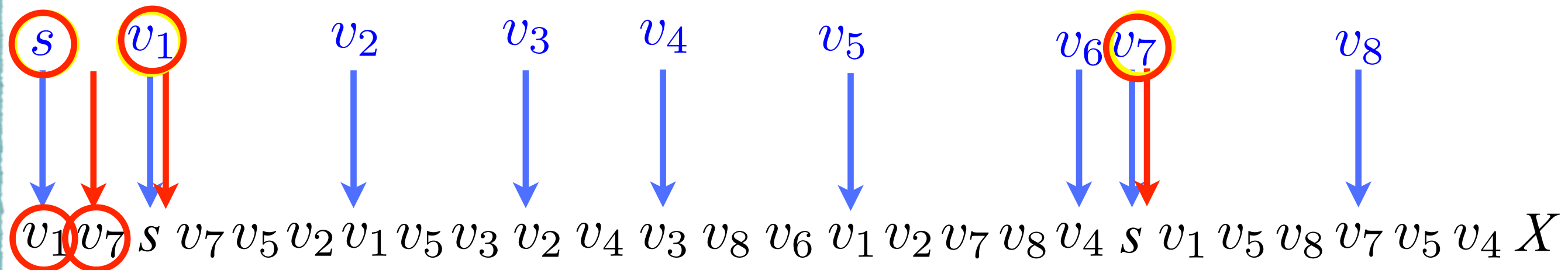
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

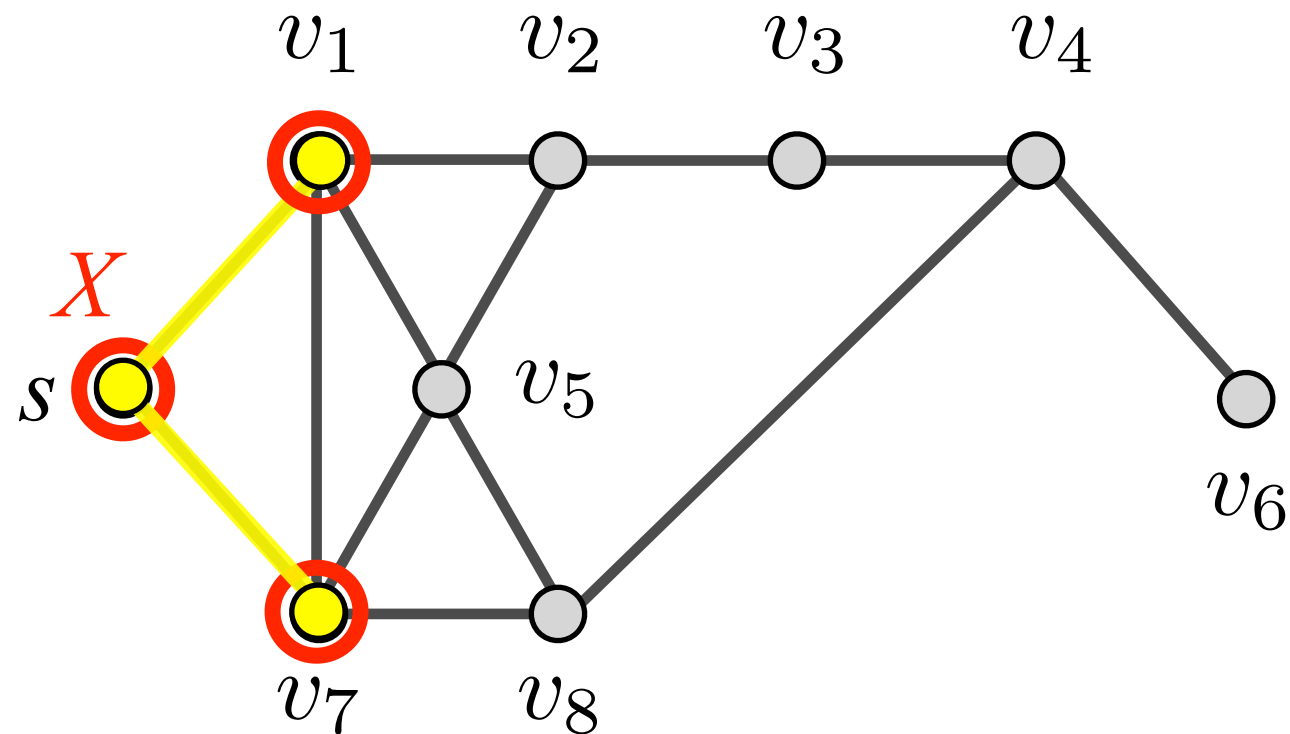
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

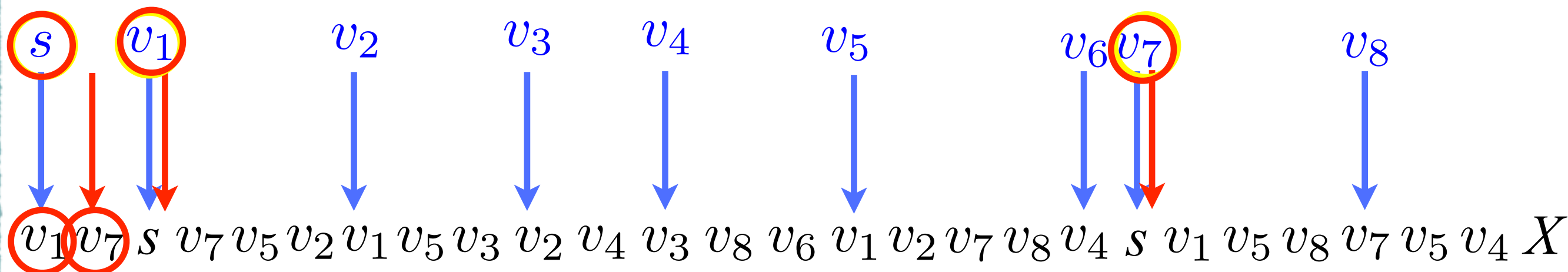
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

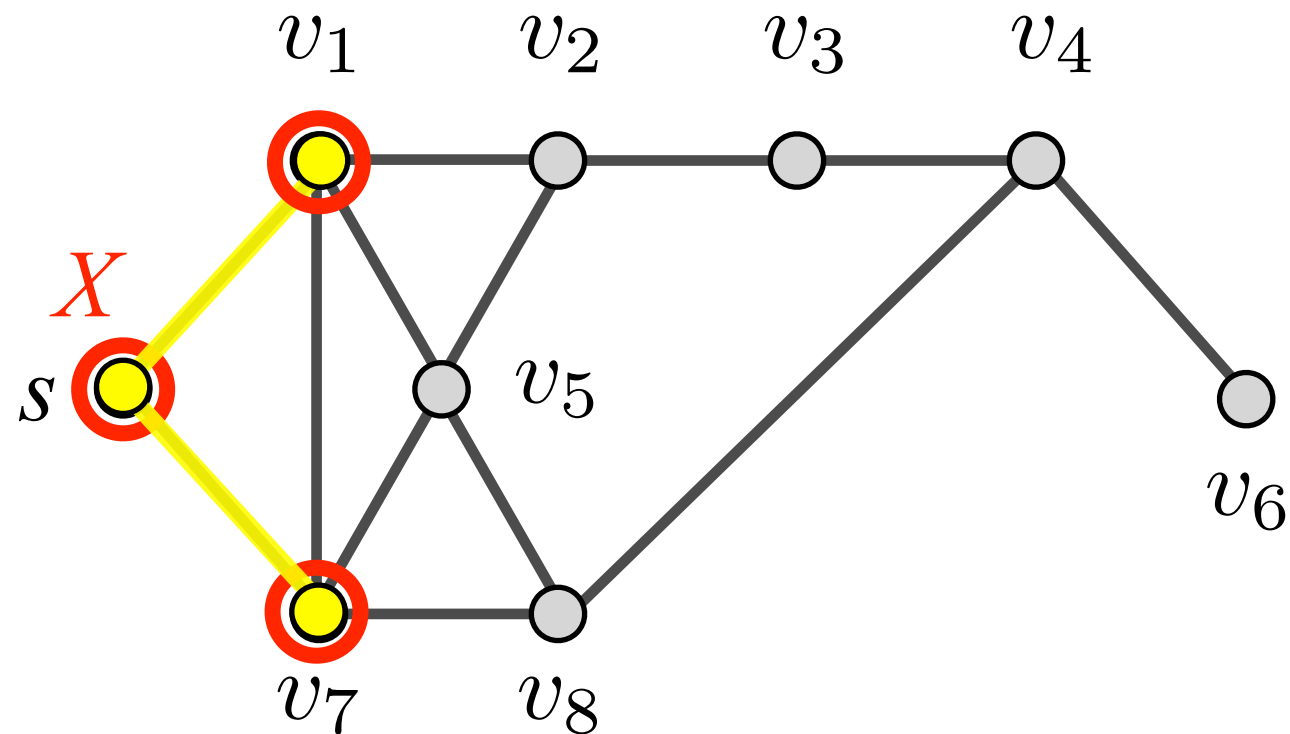
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

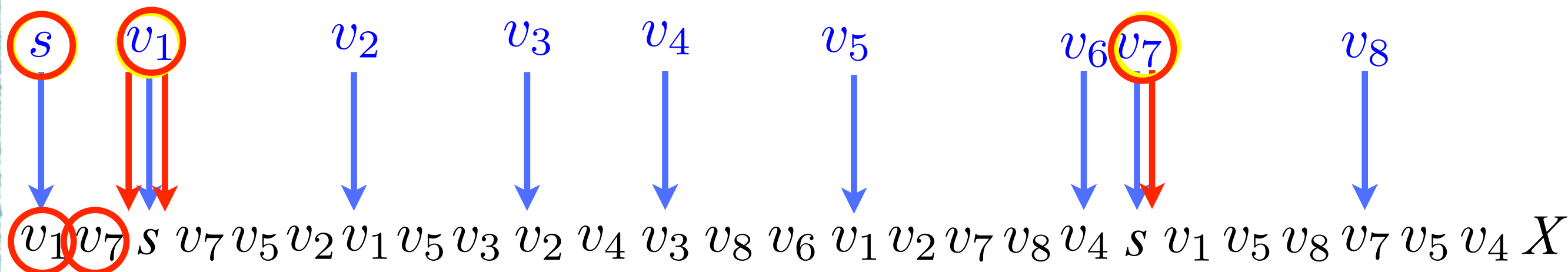
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

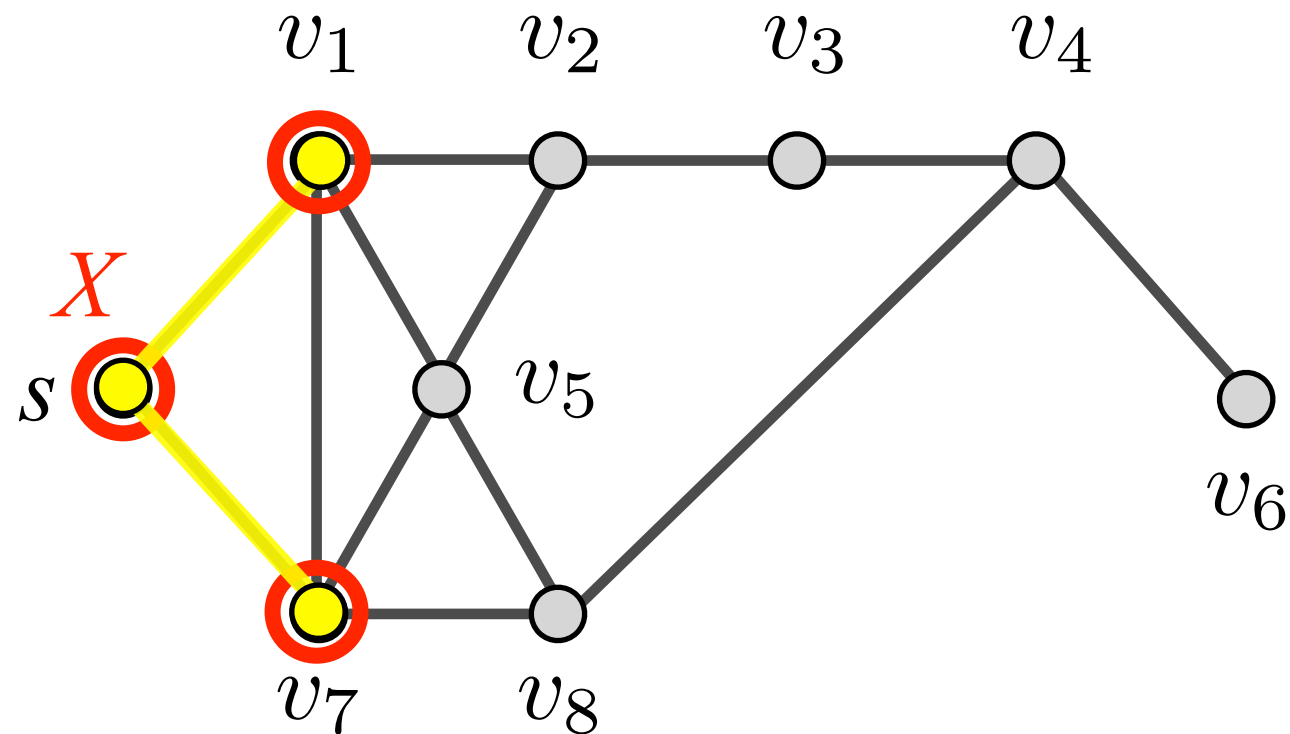
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

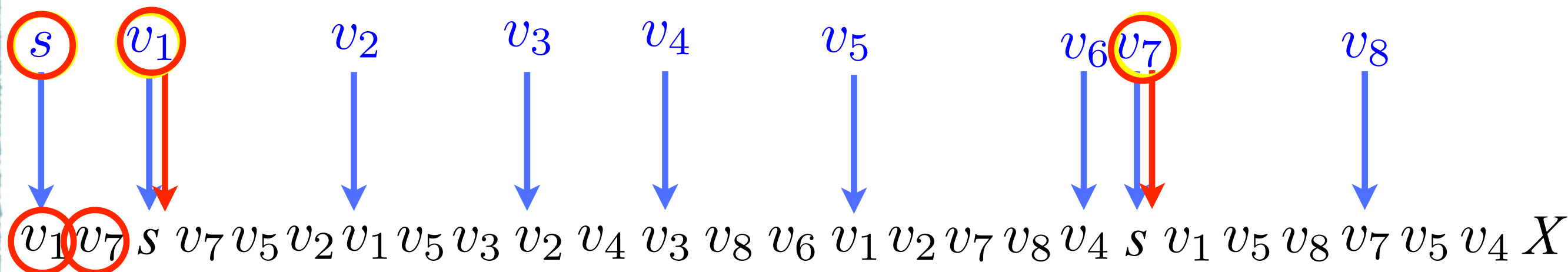
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

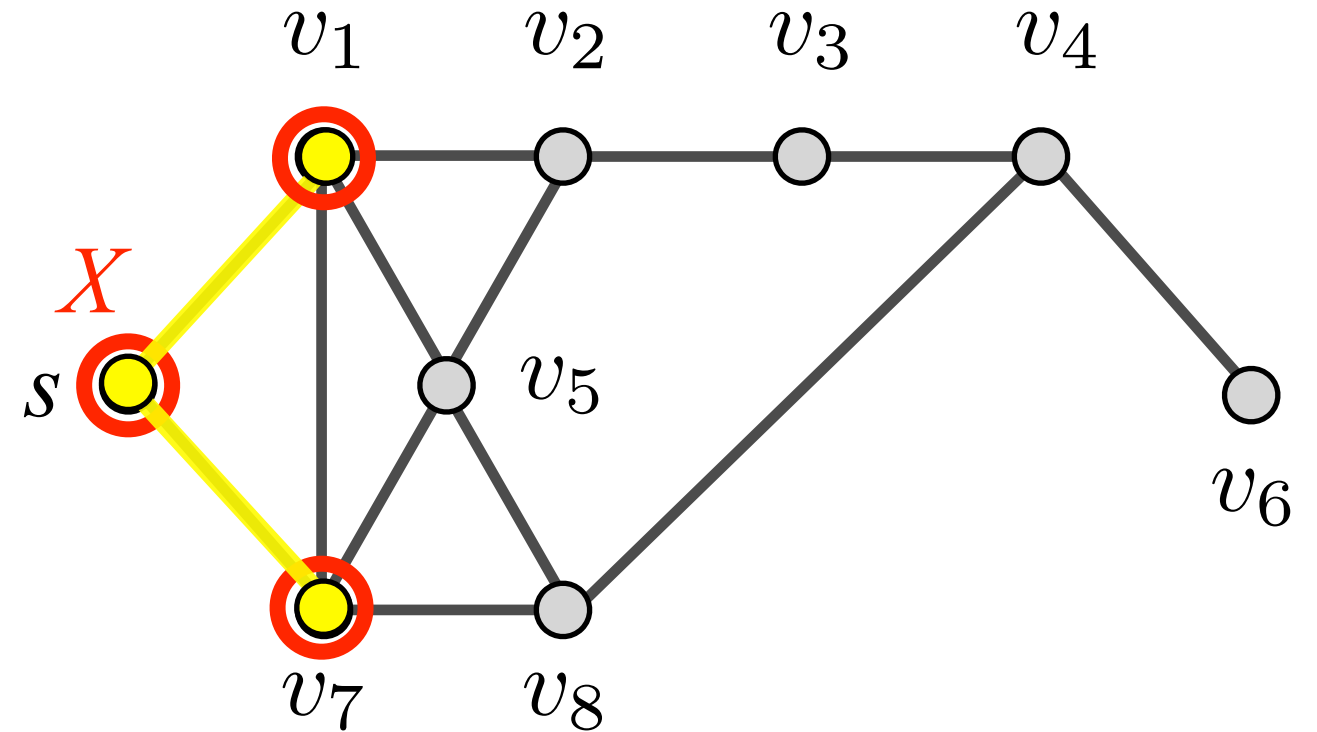
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

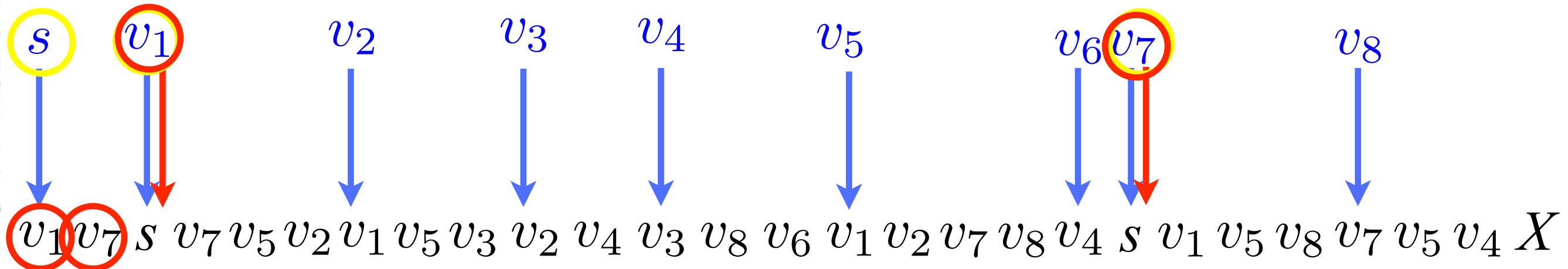
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

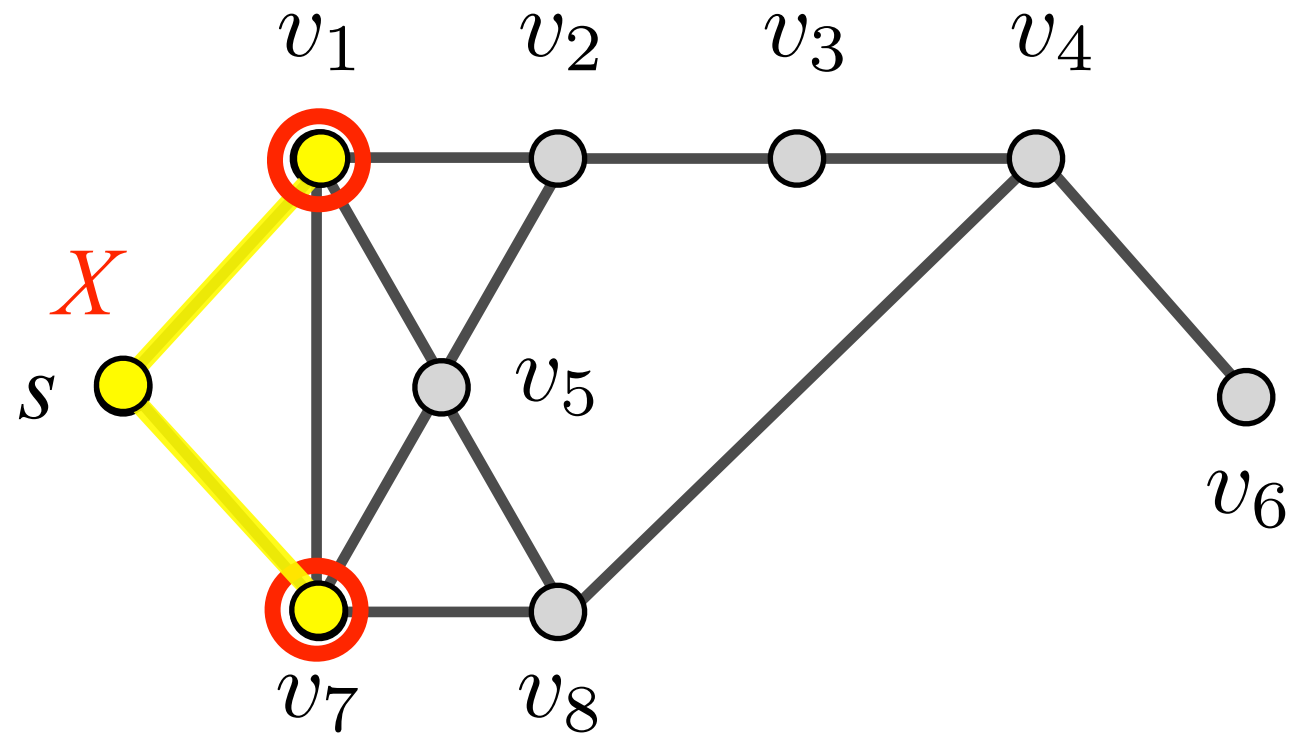
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

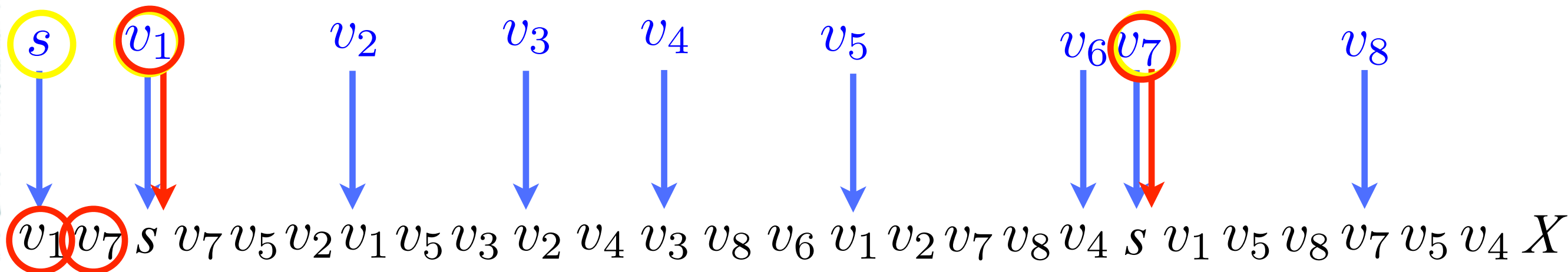
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

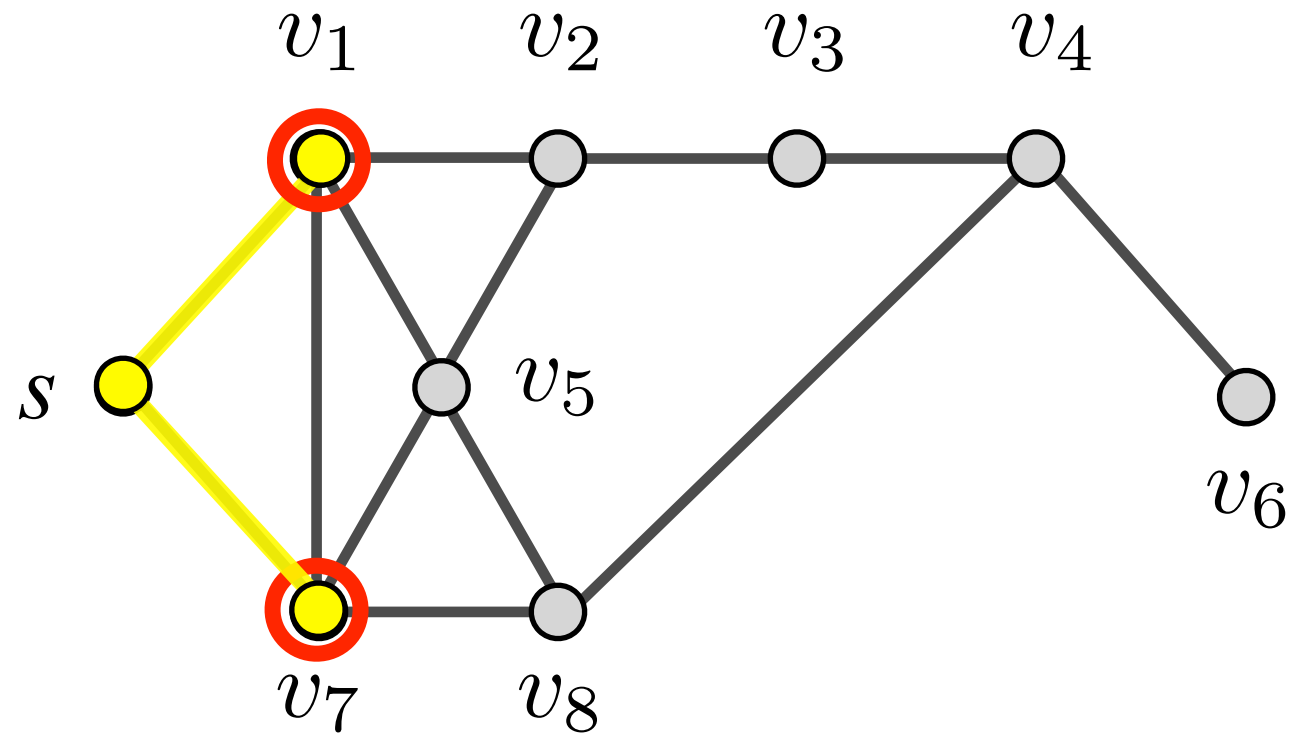
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

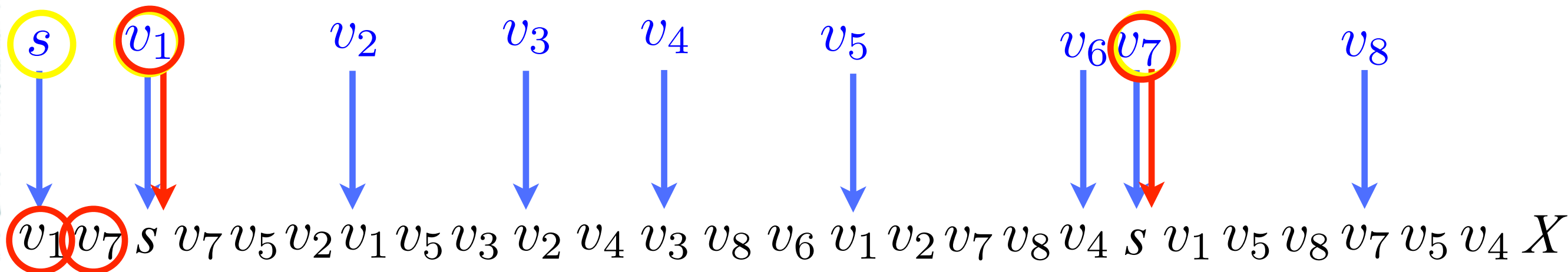
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

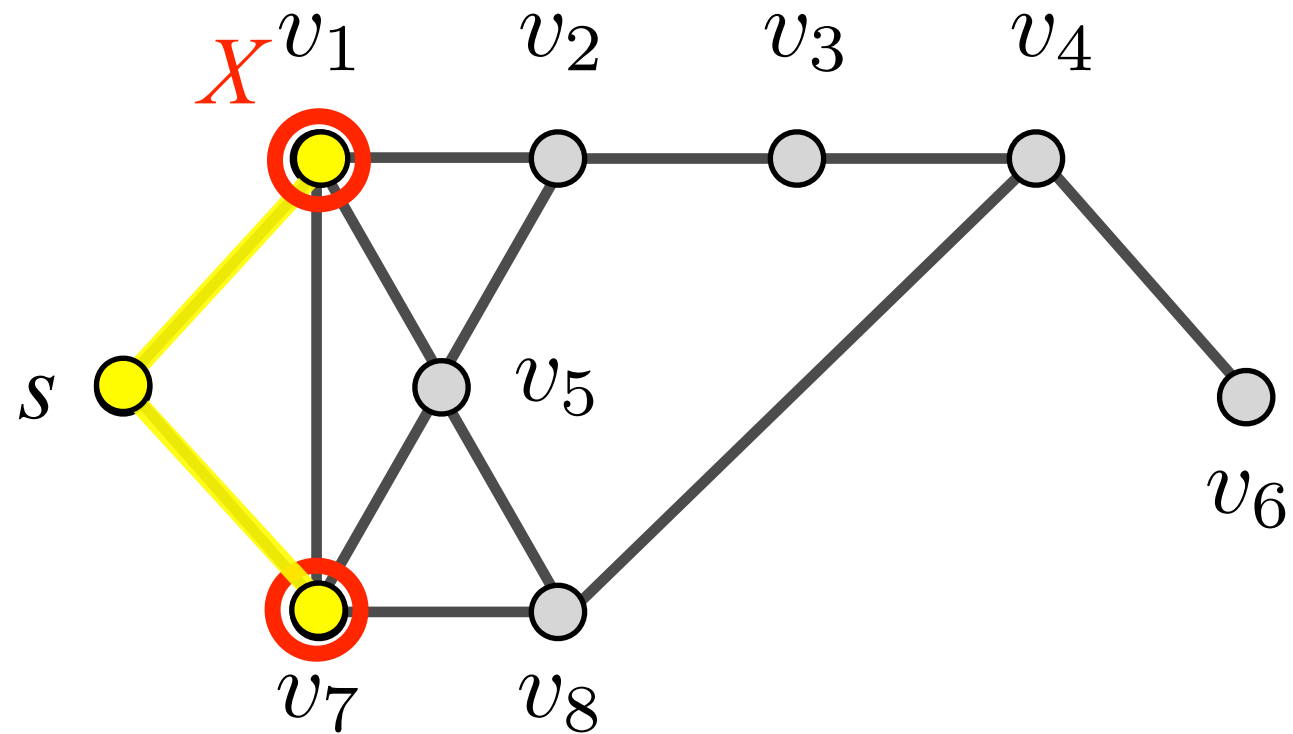
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

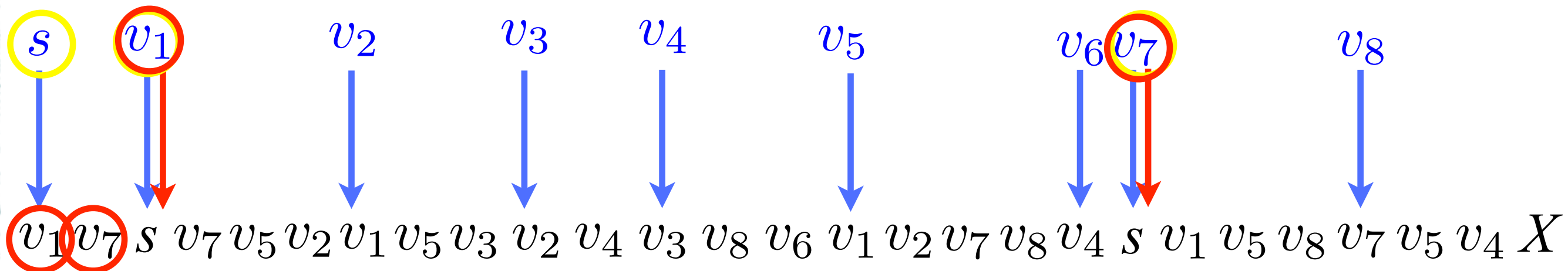
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

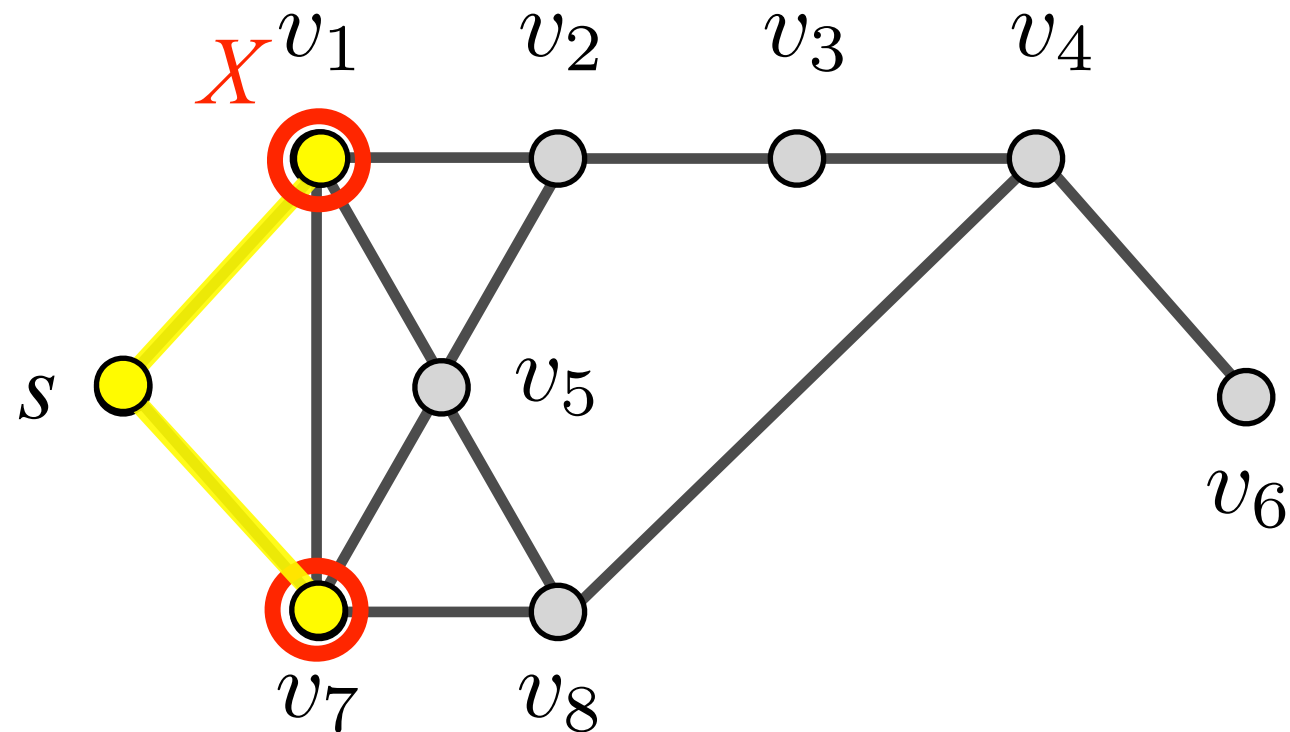
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

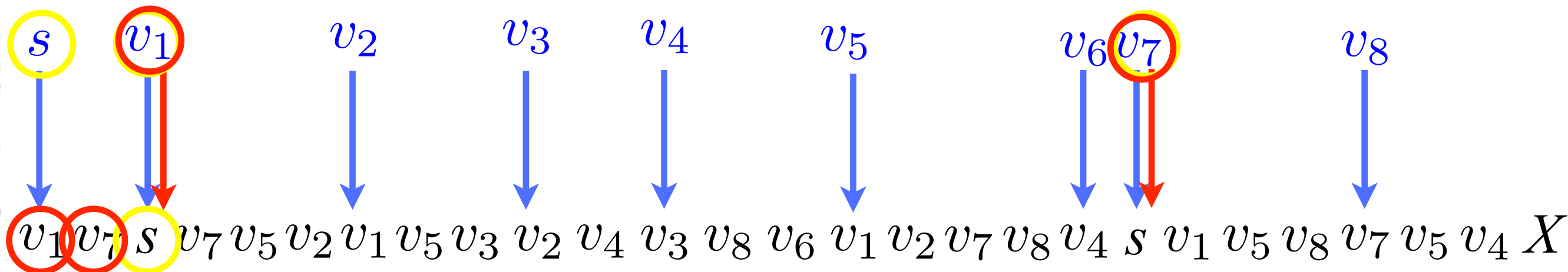
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

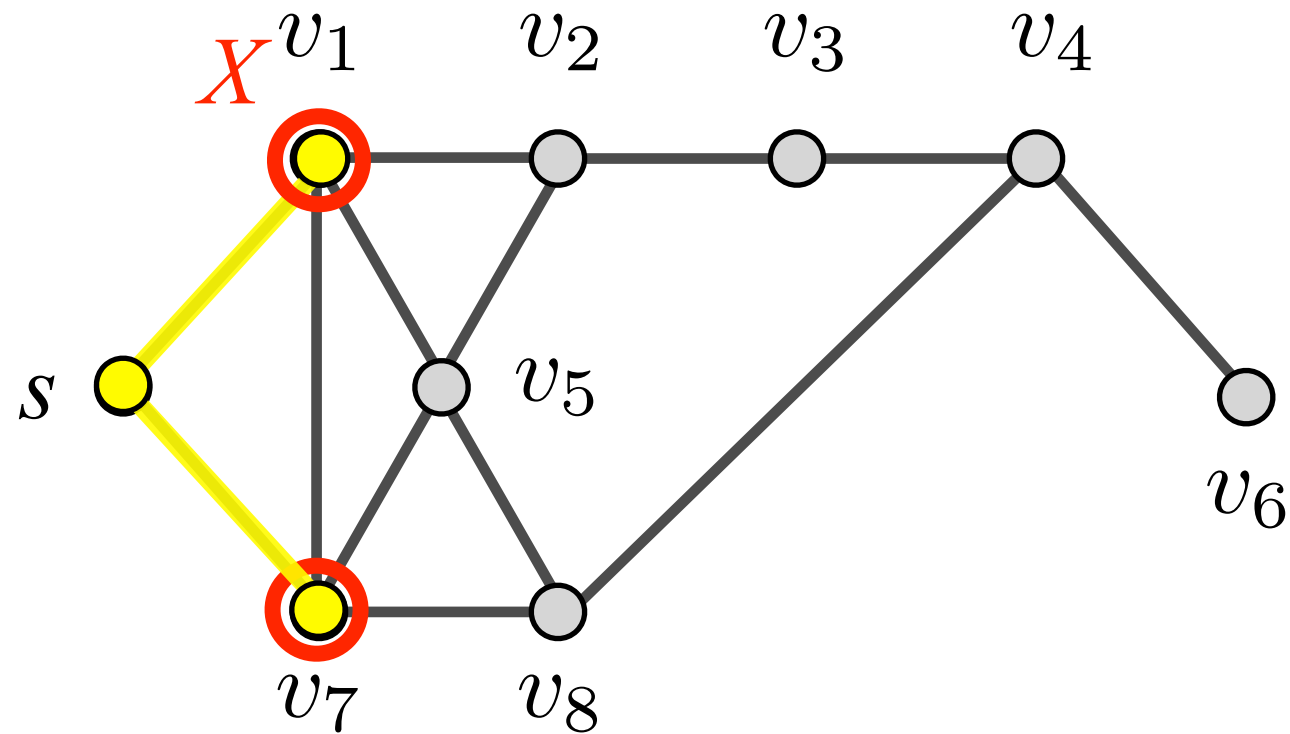
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

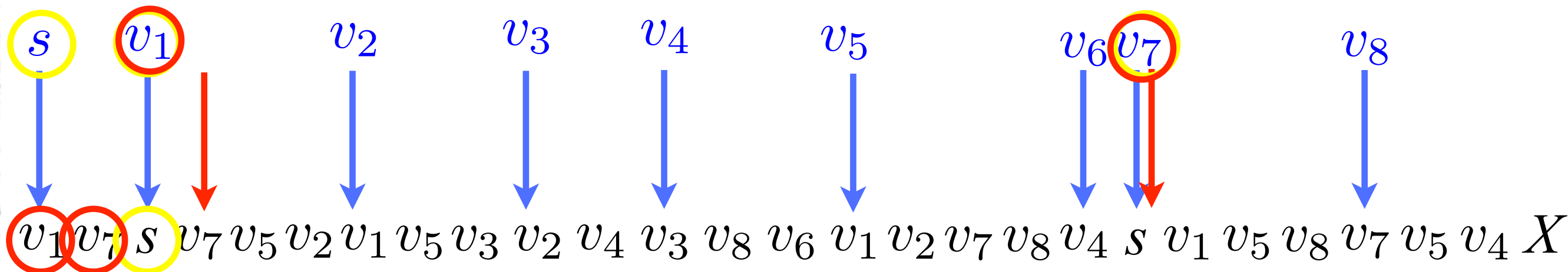
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

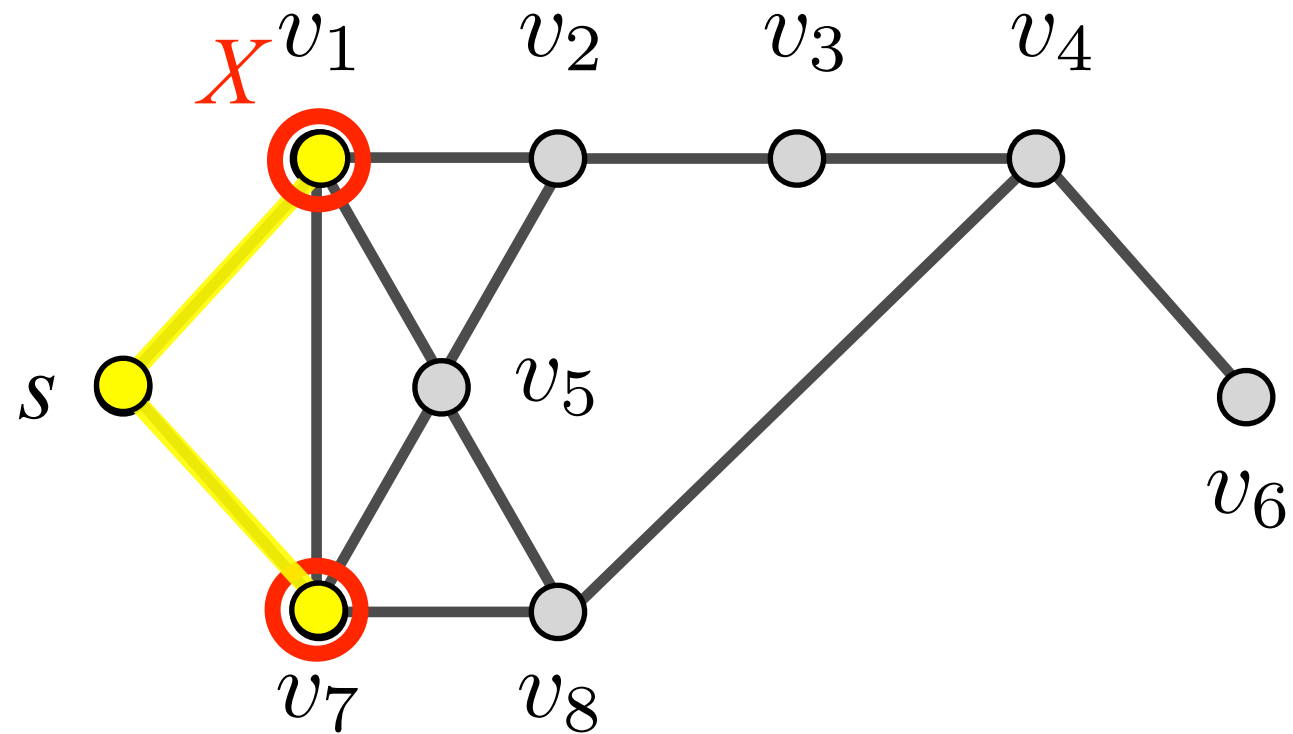
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

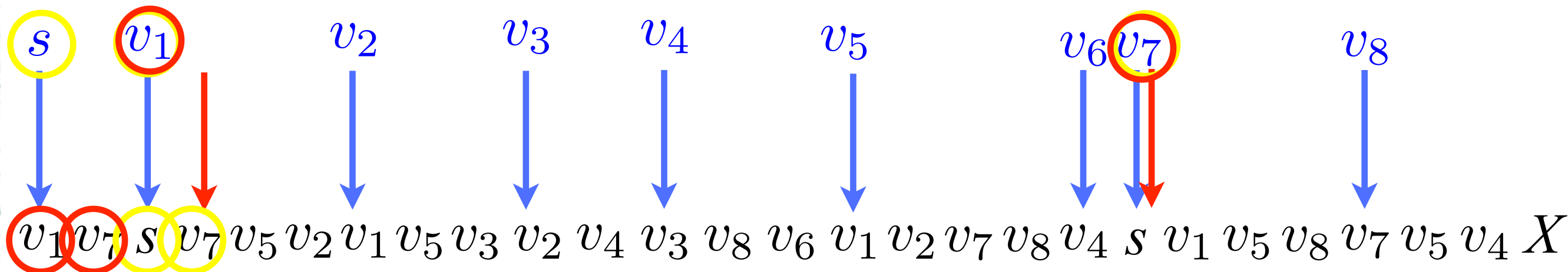
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

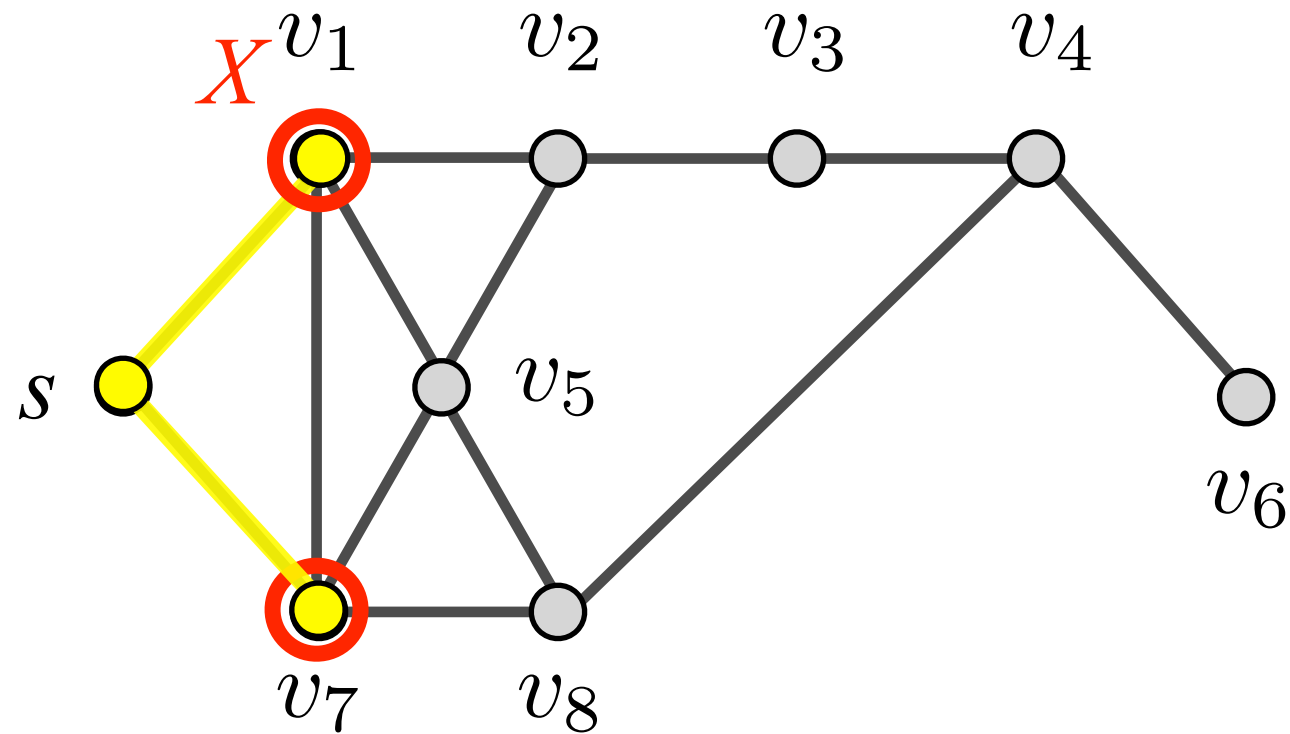
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

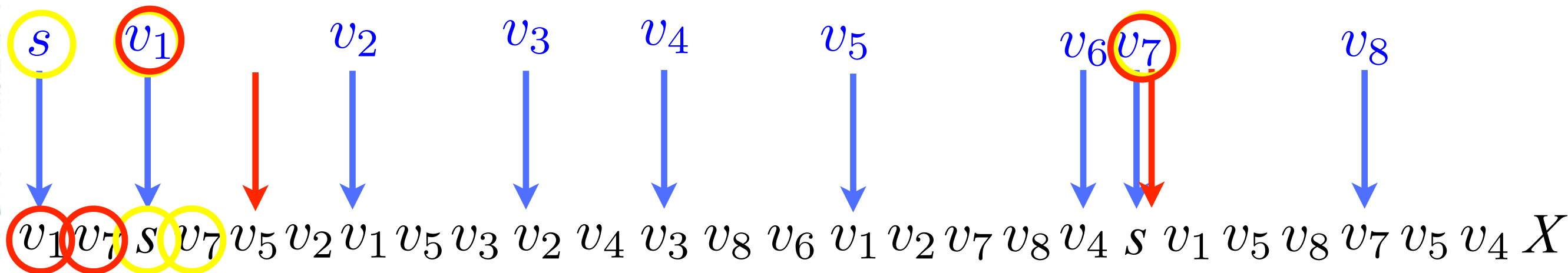
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

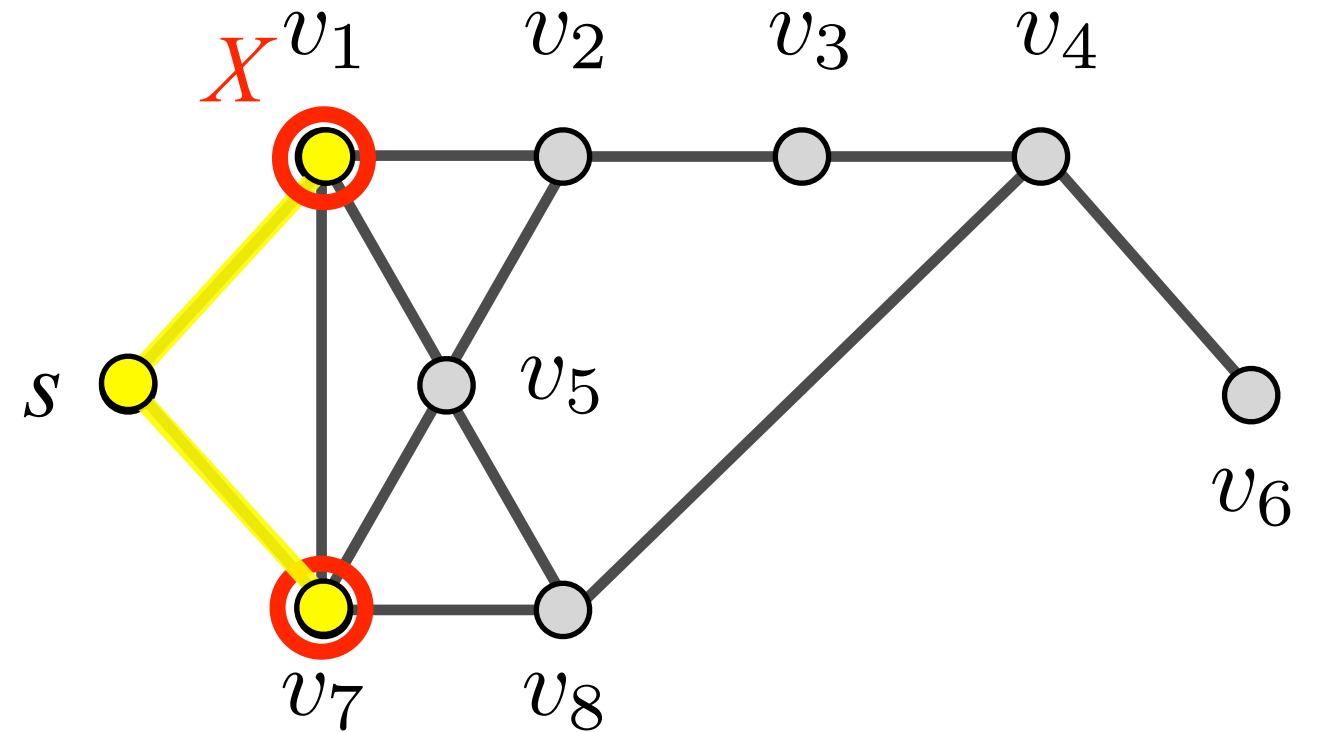
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

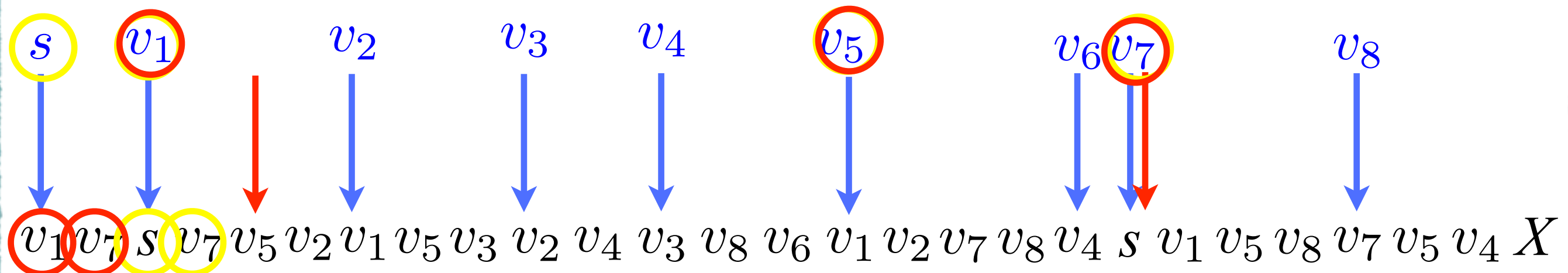
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

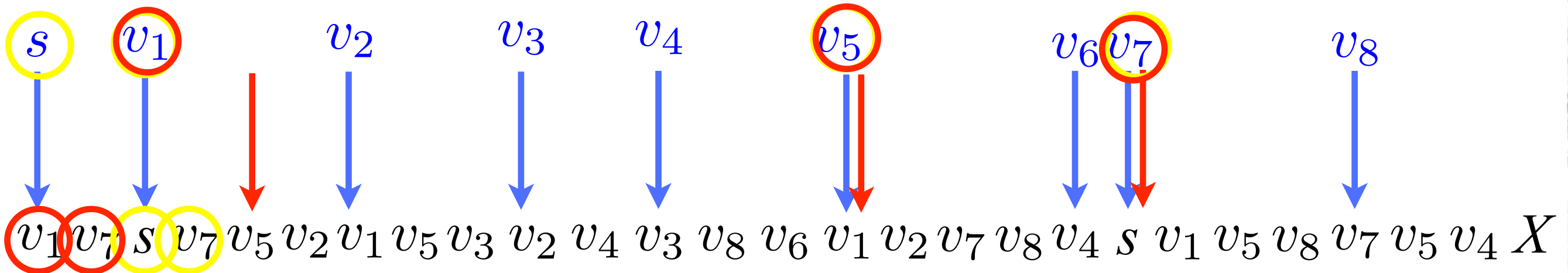
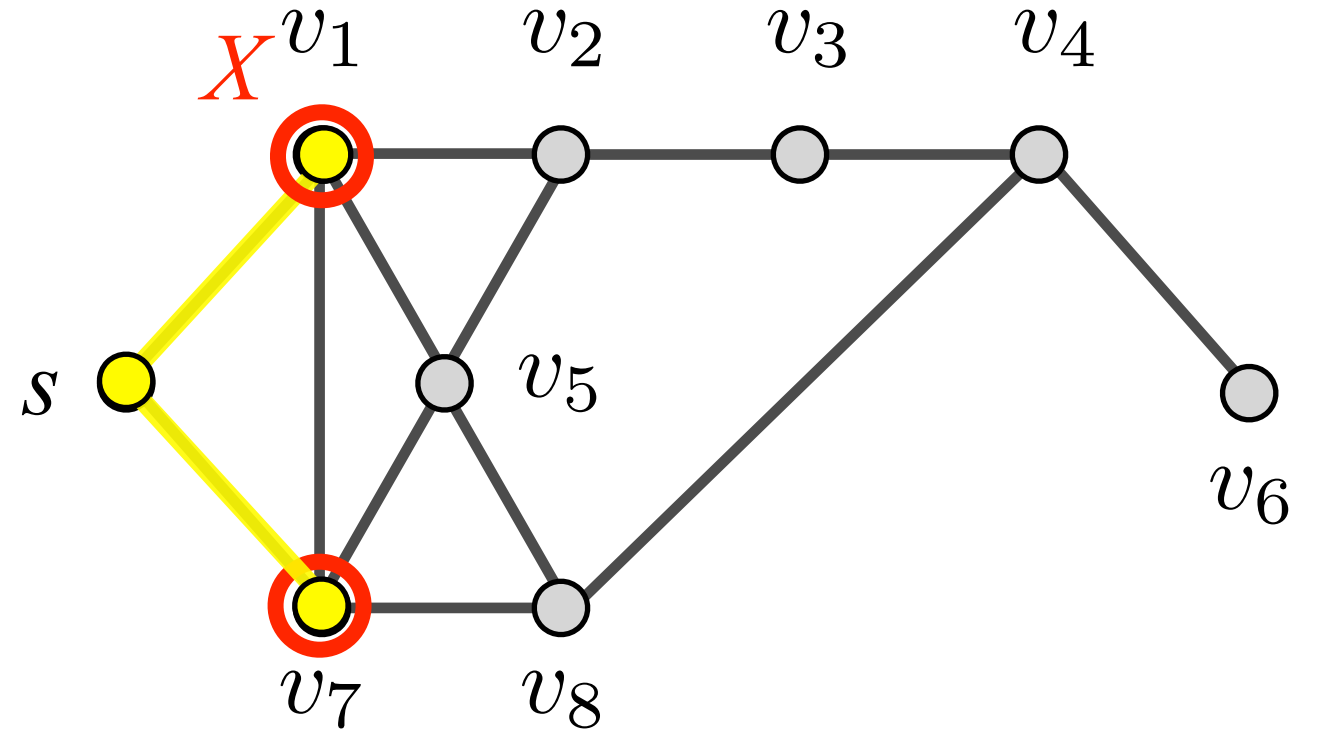
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

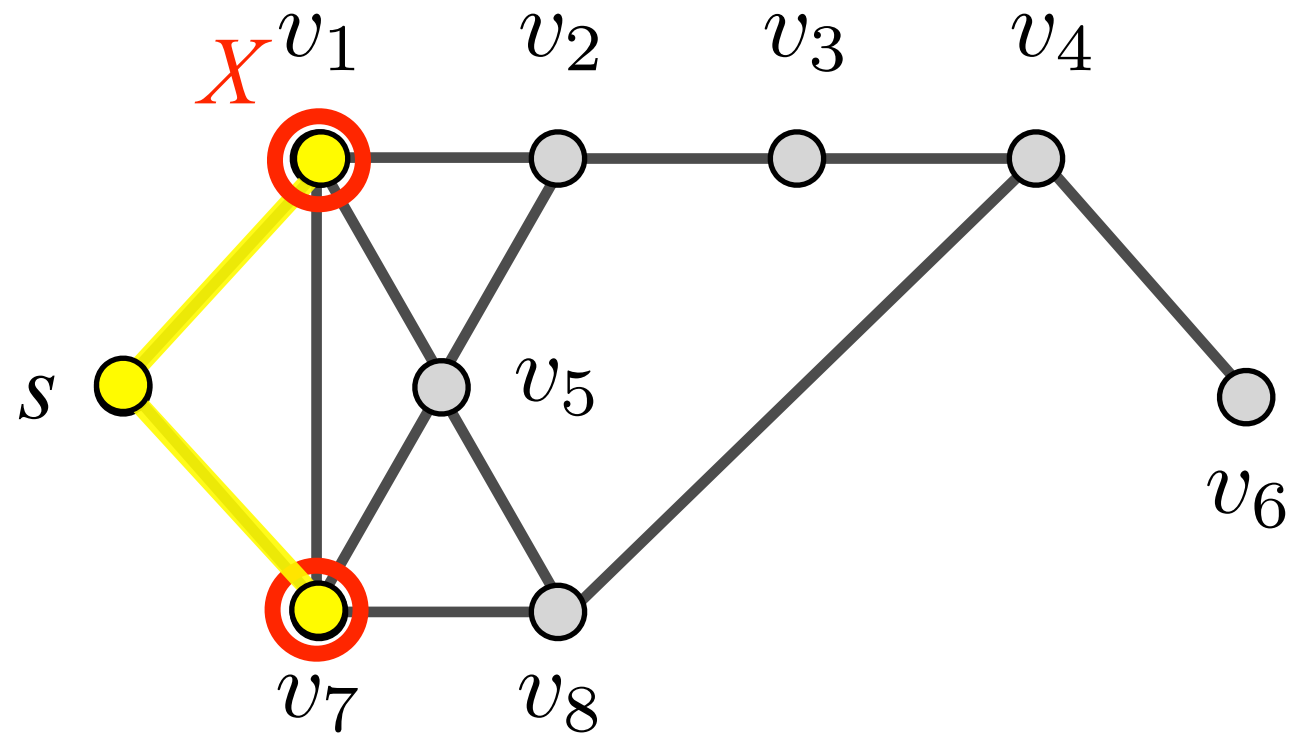
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

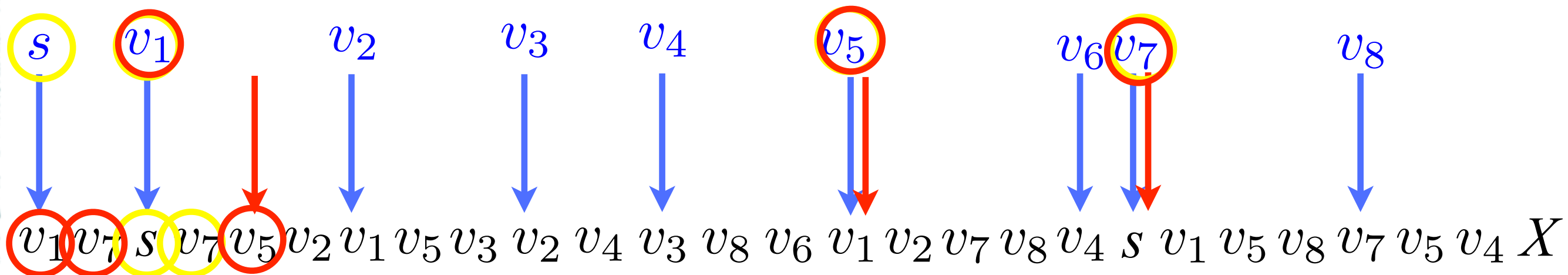
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

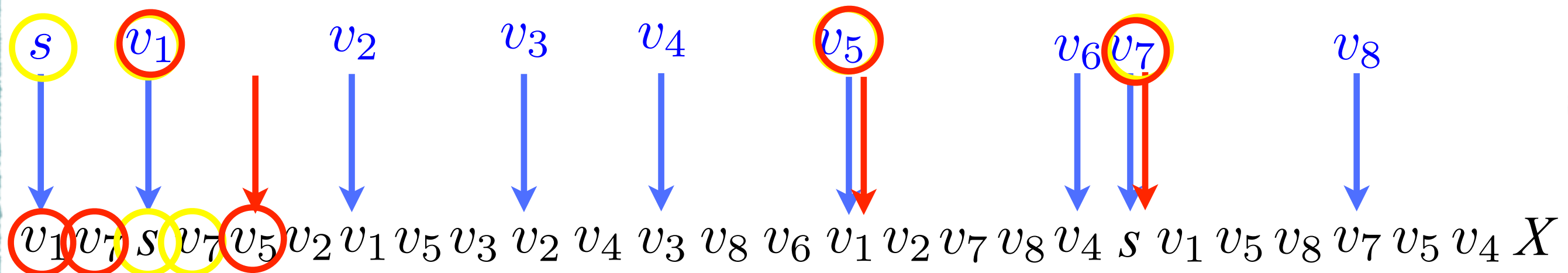
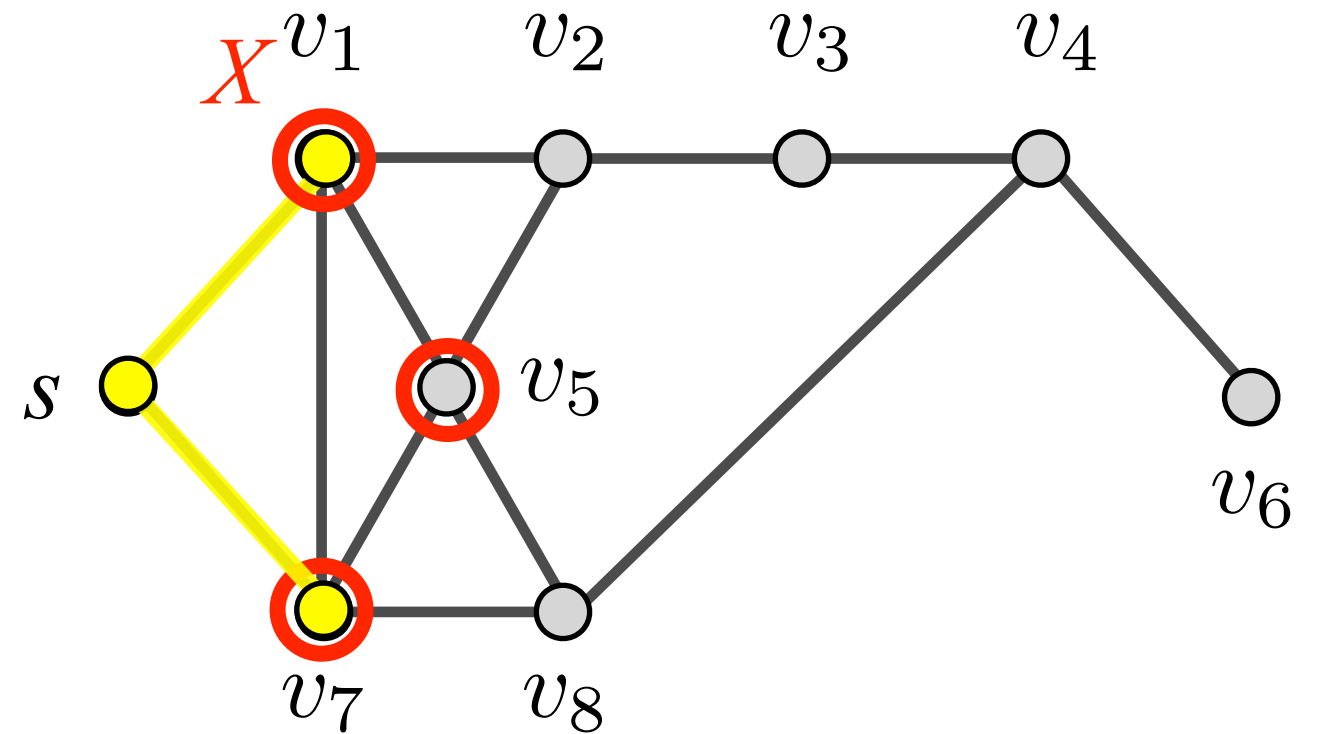
2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP

2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

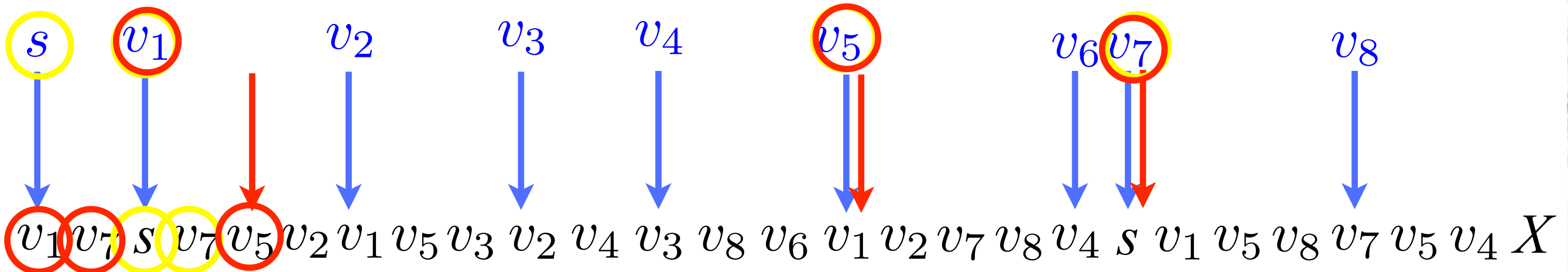
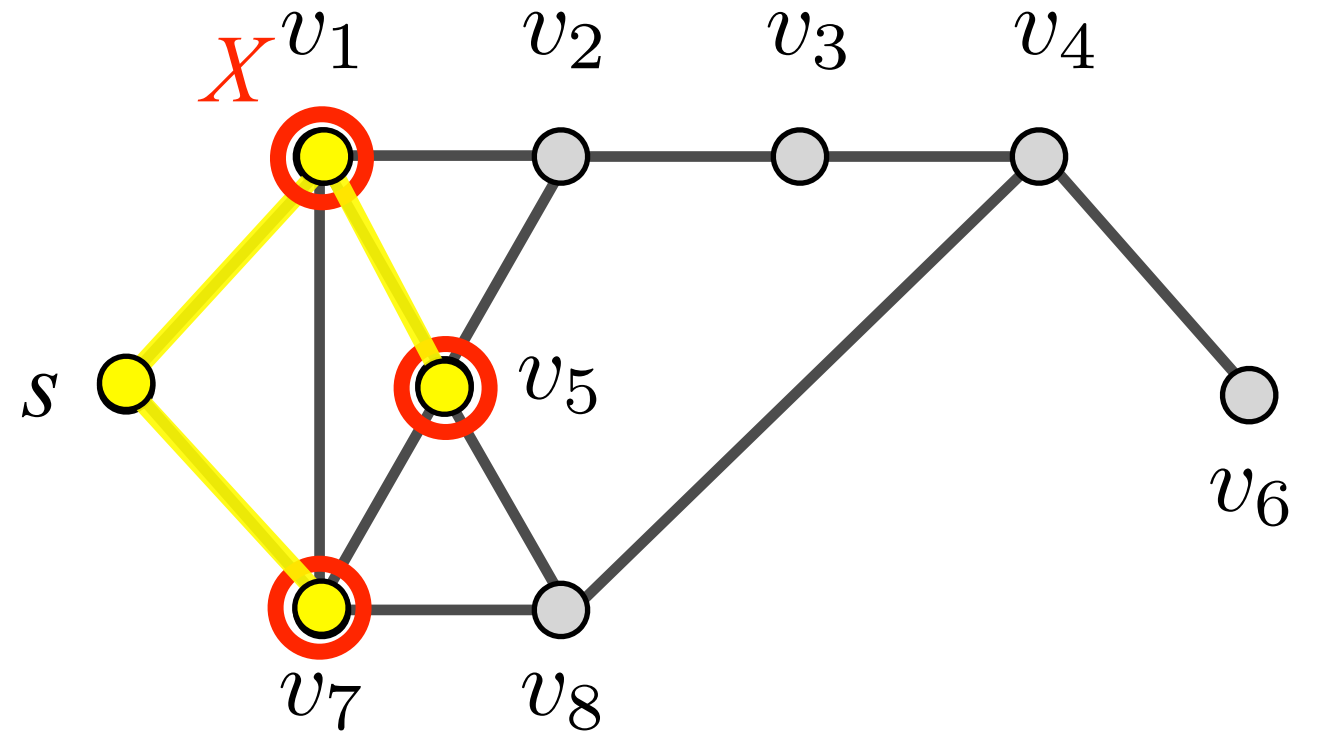
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

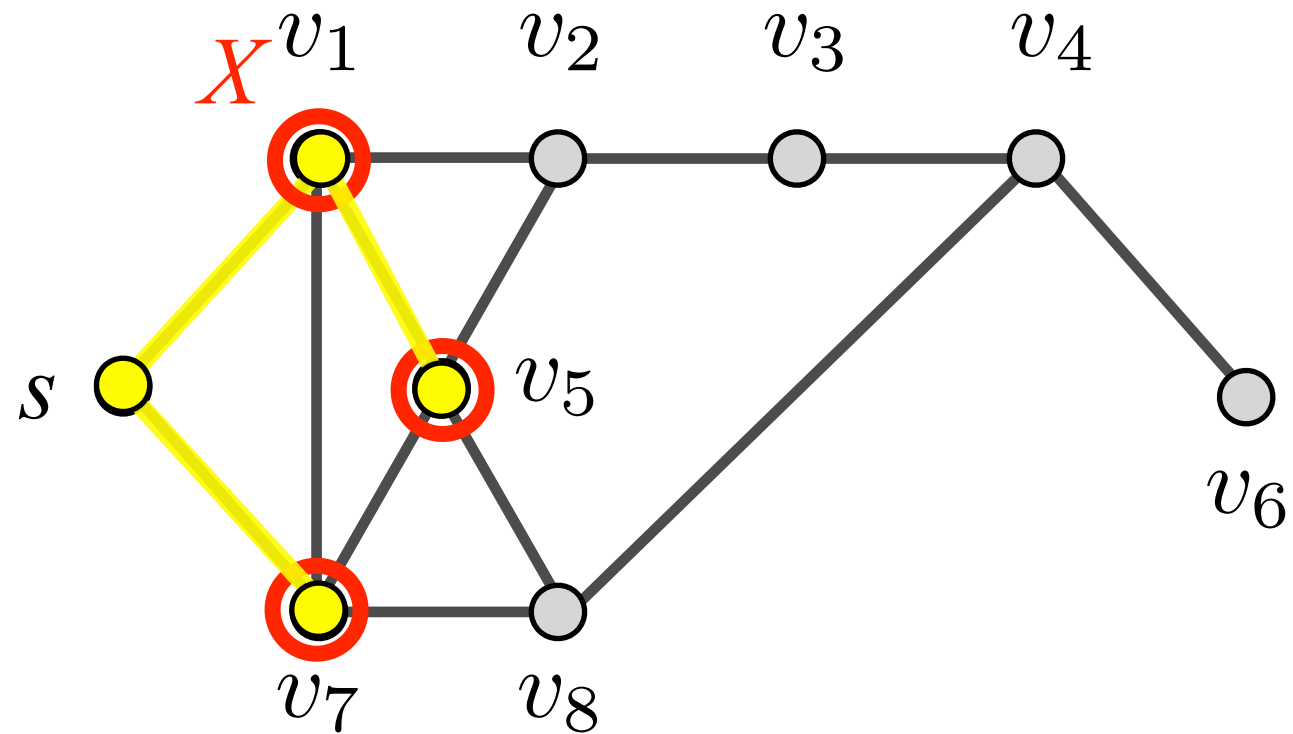
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

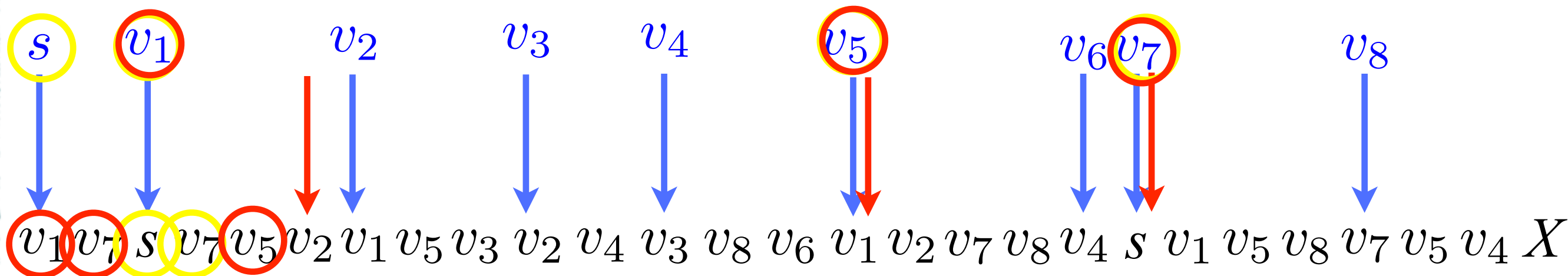
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

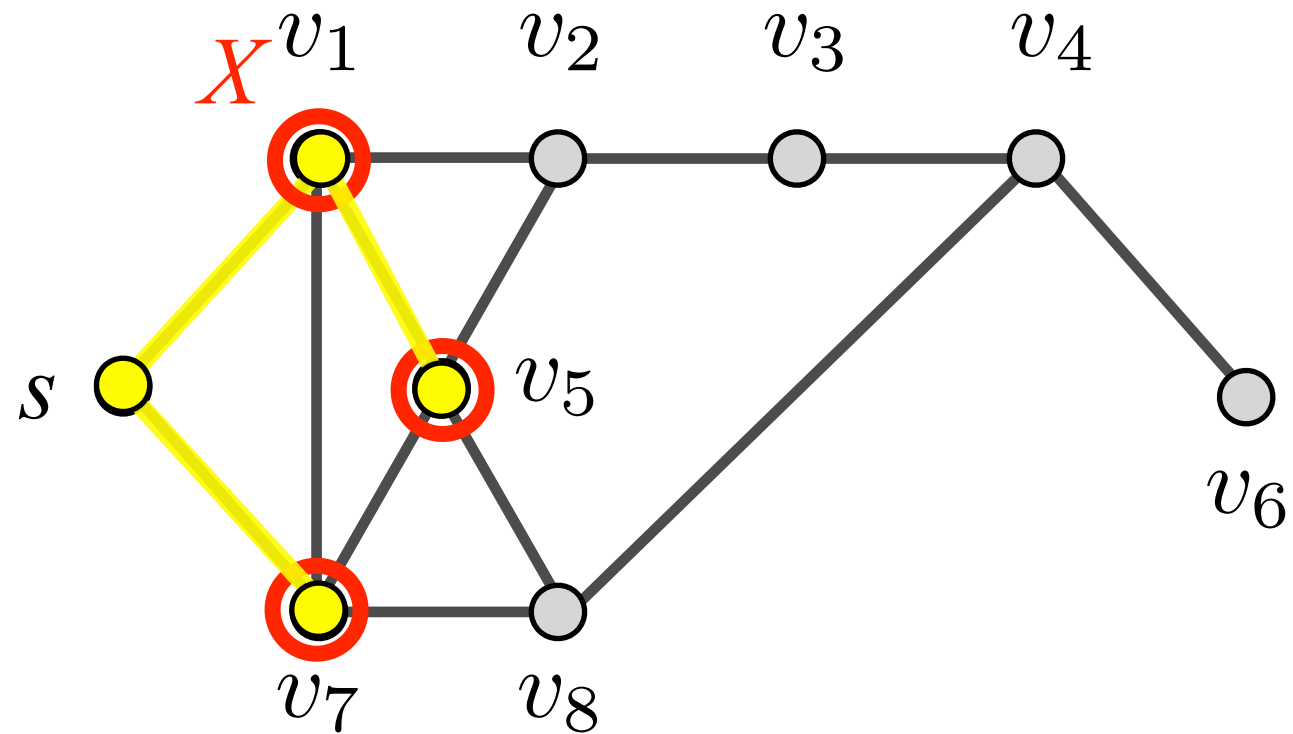
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

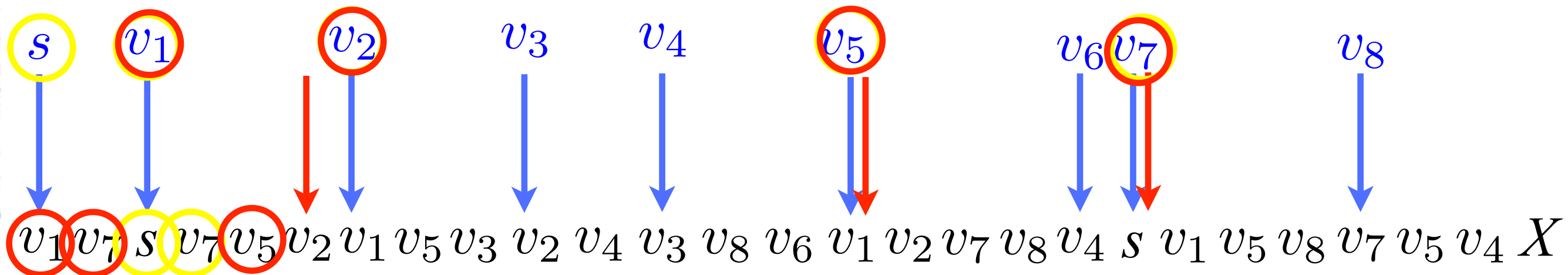
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

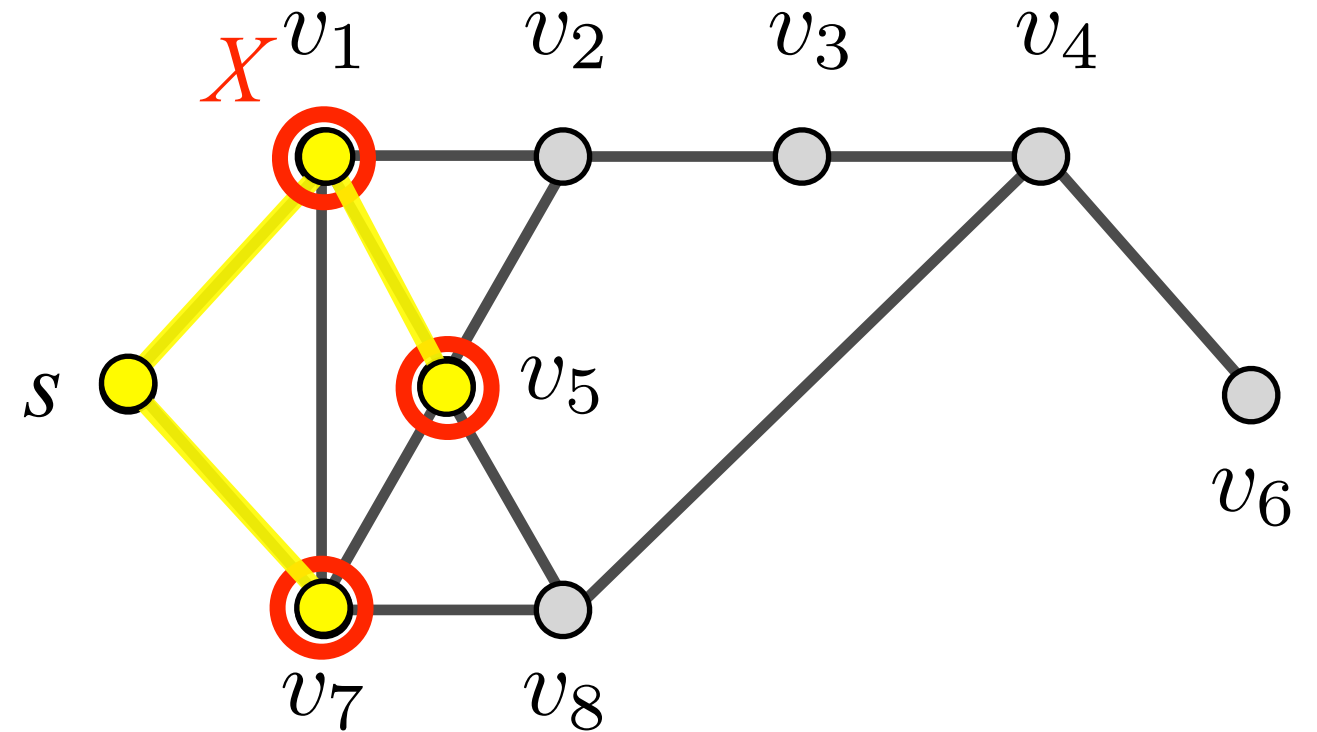
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

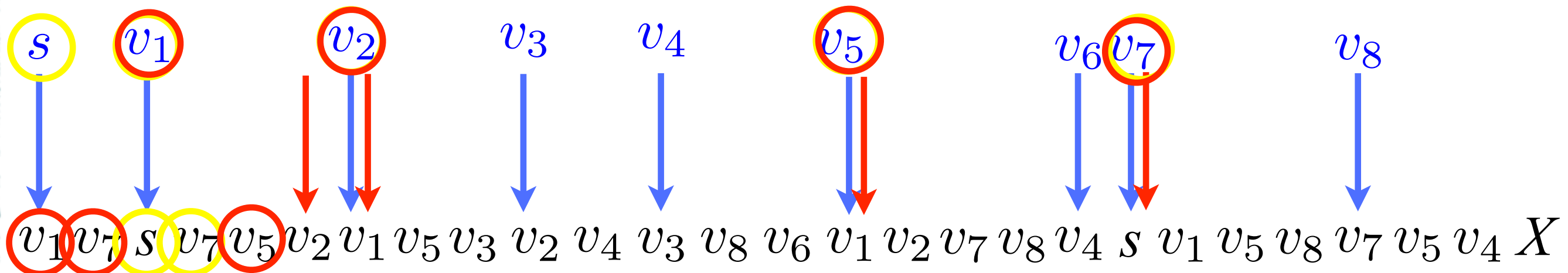
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

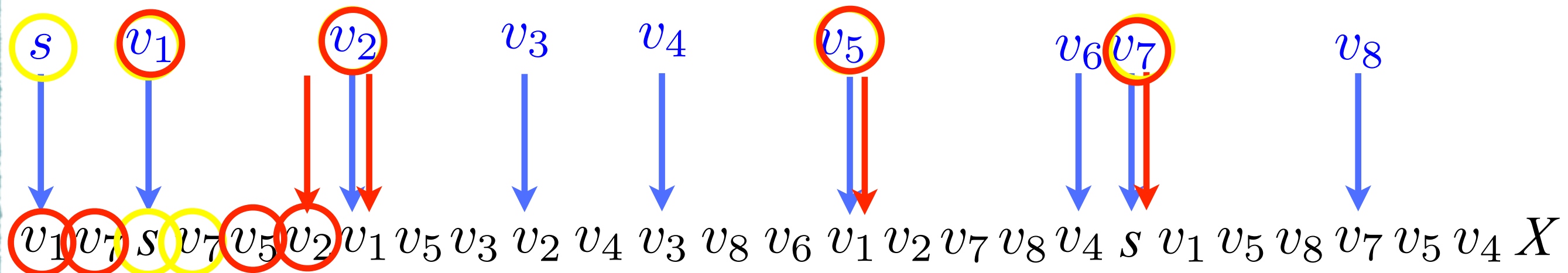
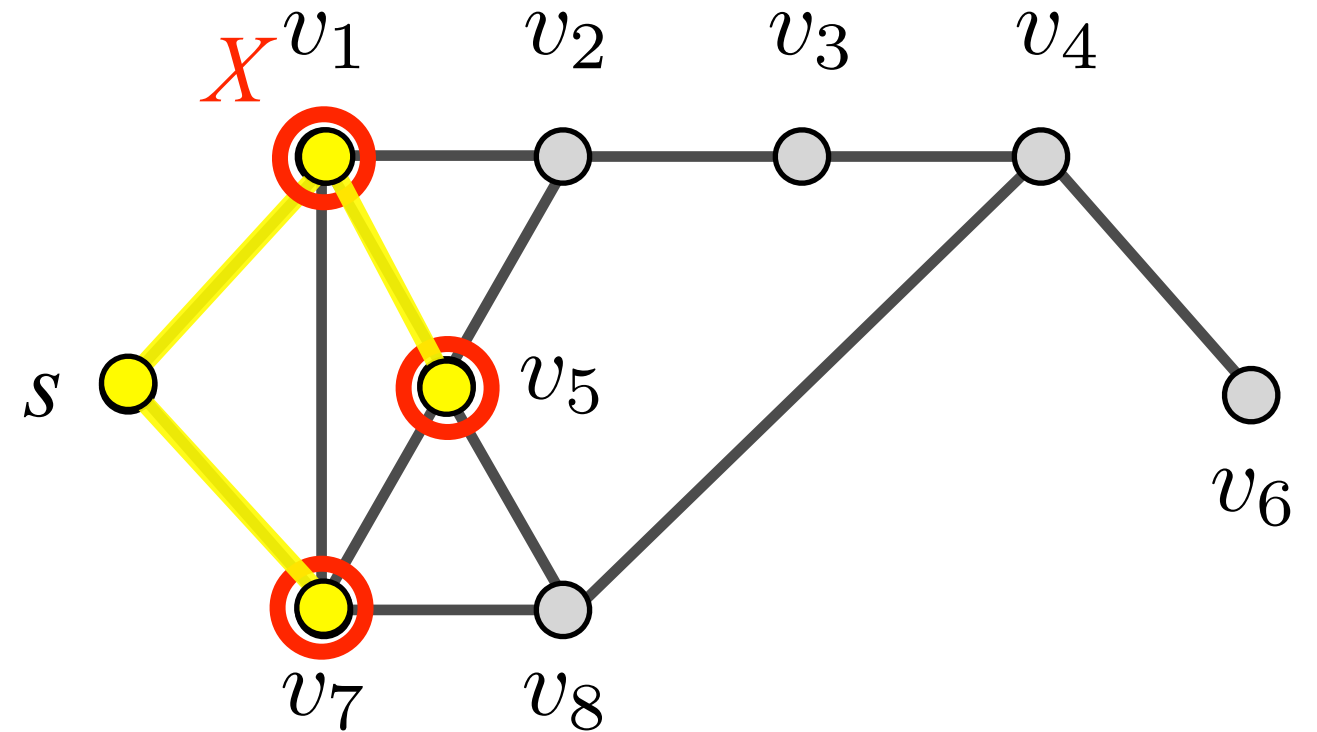
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

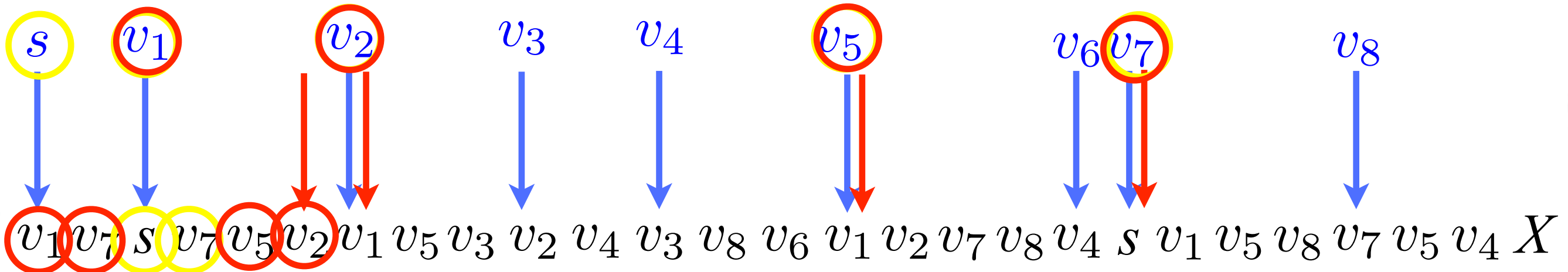
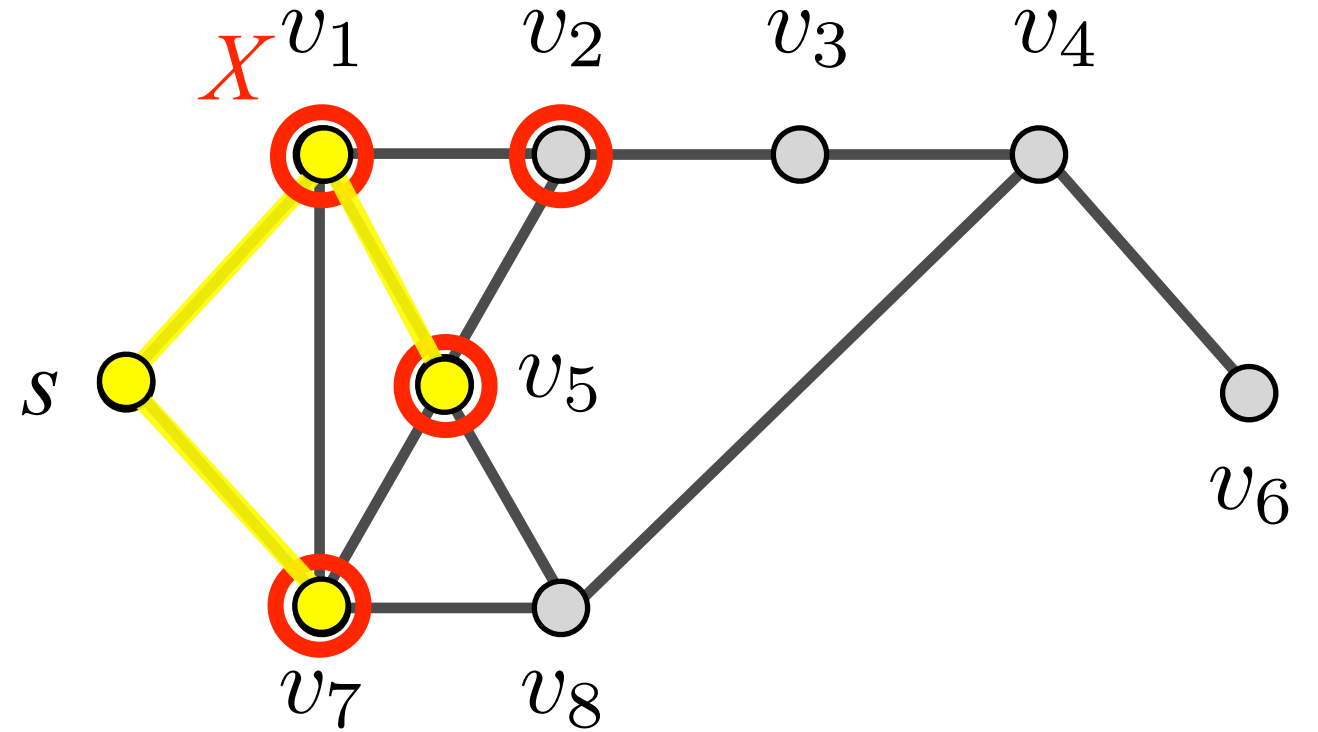
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

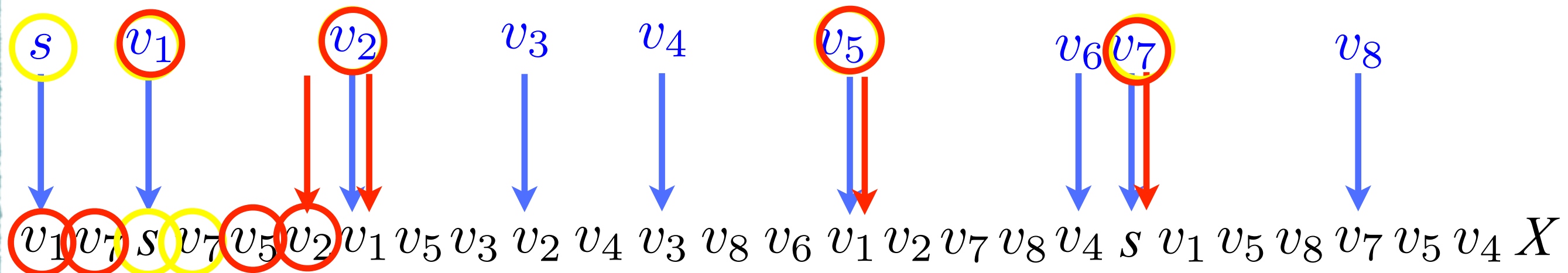
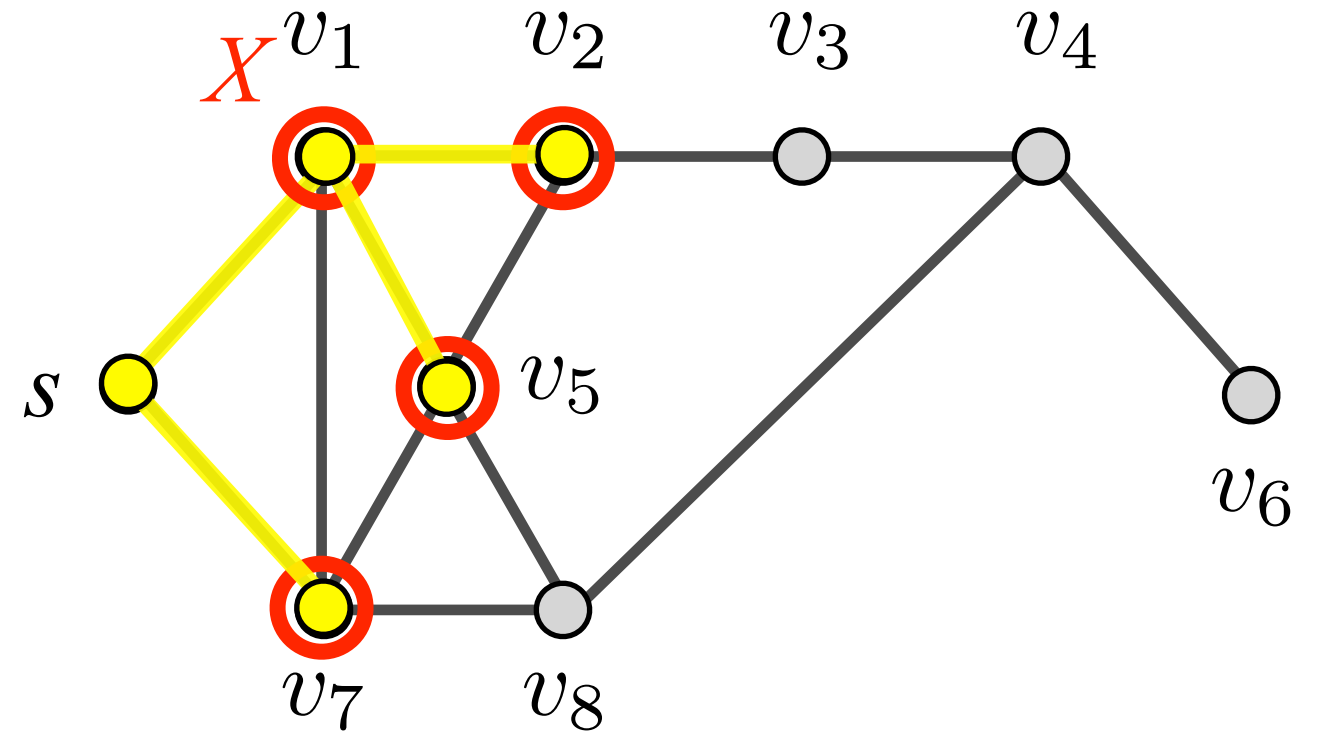
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

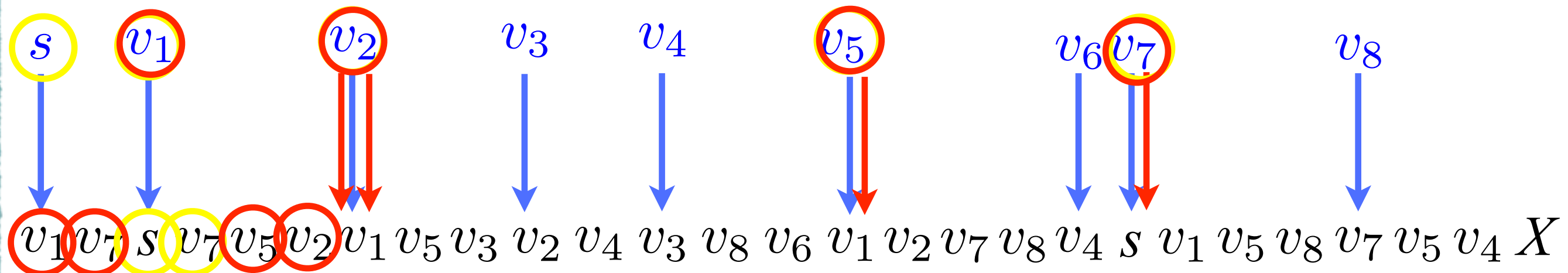
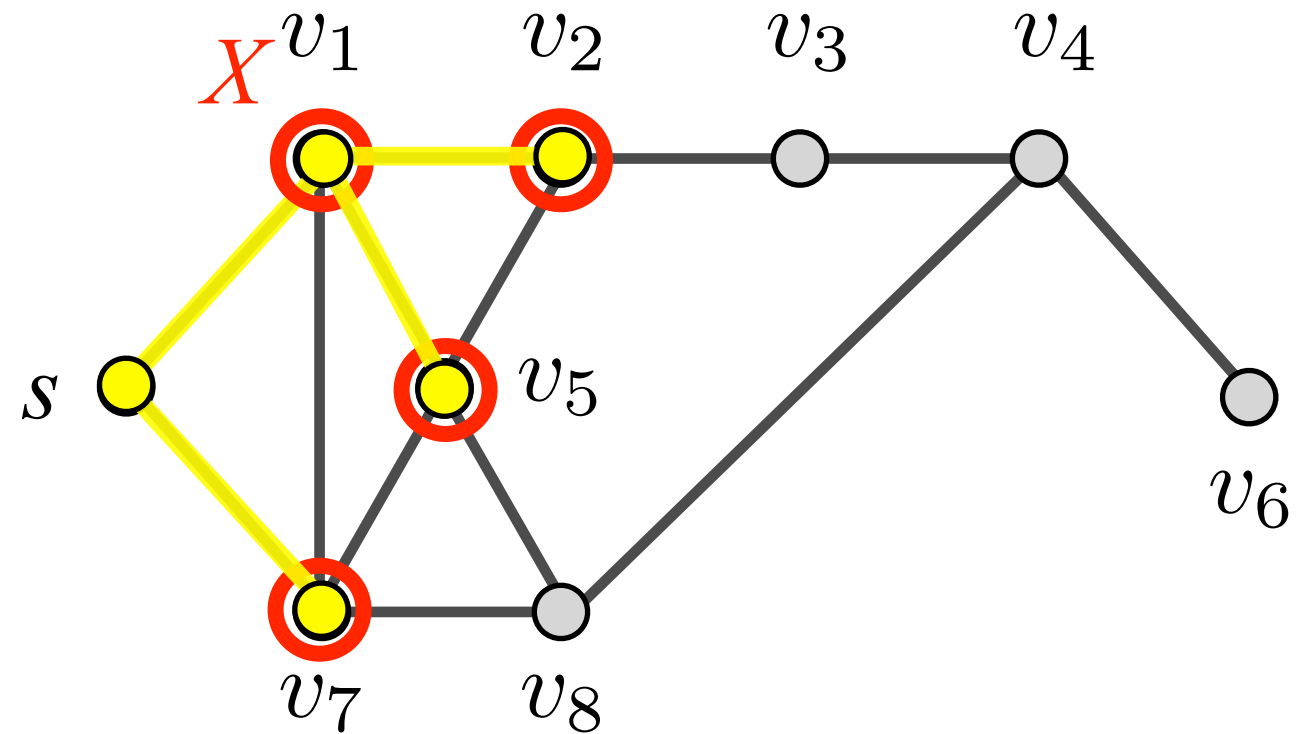
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

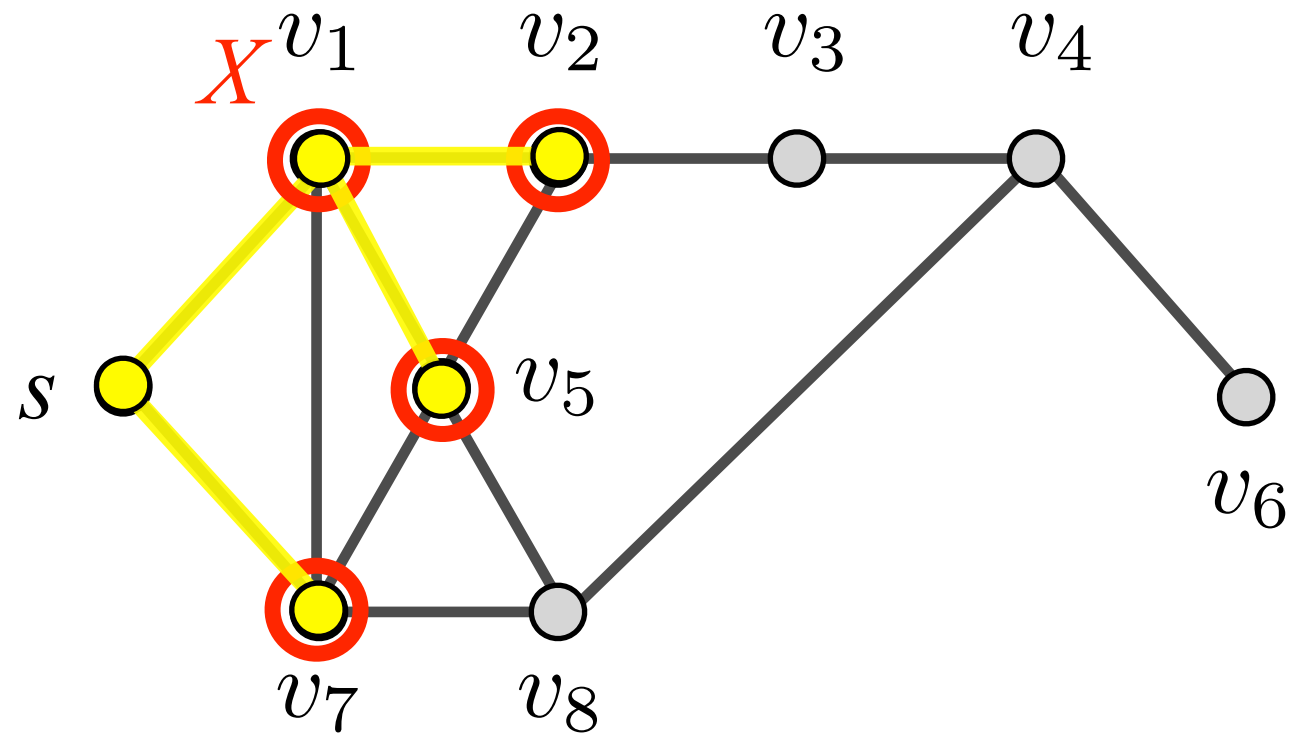
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

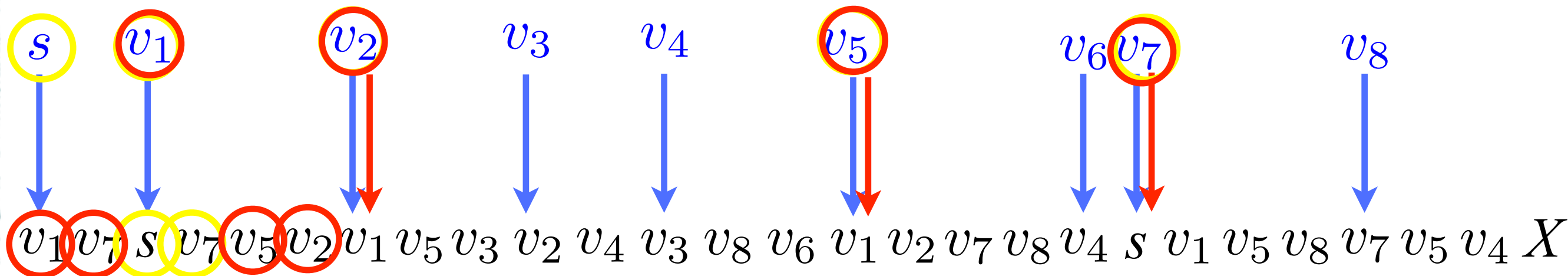
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

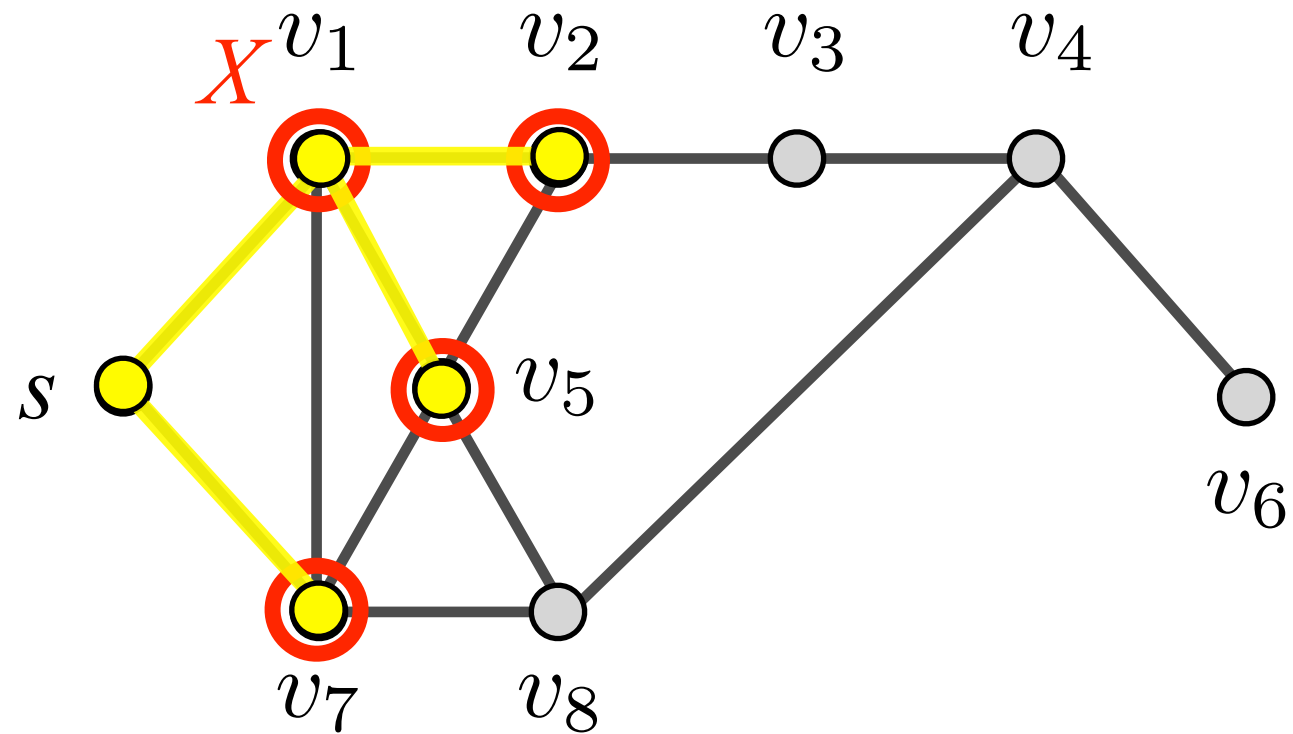
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

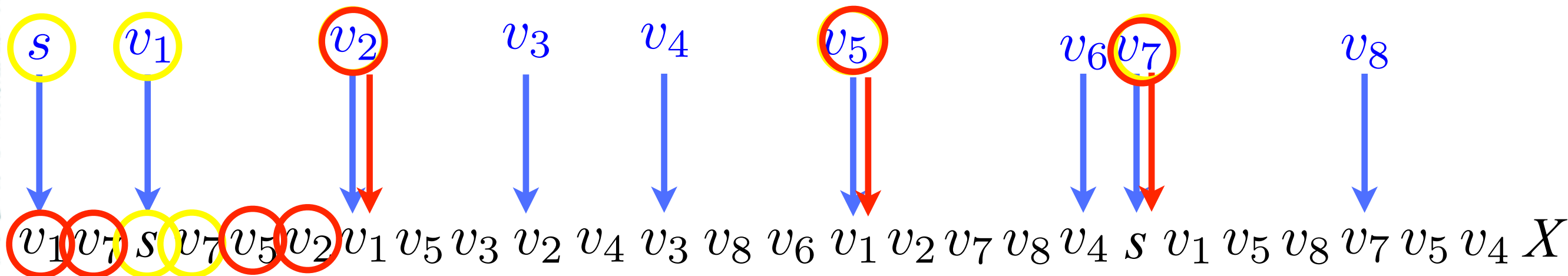
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

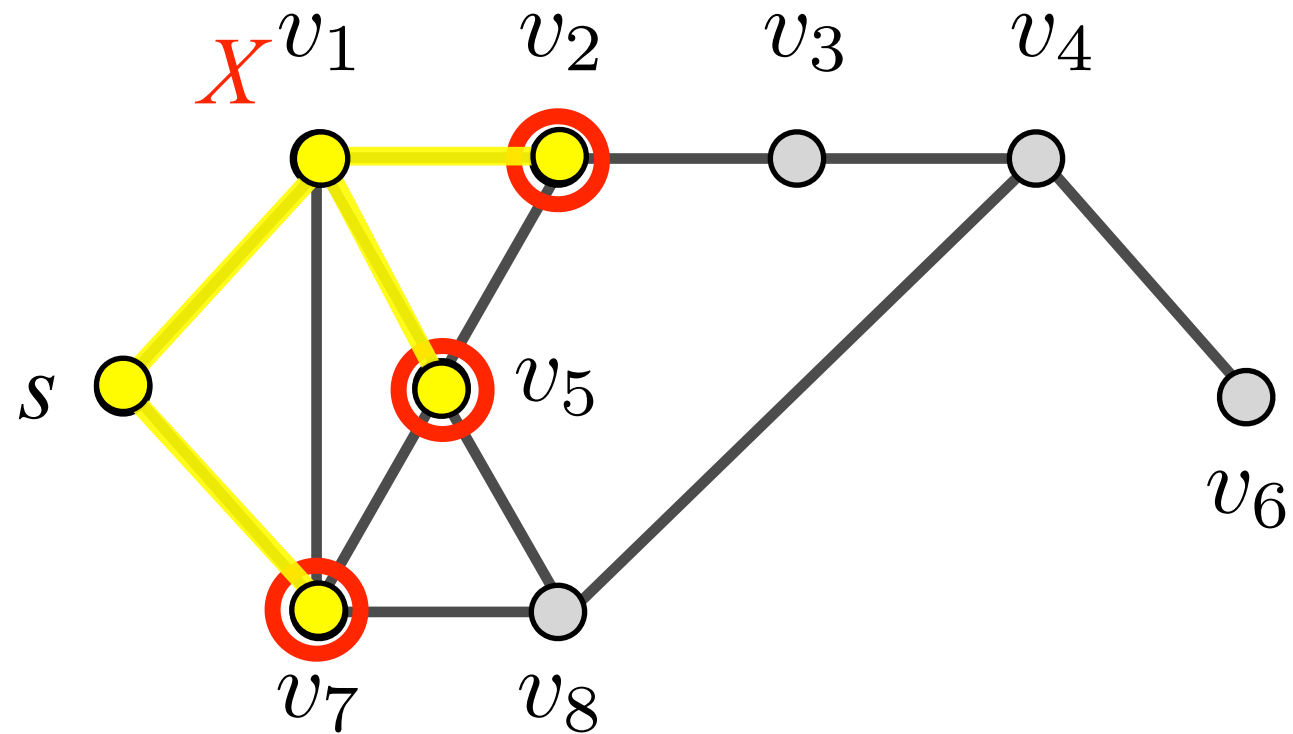
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

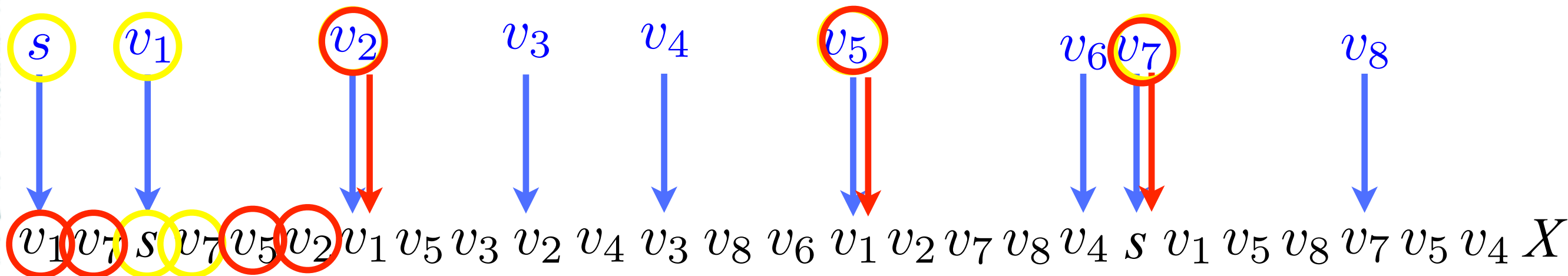
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

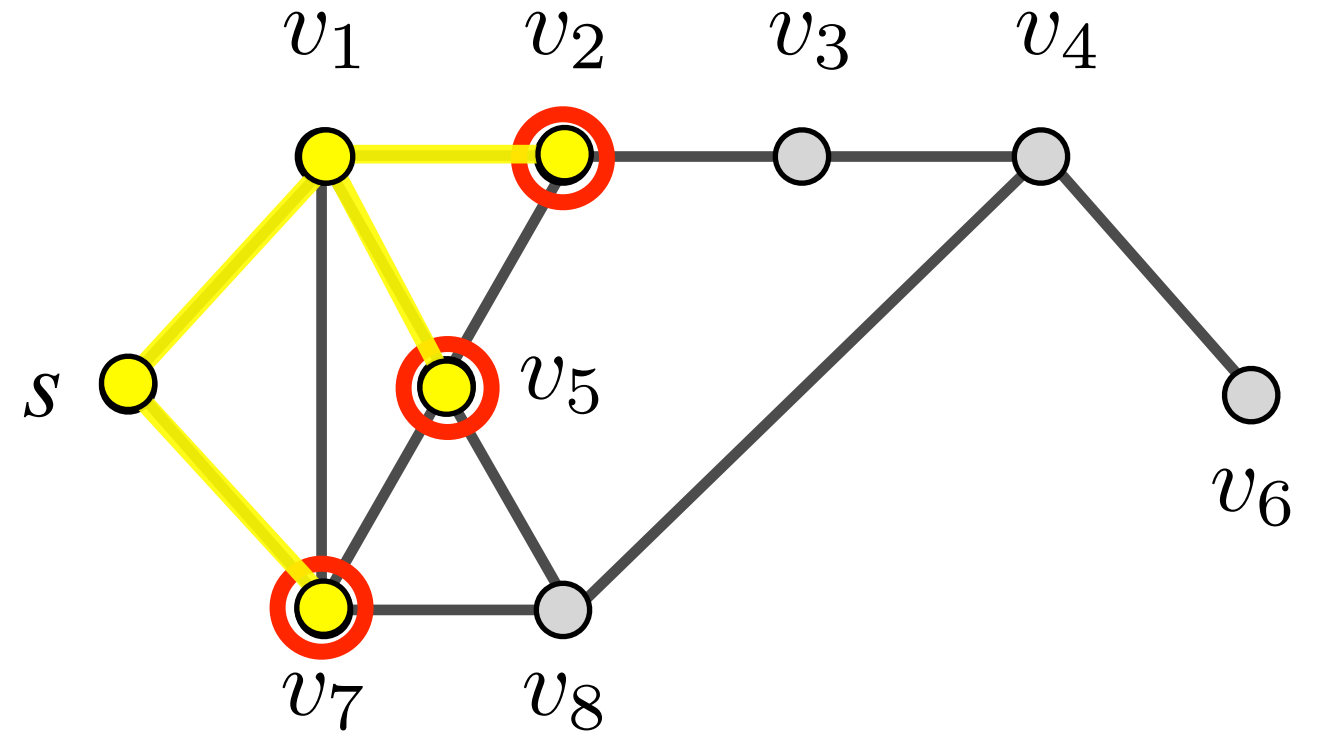
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

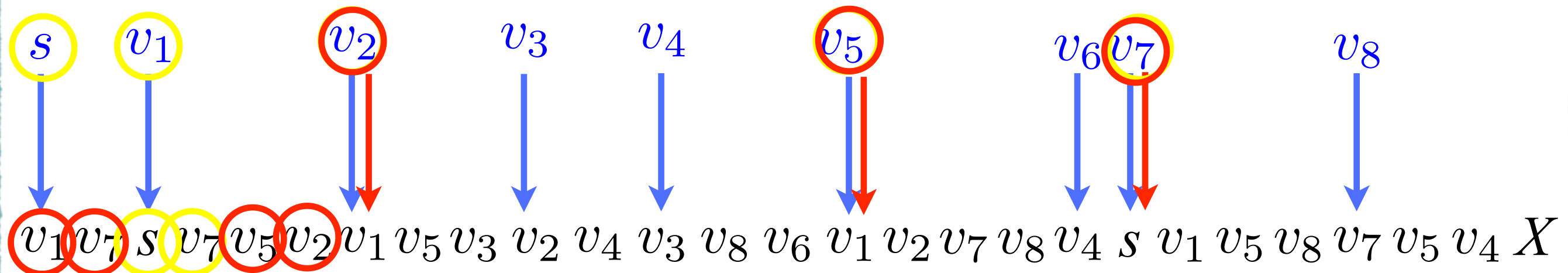
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

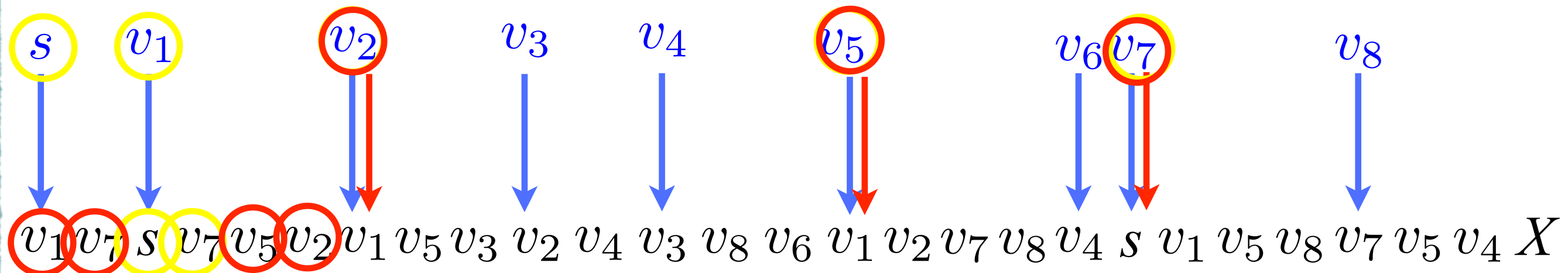
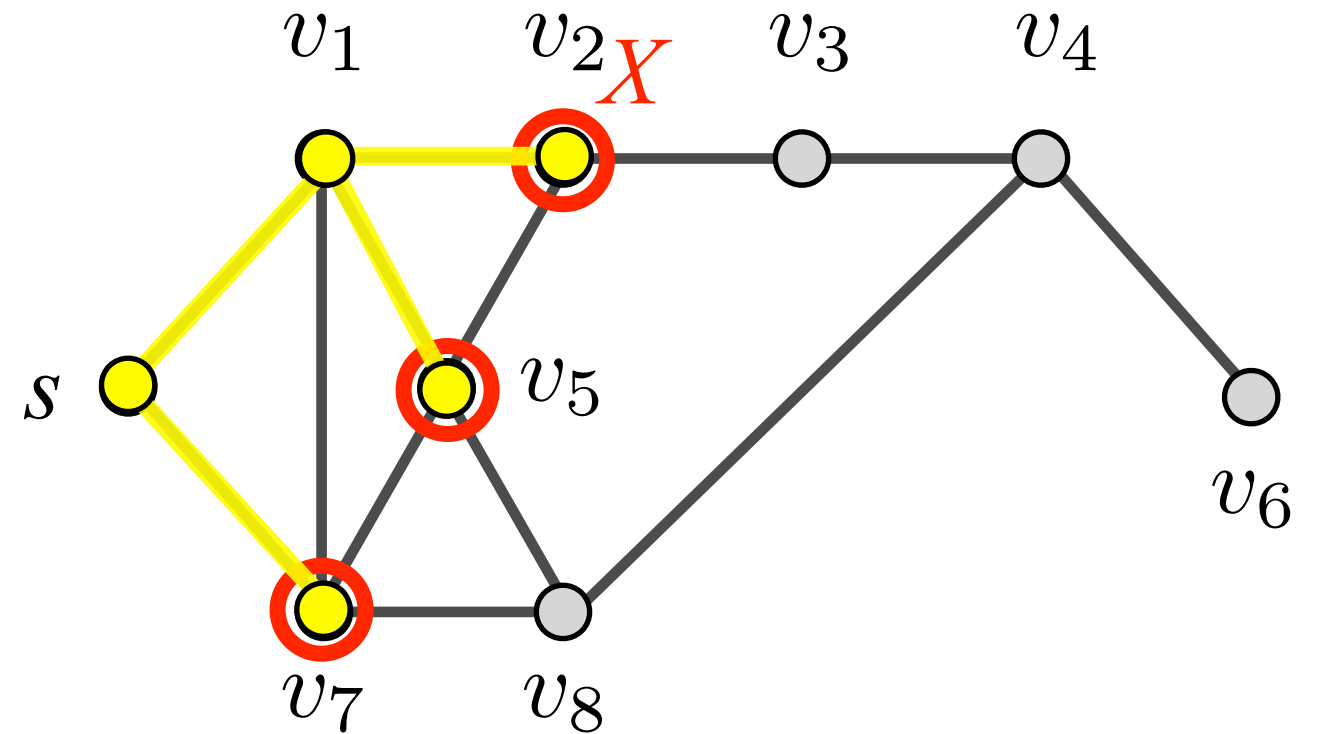
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

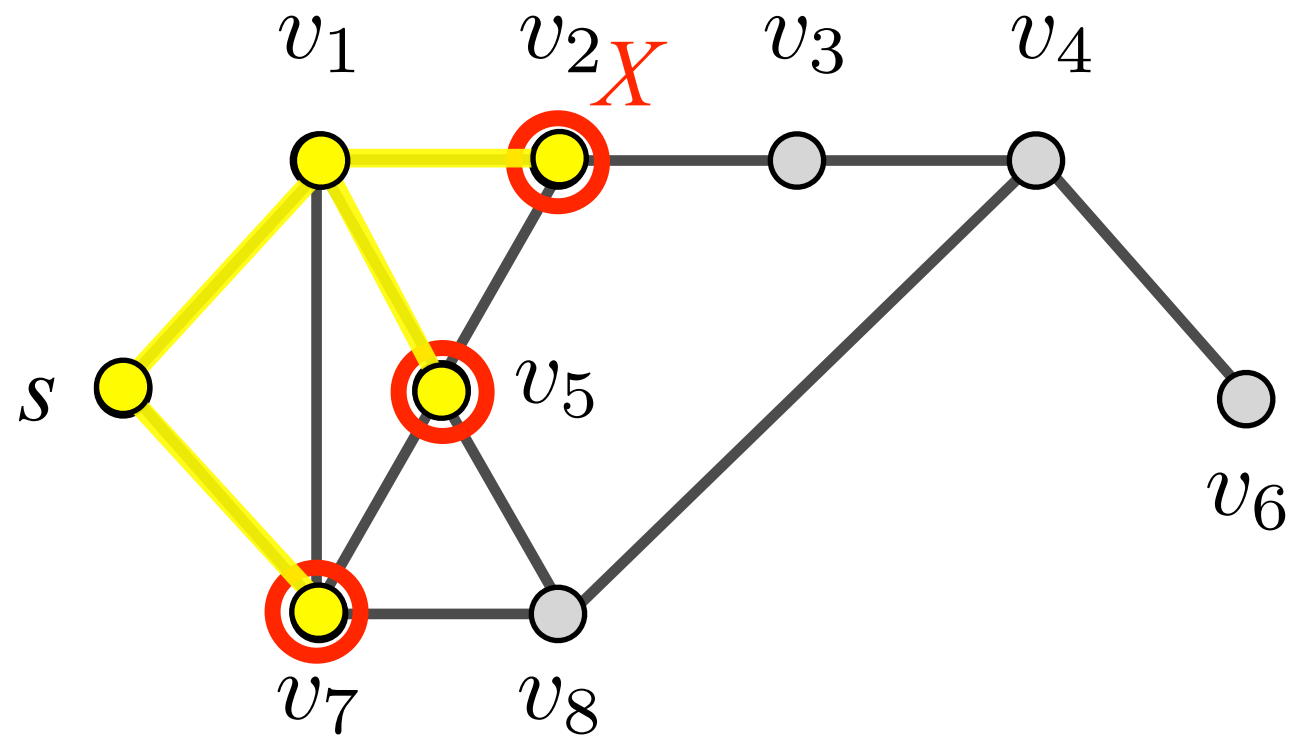
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

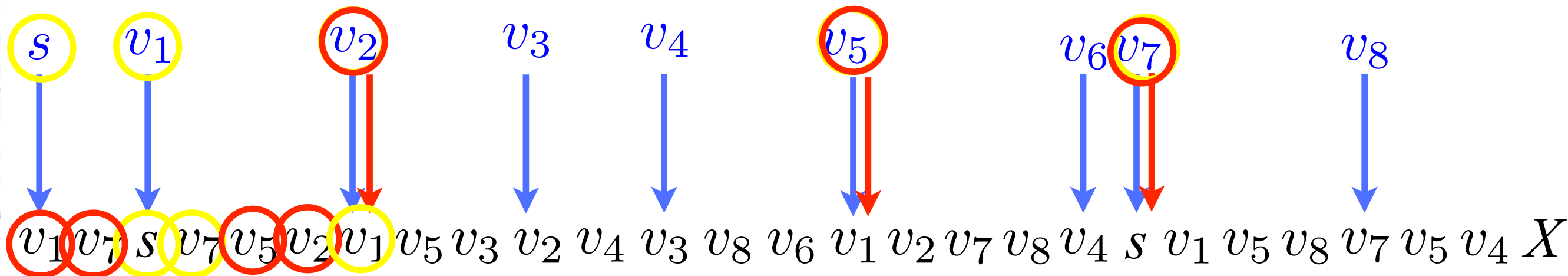
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

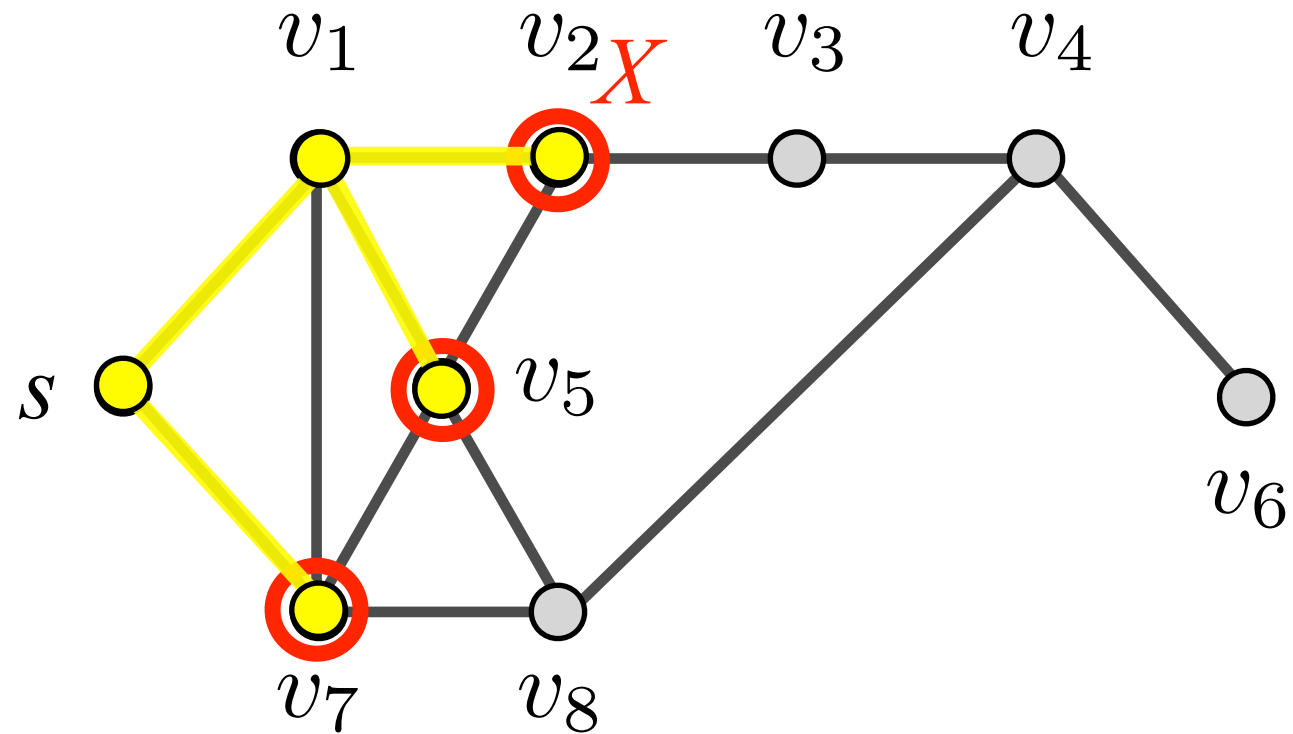
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

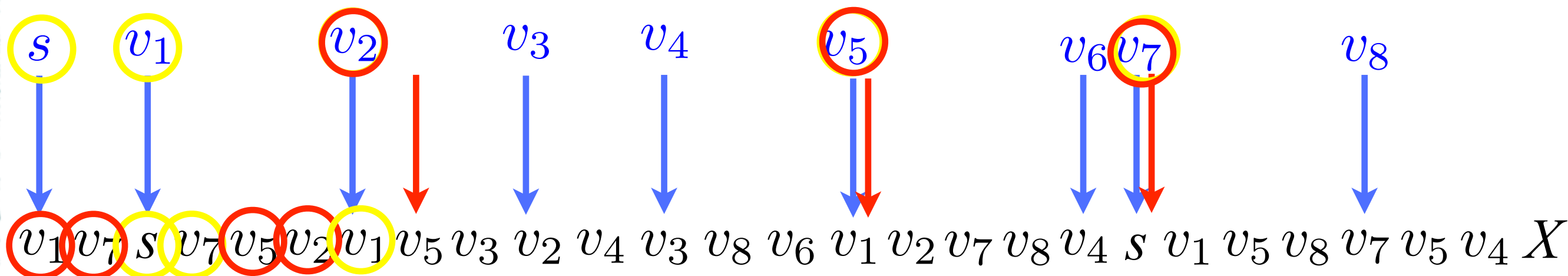
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

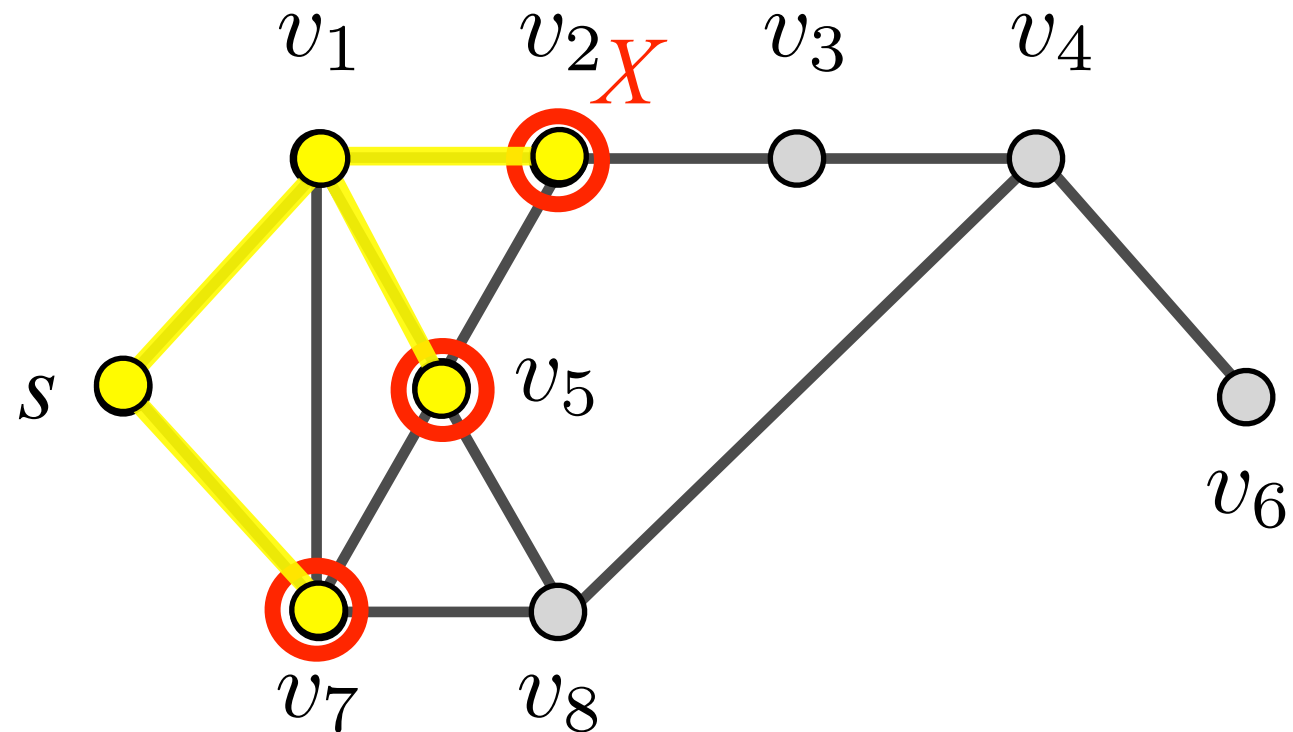
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

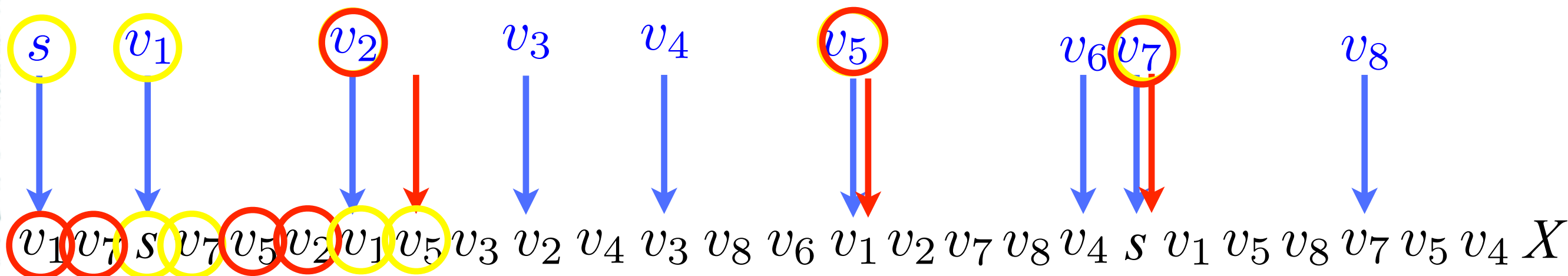
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

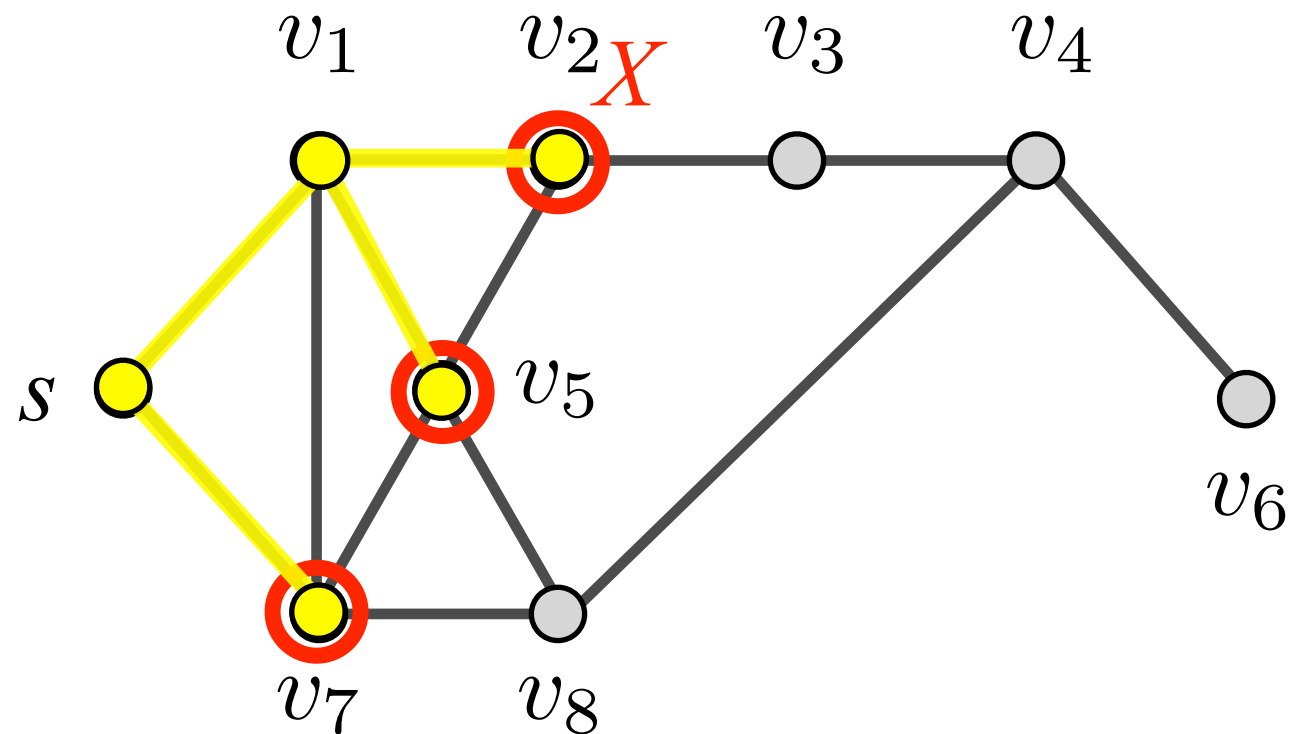
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

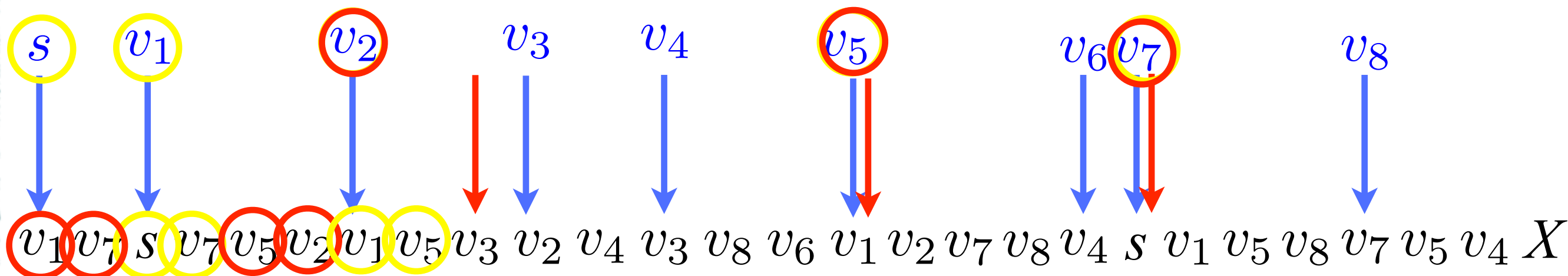
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

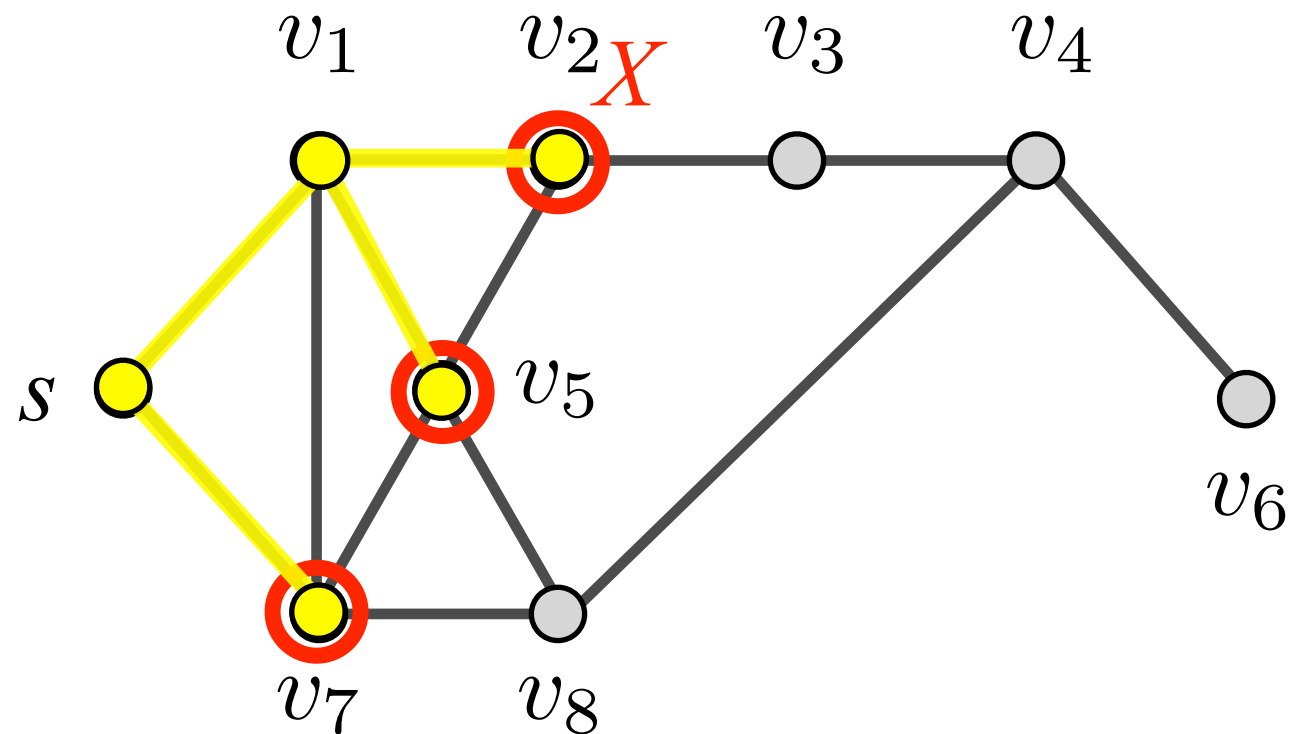
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

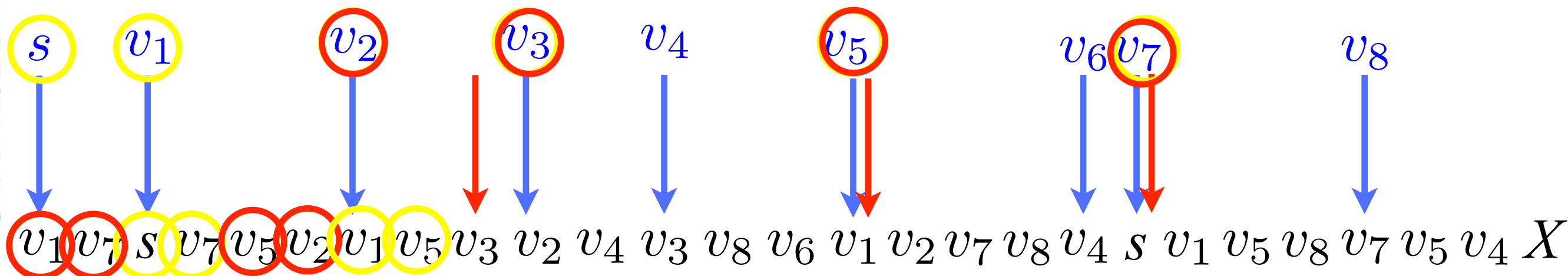
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

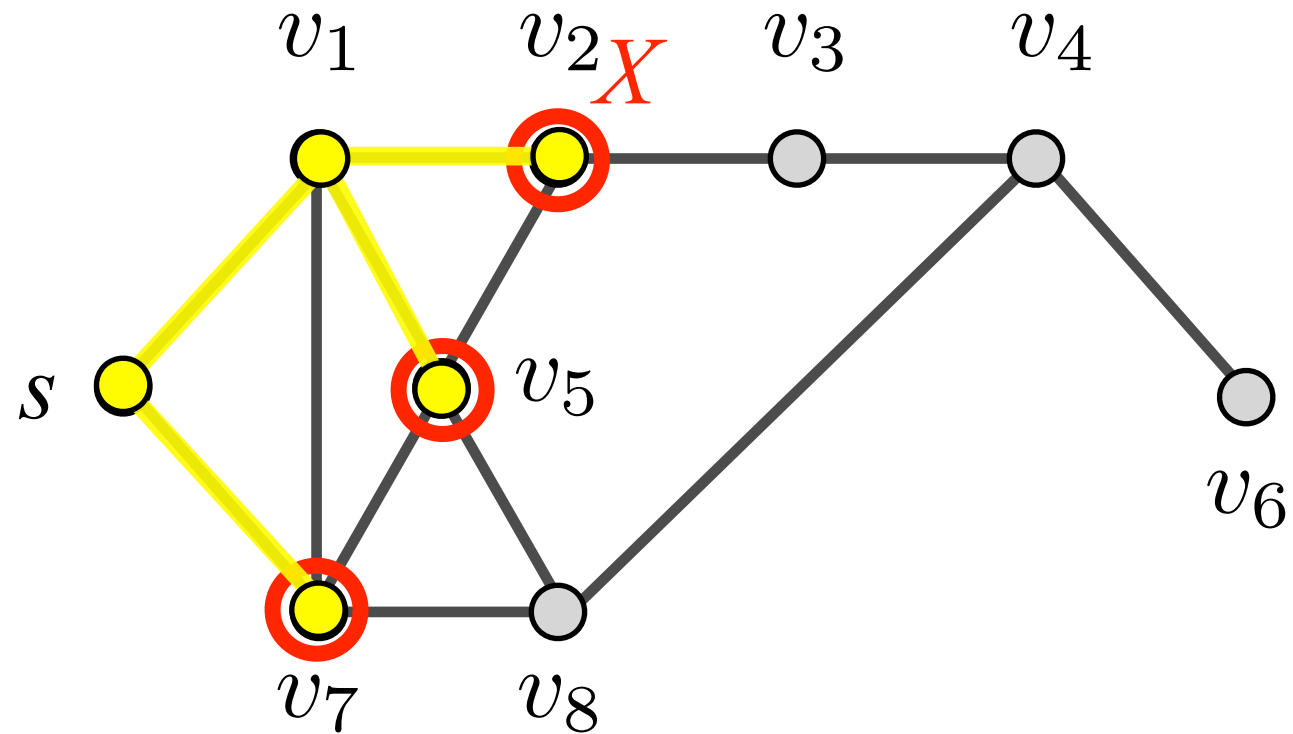
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

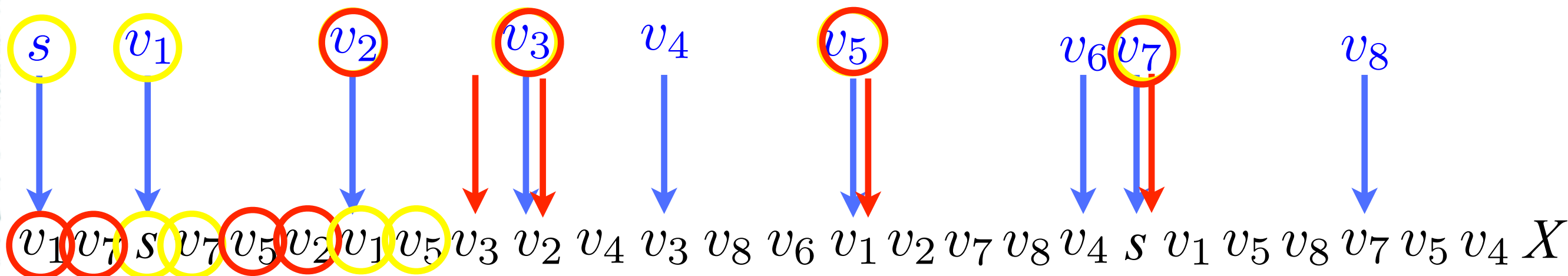
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

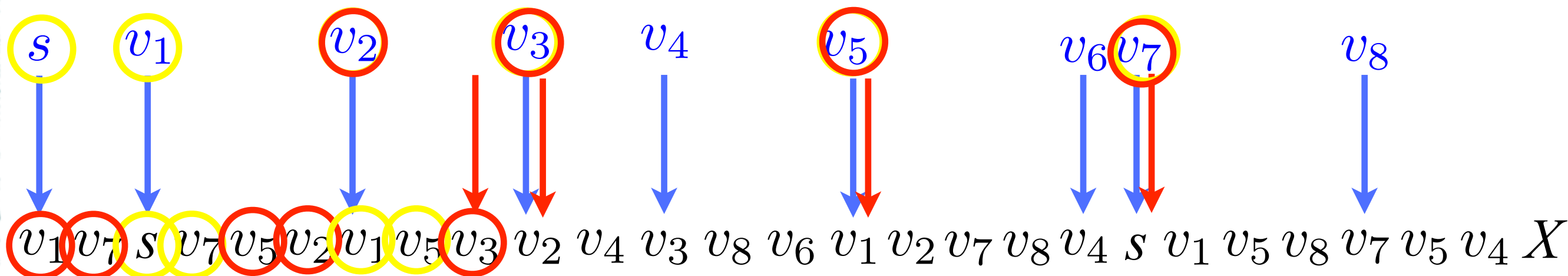
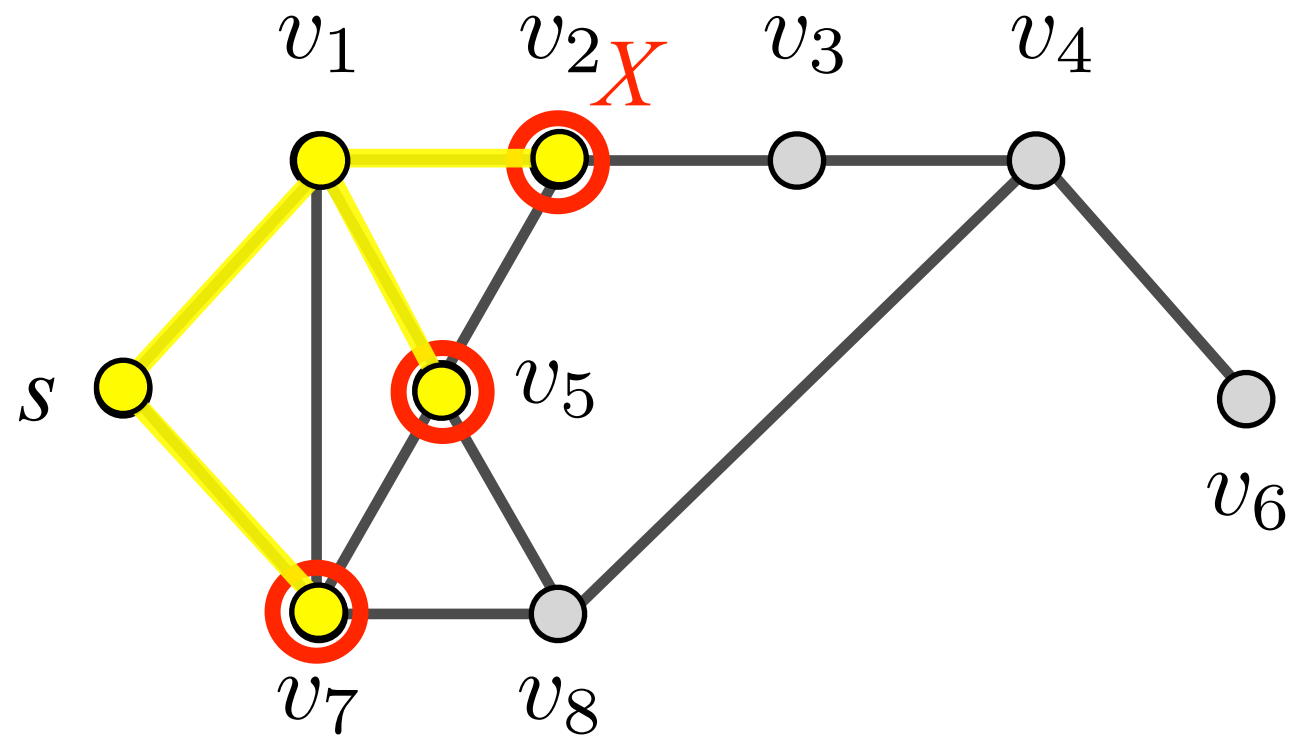
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

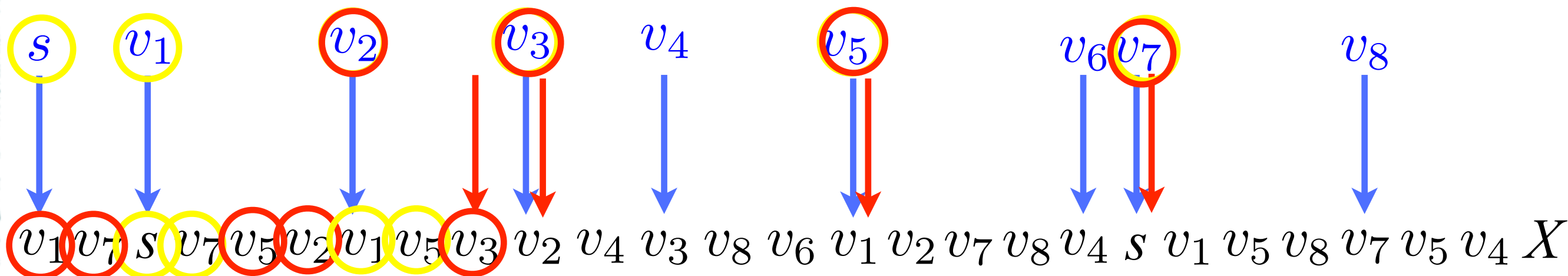
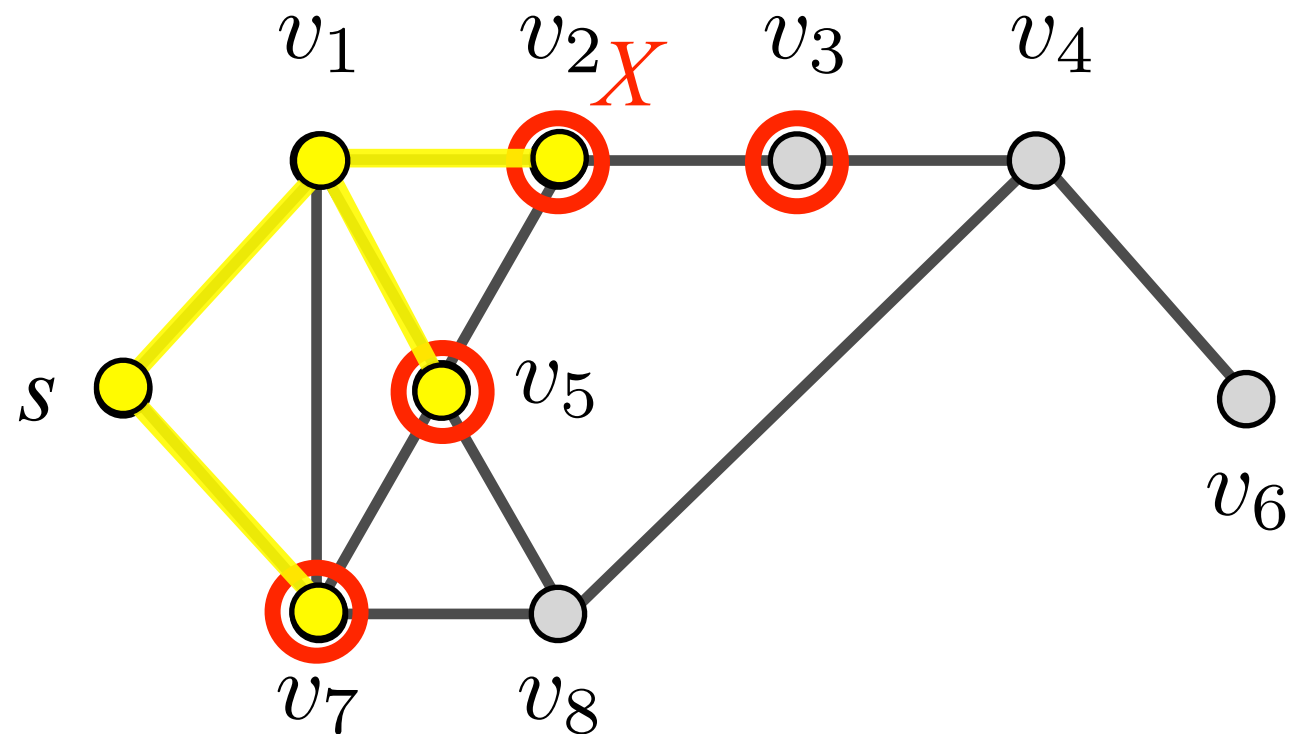
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

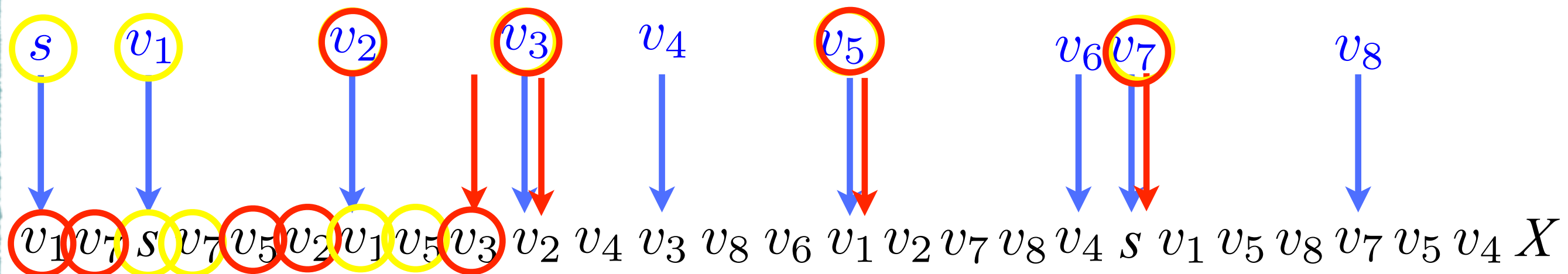
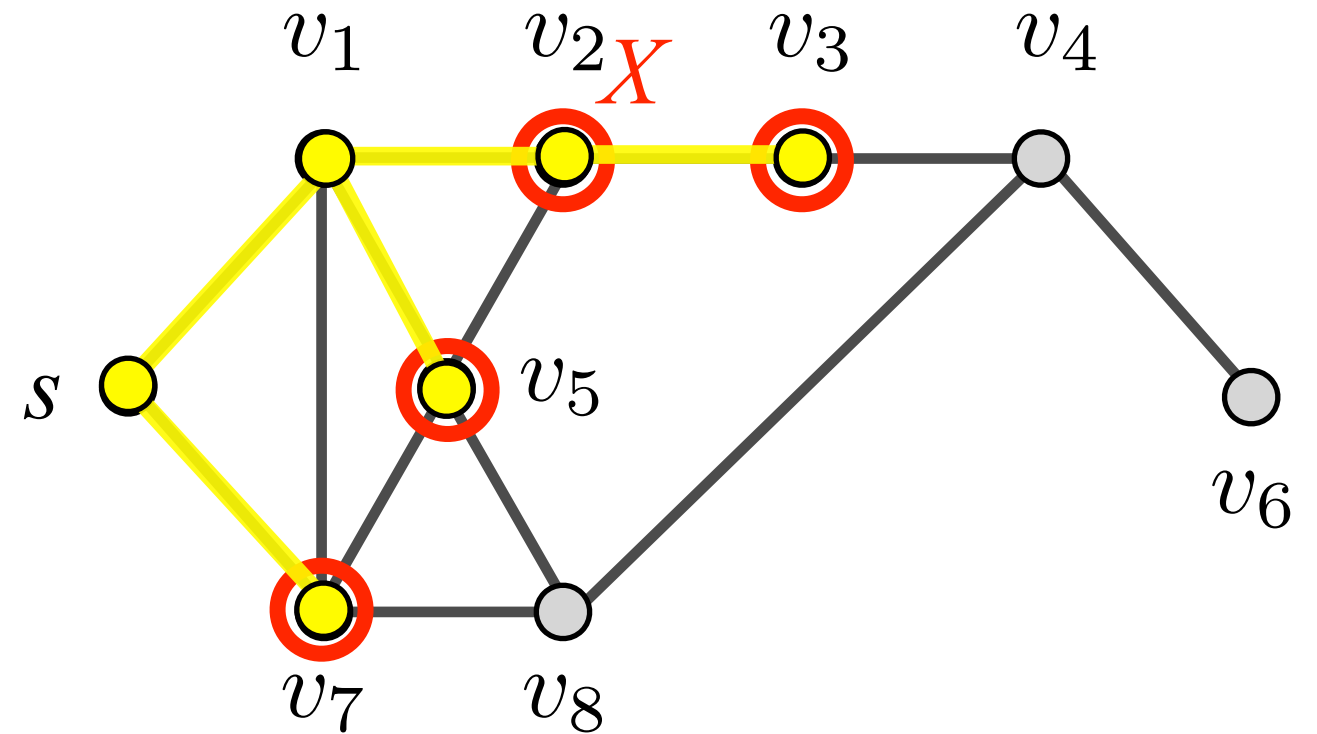
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

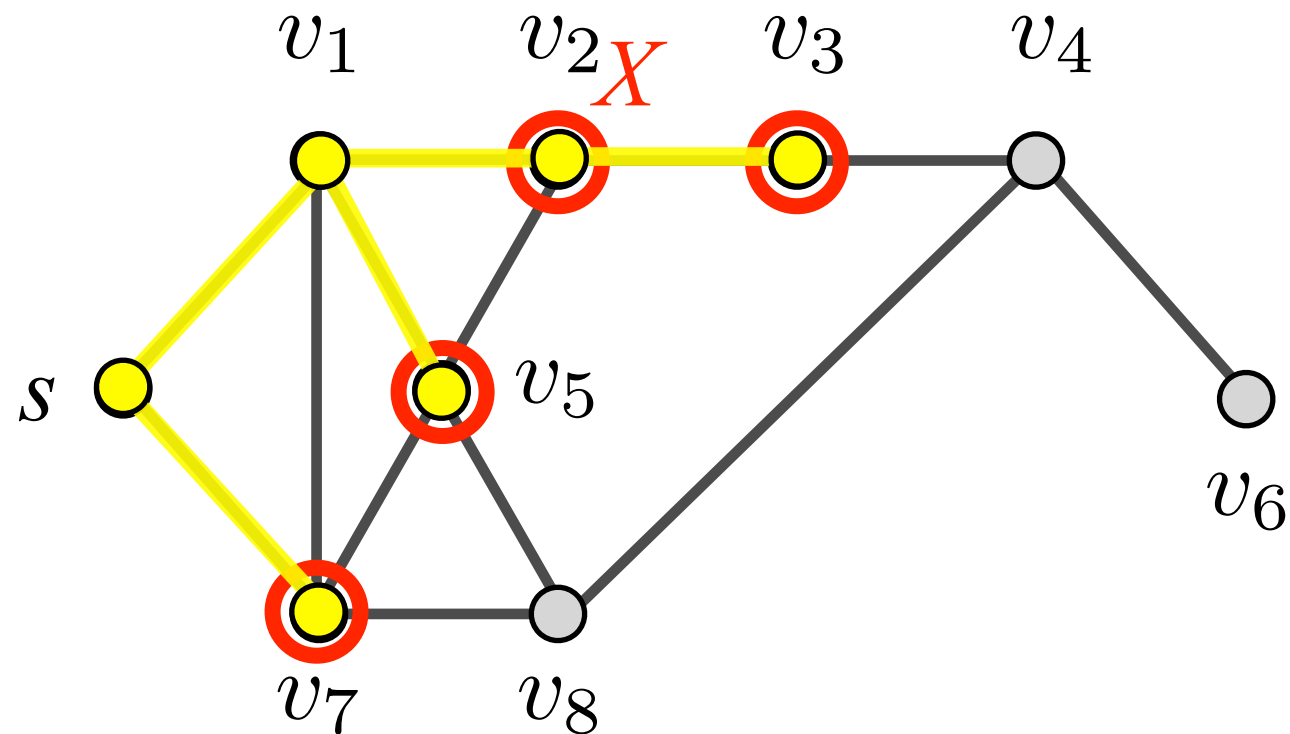
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

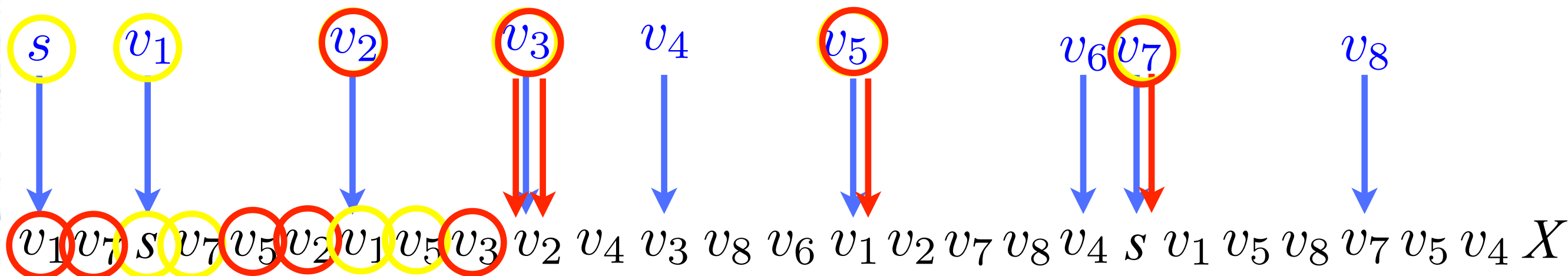
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

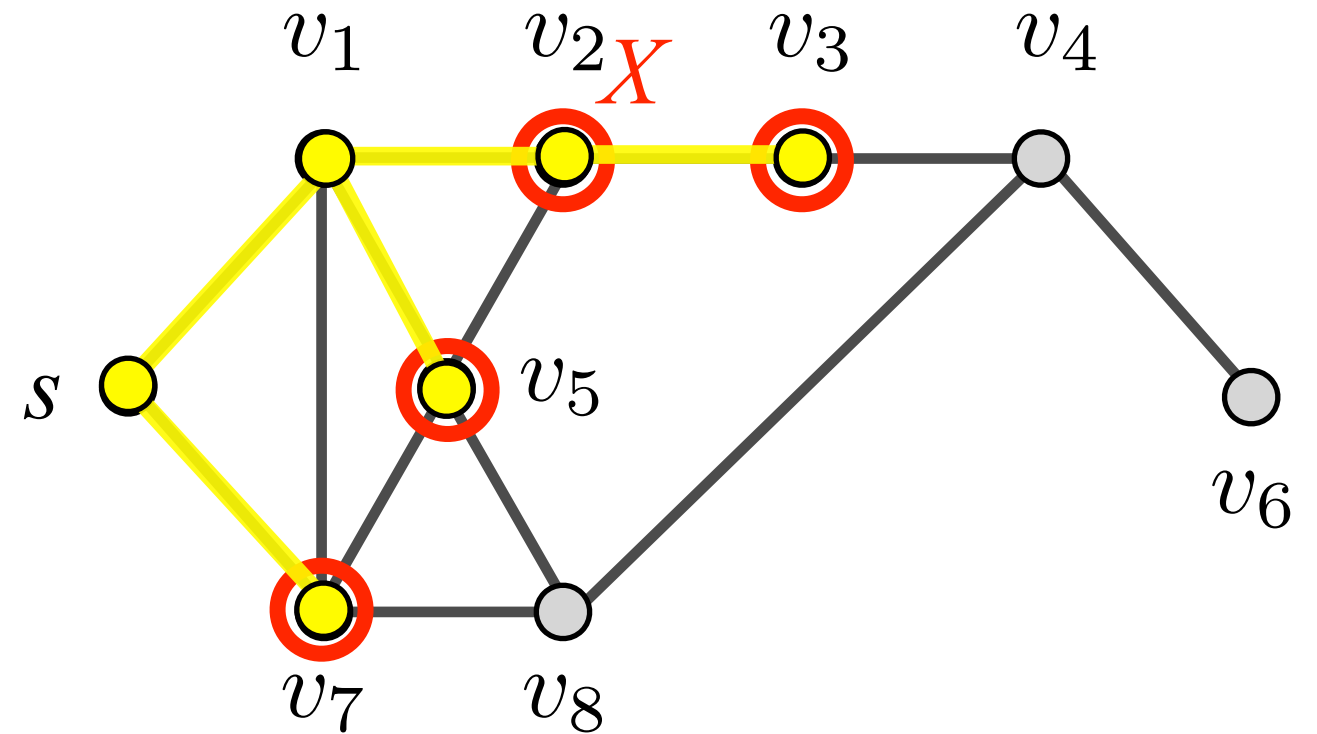
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

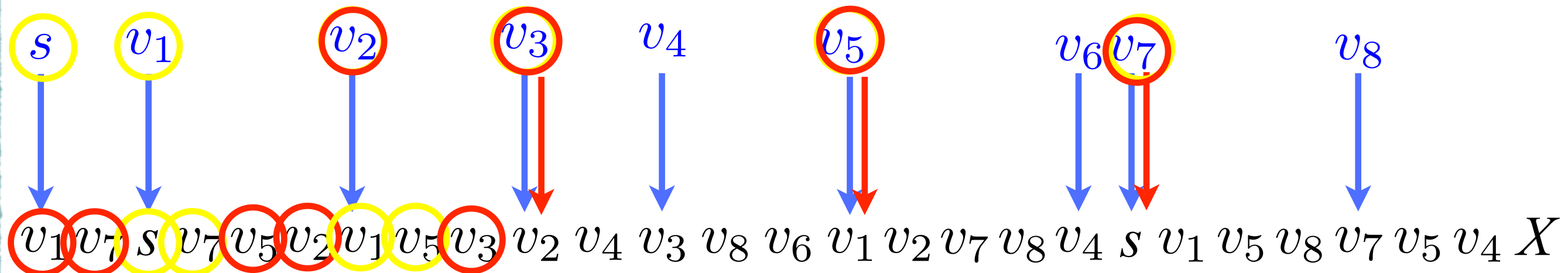
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

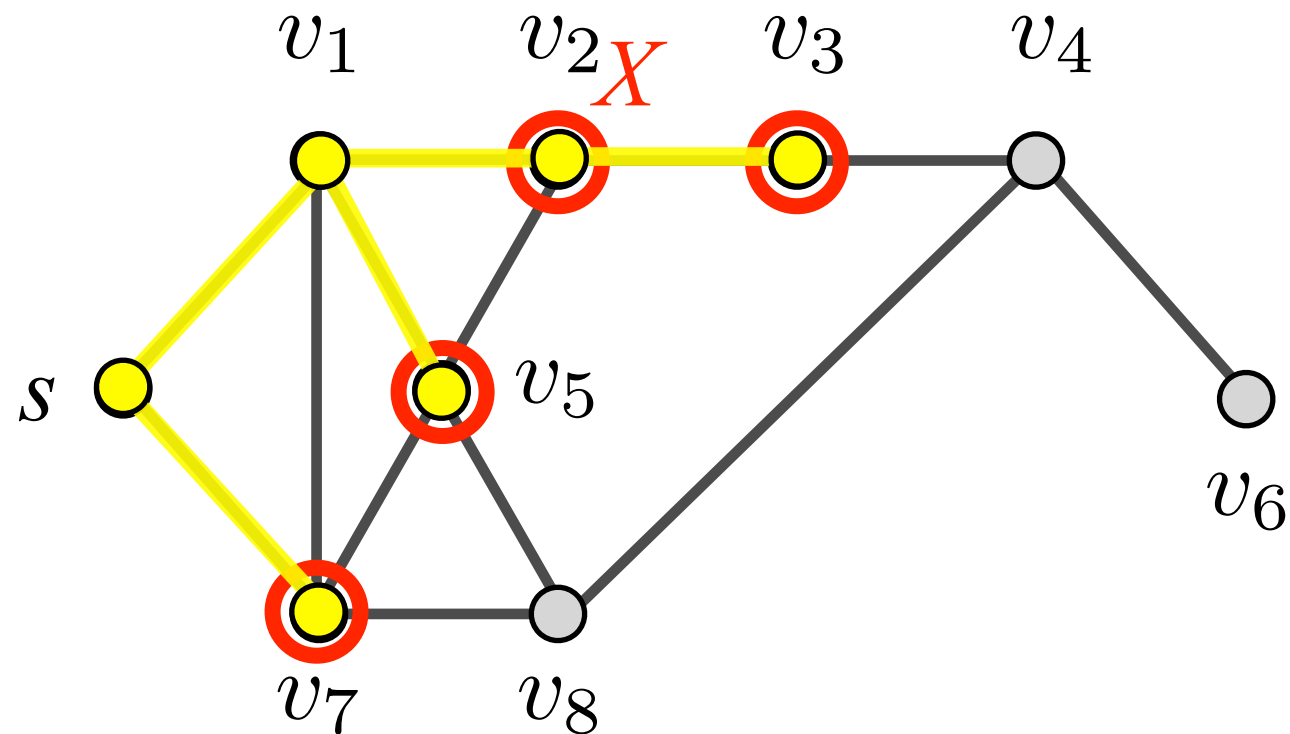
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

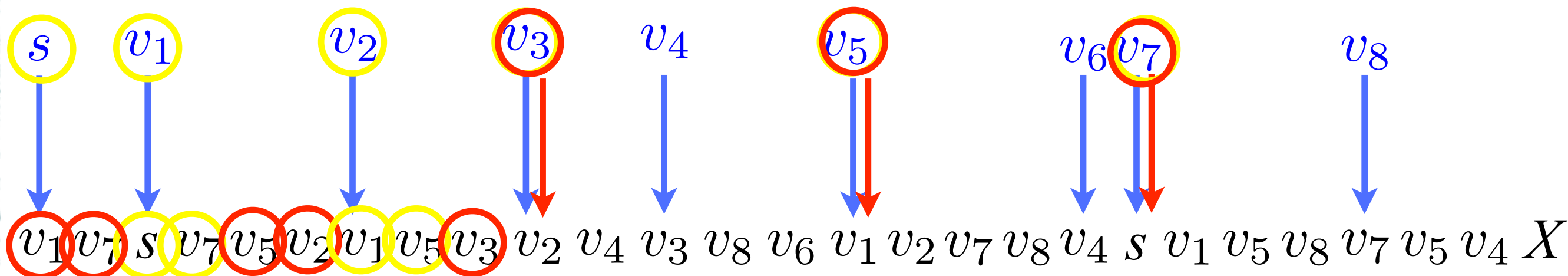
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

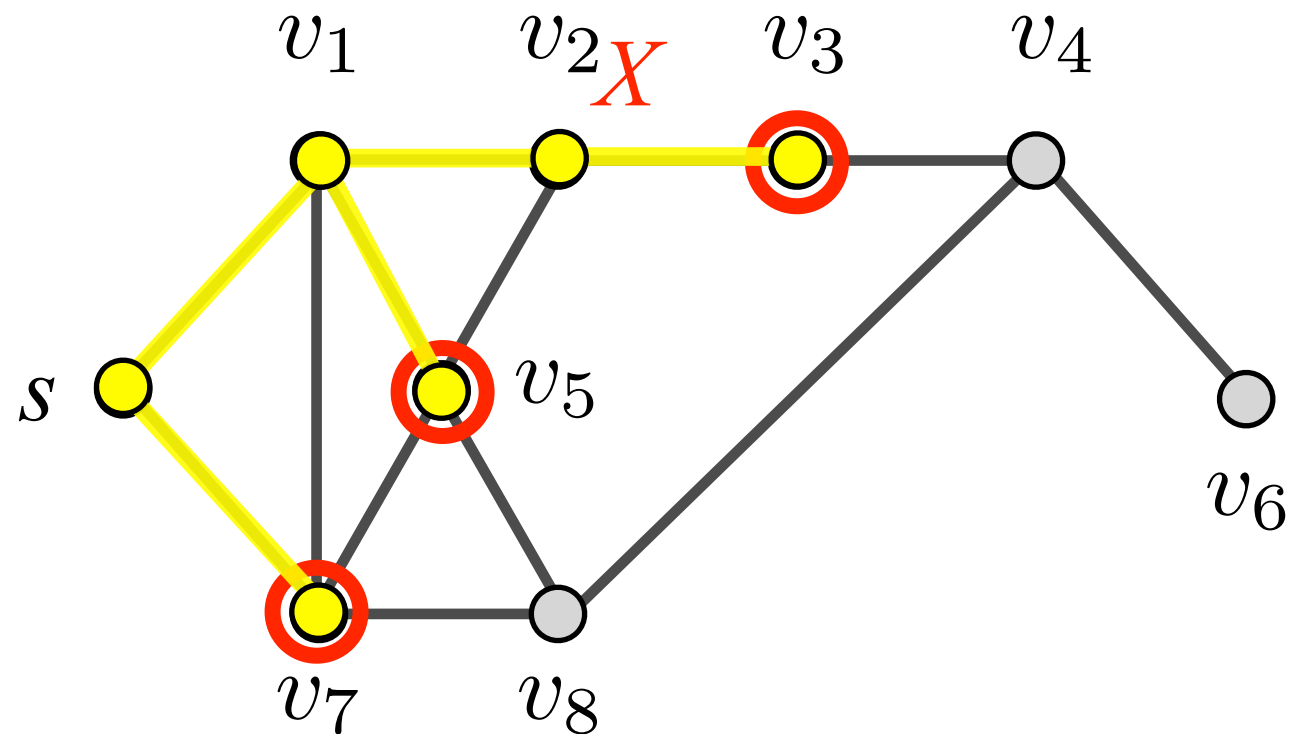
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

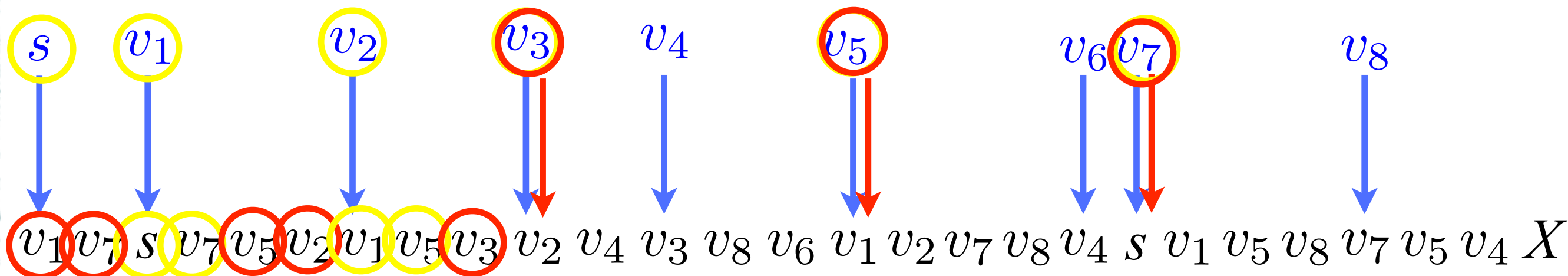
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

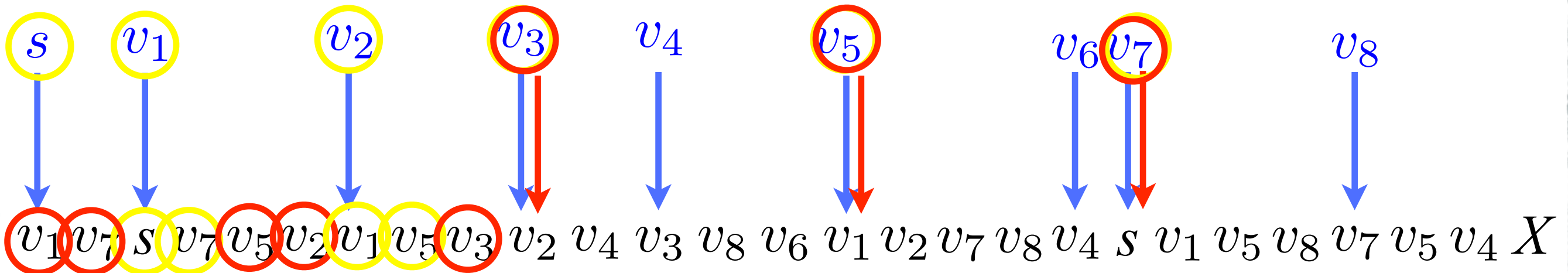
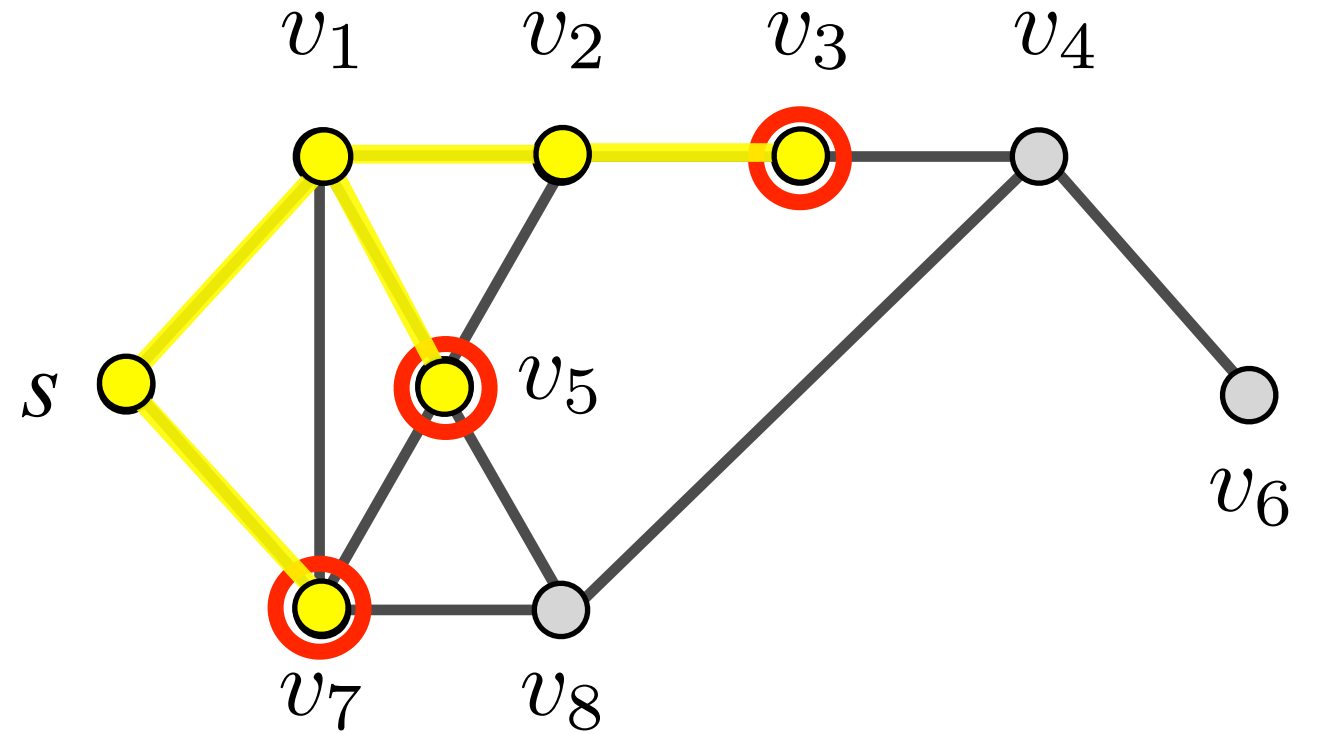
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

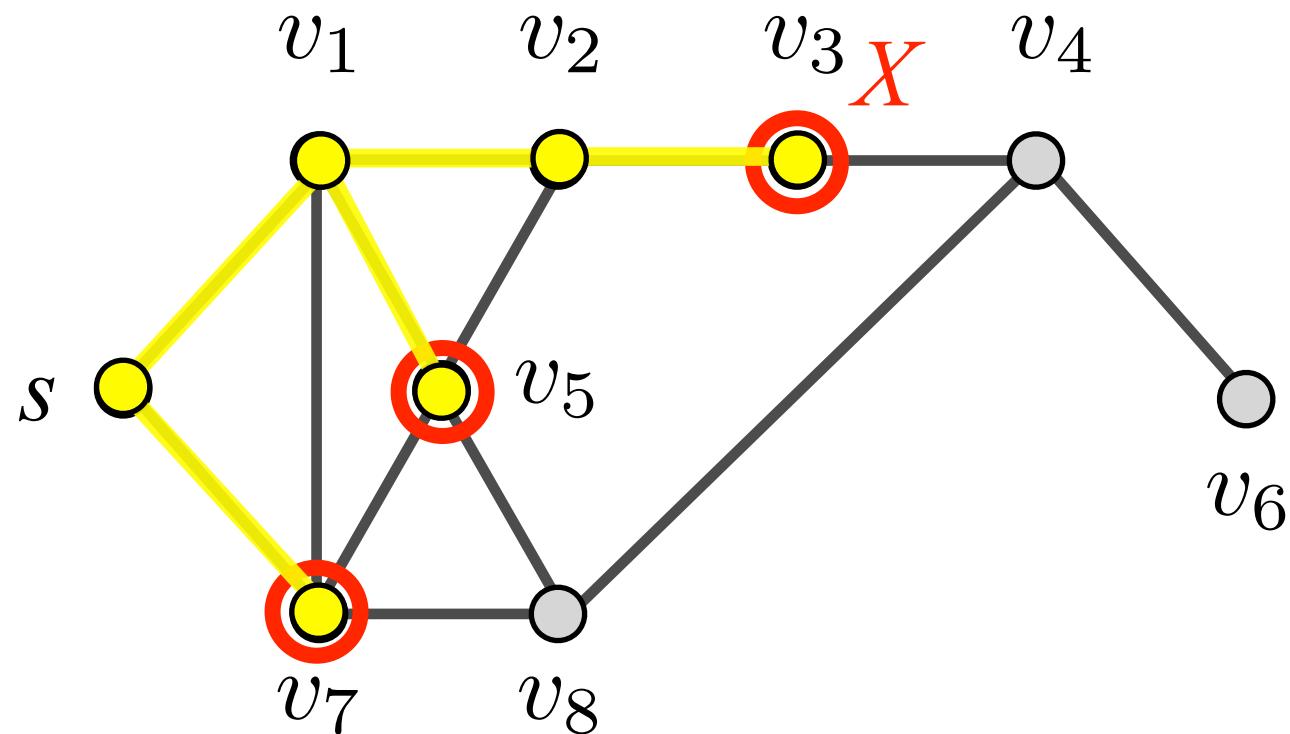
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

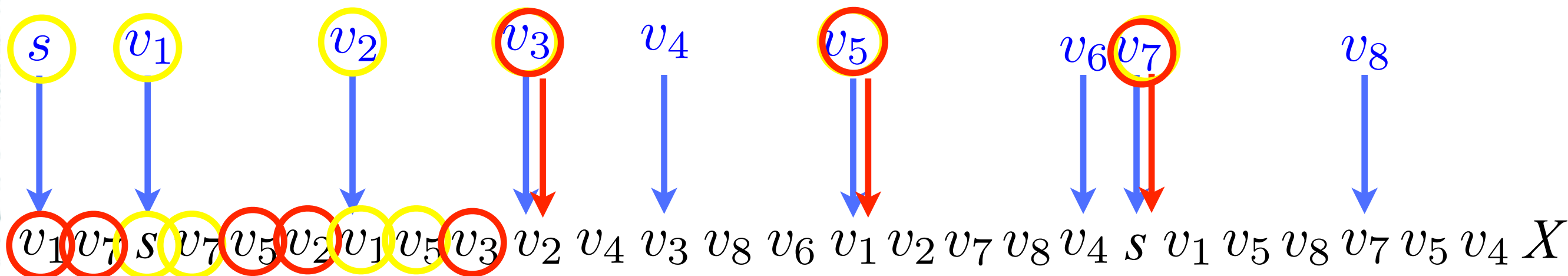
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

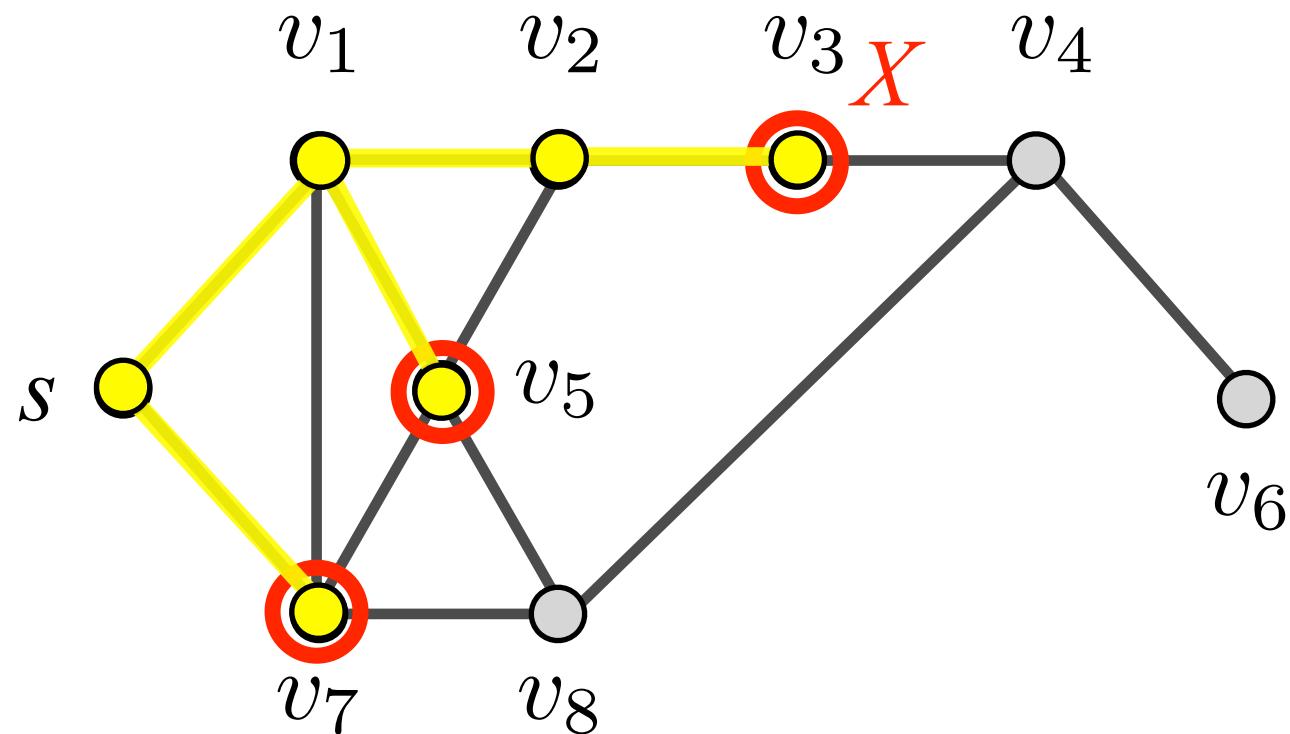
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

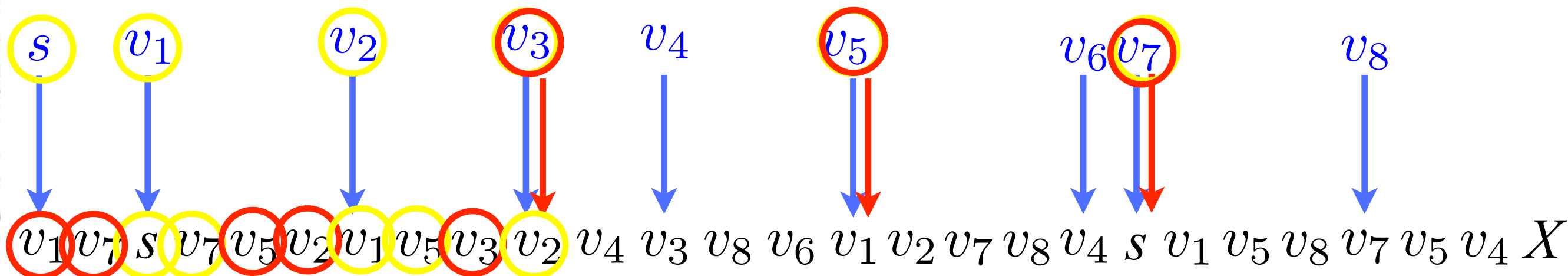
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

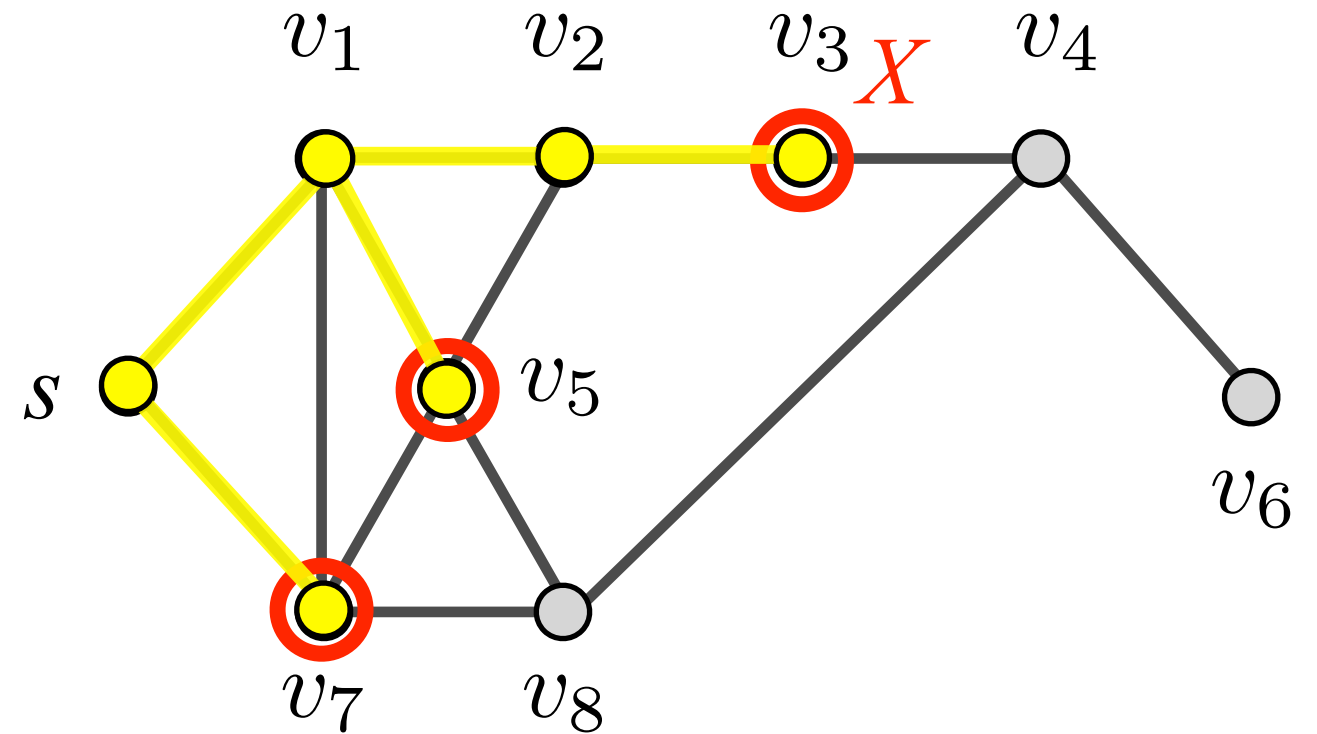
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

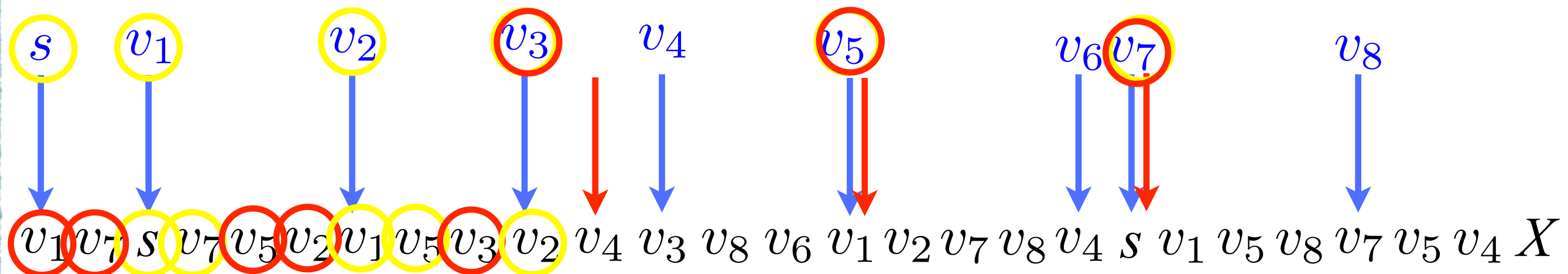
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

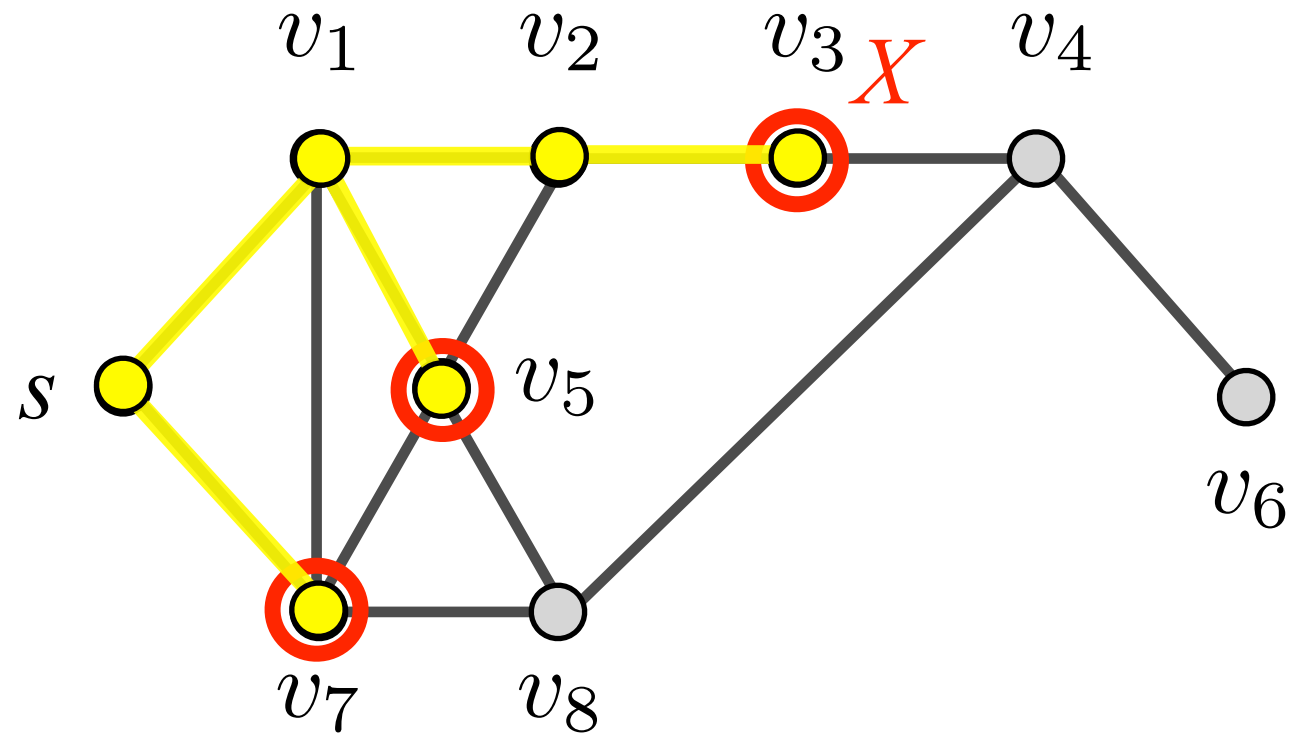
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

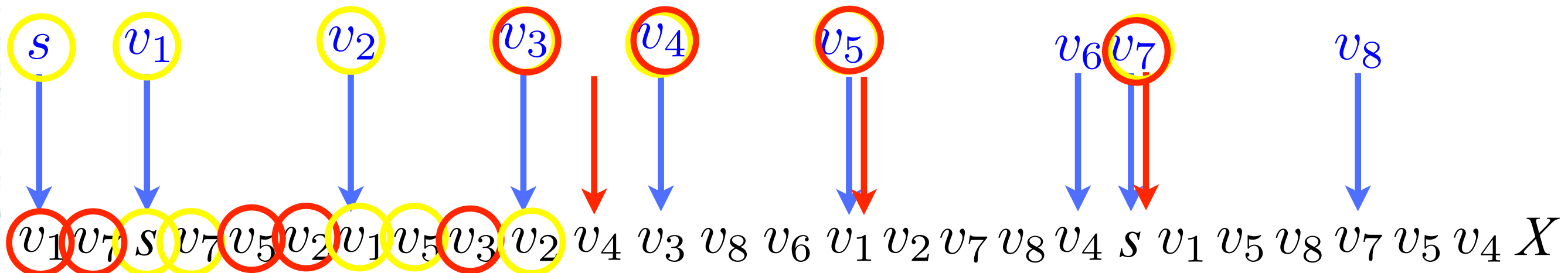
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

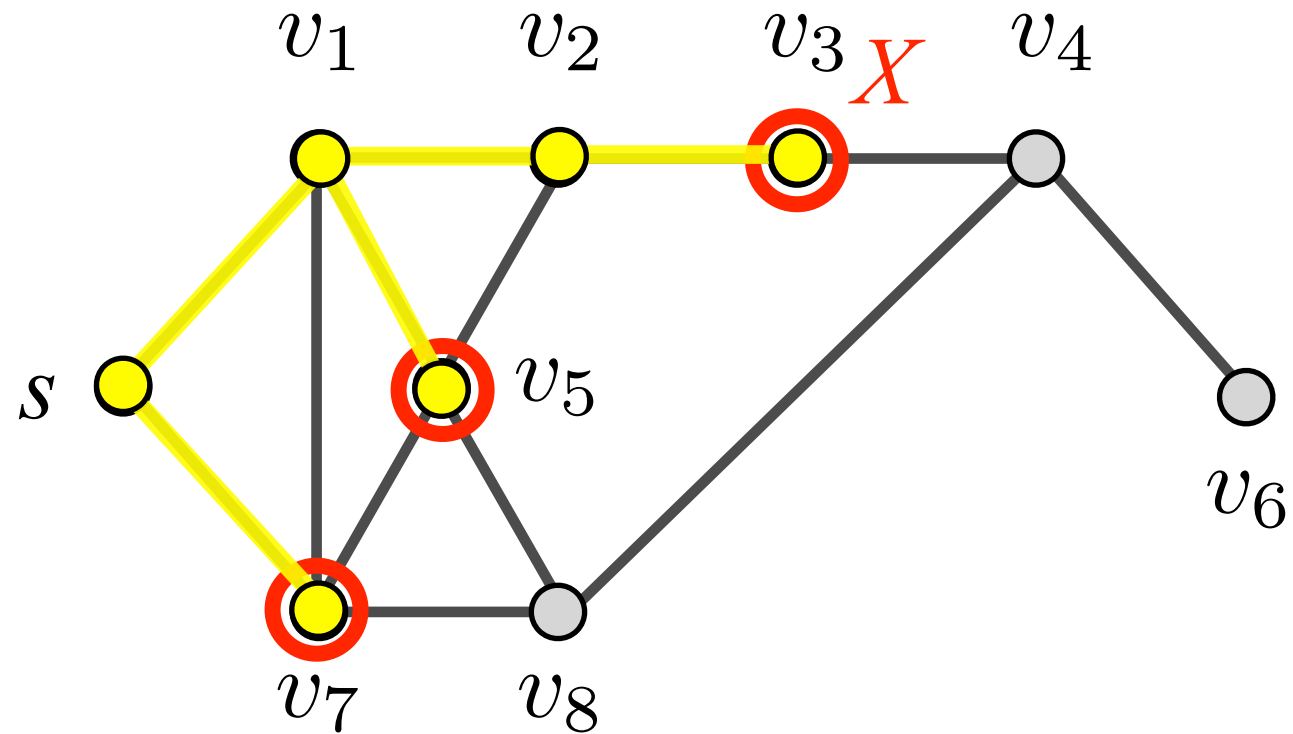
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

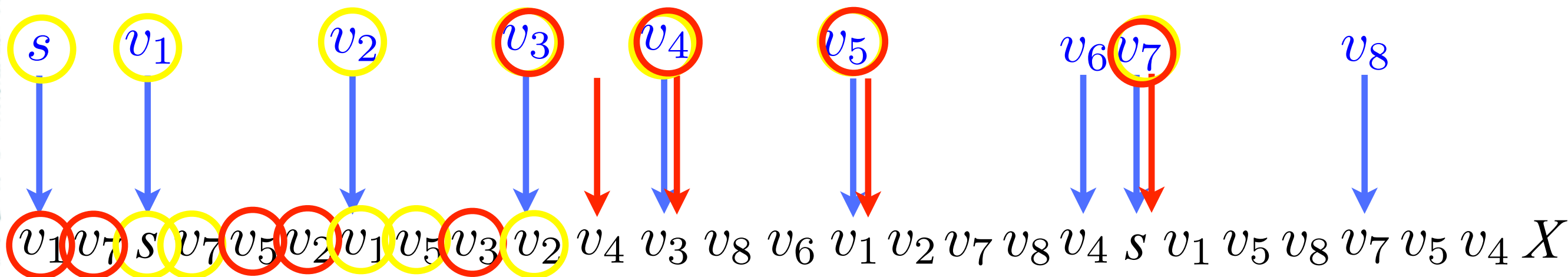
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

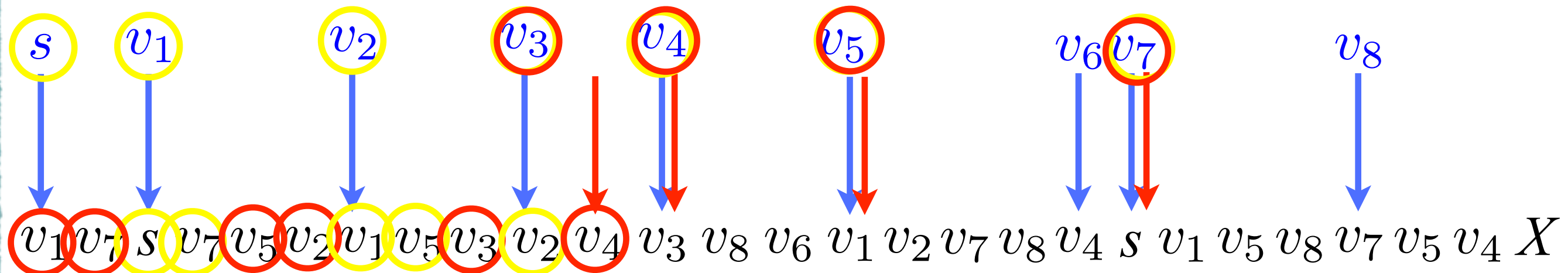
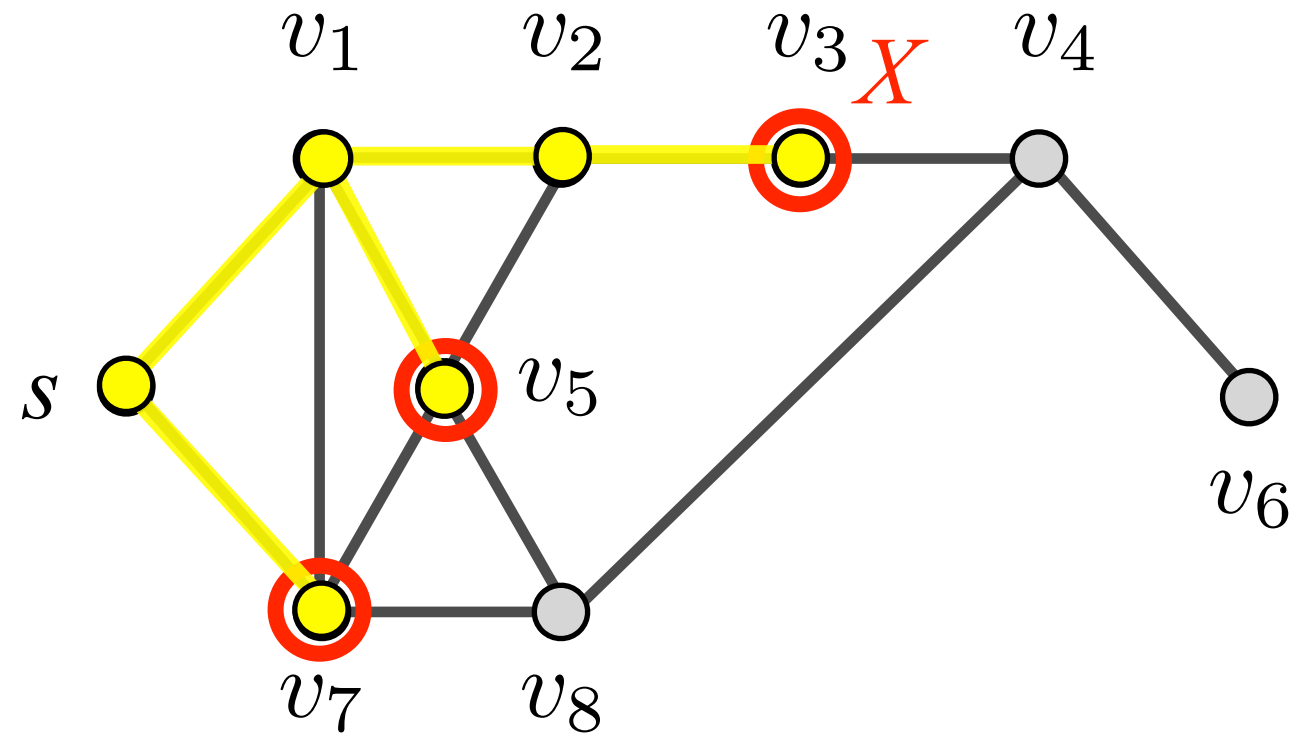
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

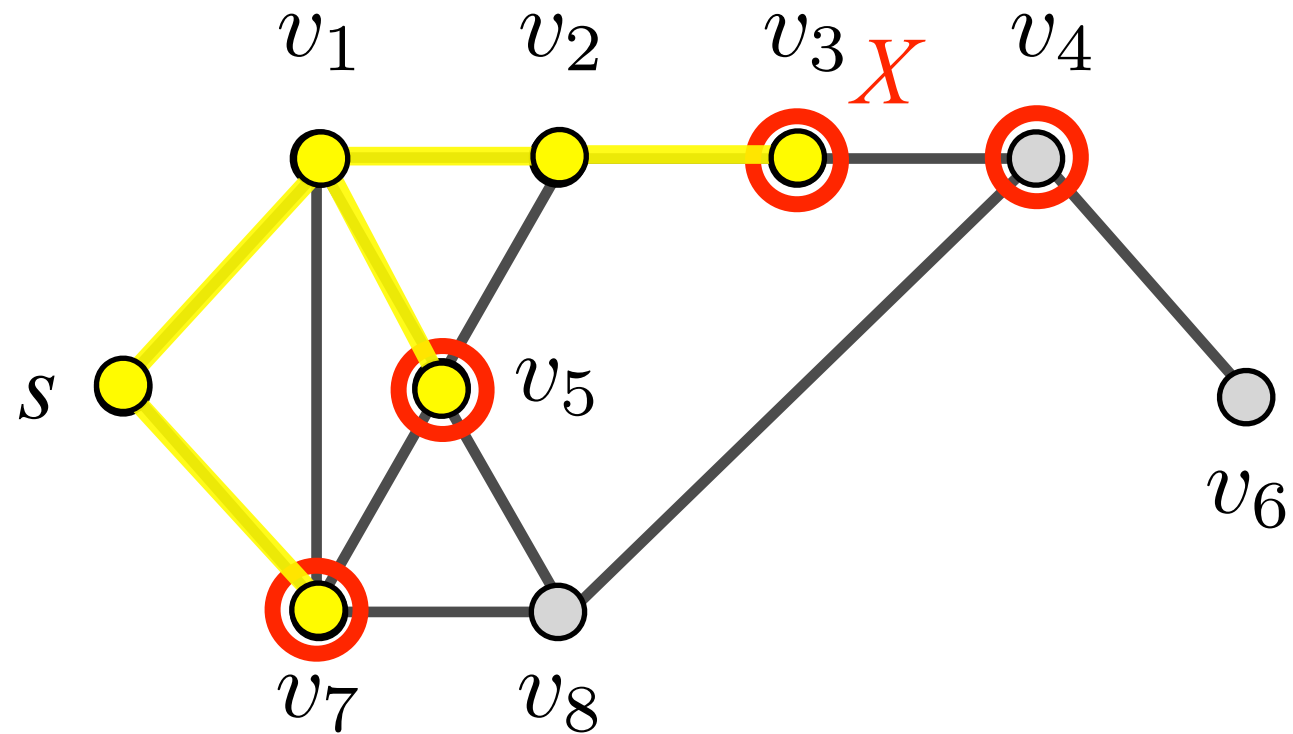
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

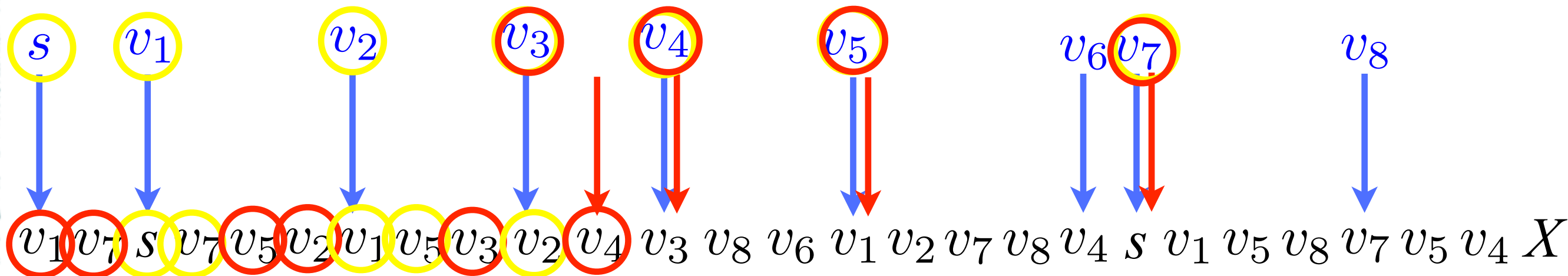
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

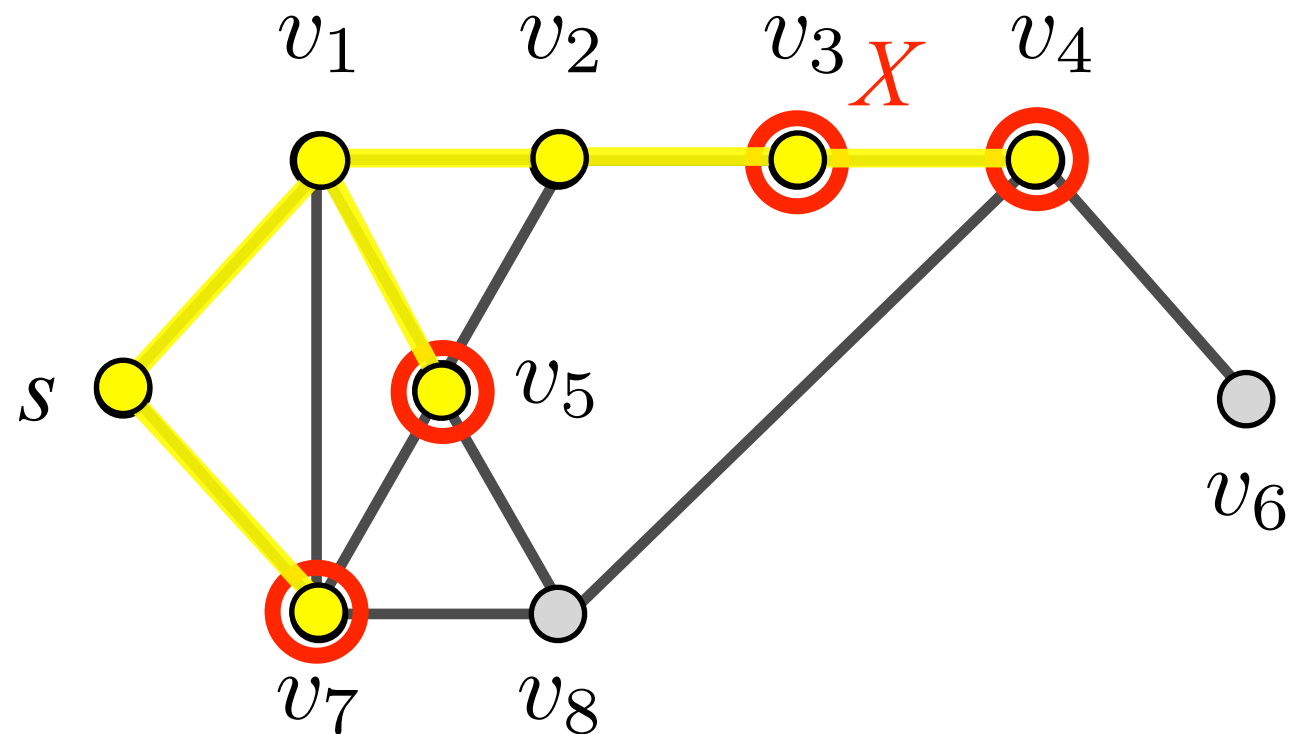
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

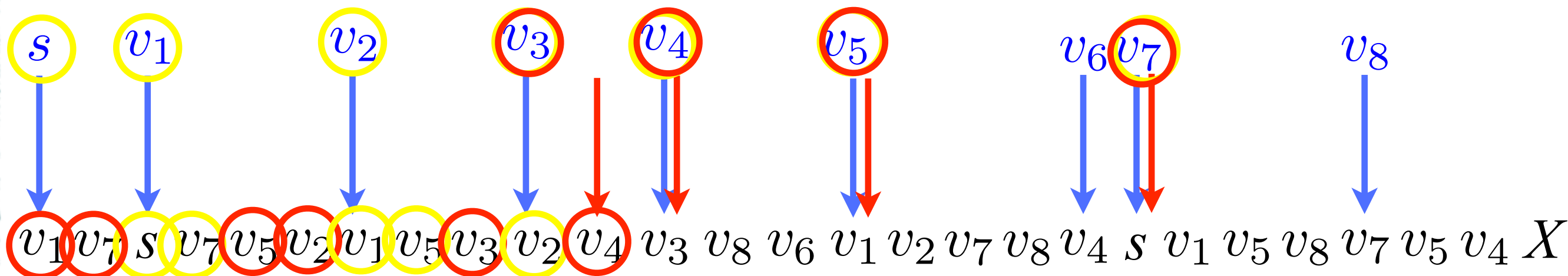
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

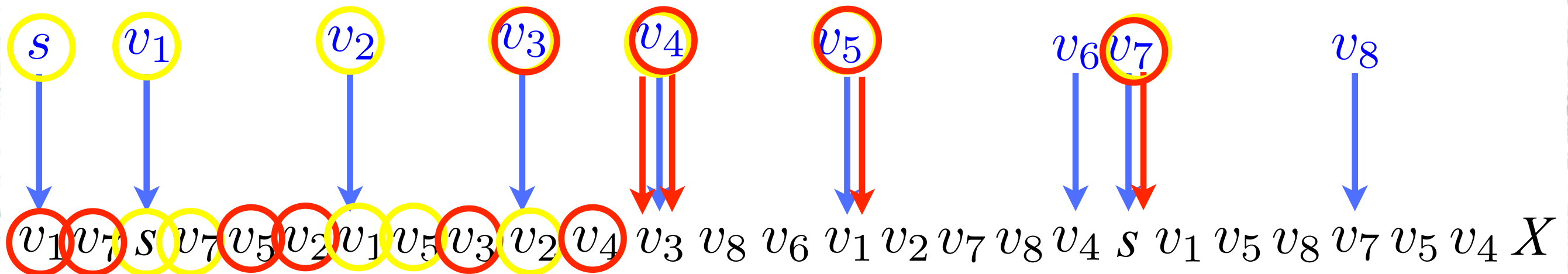
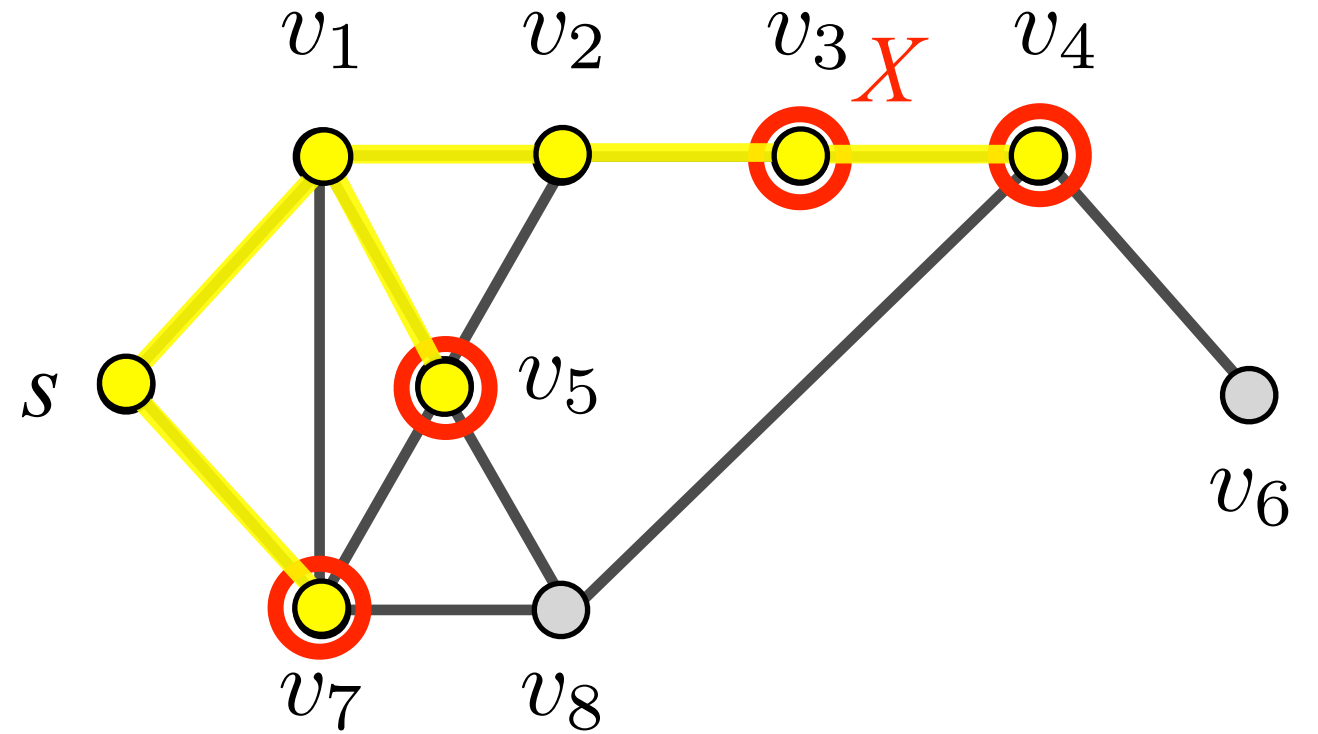
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

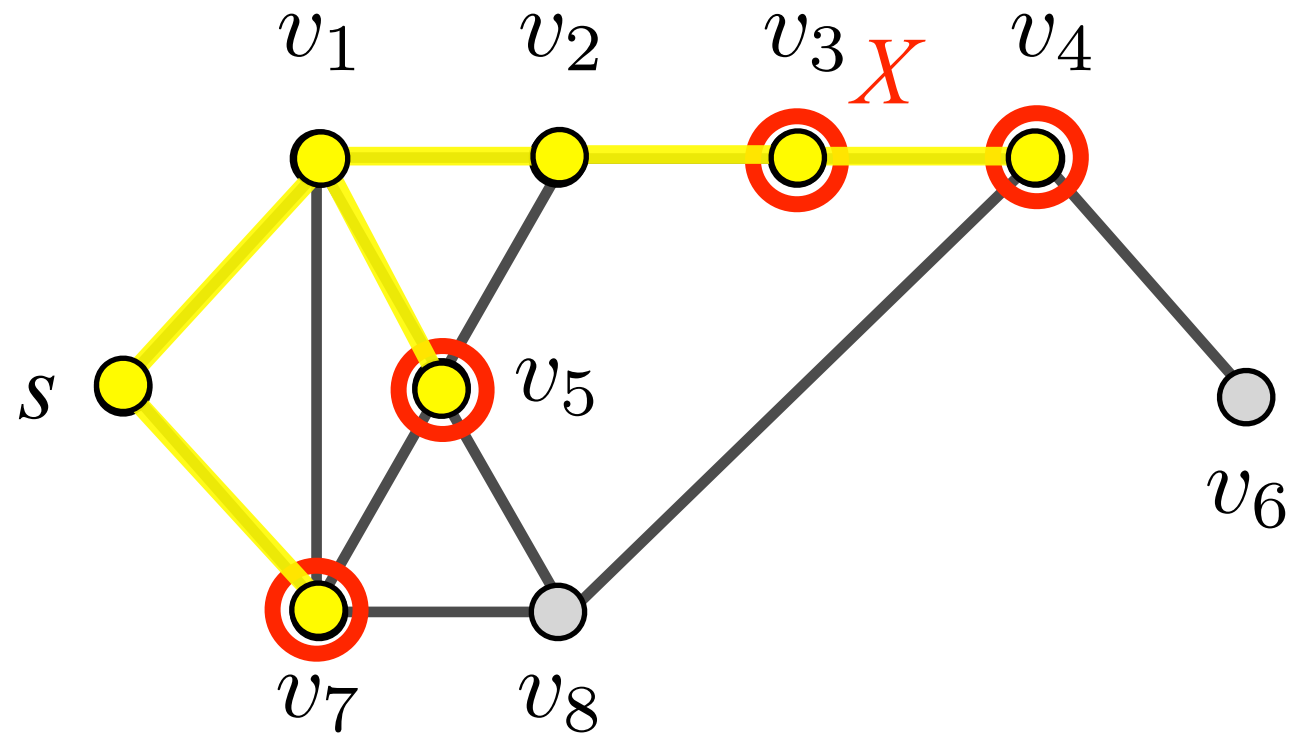
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

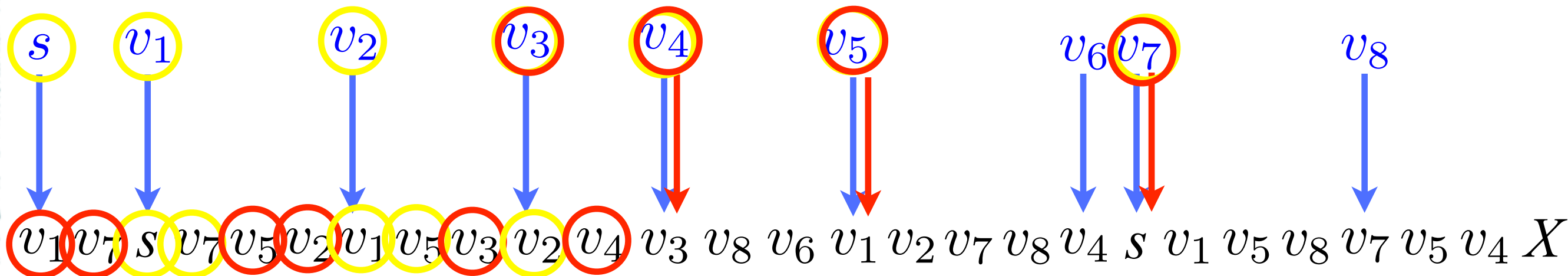
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

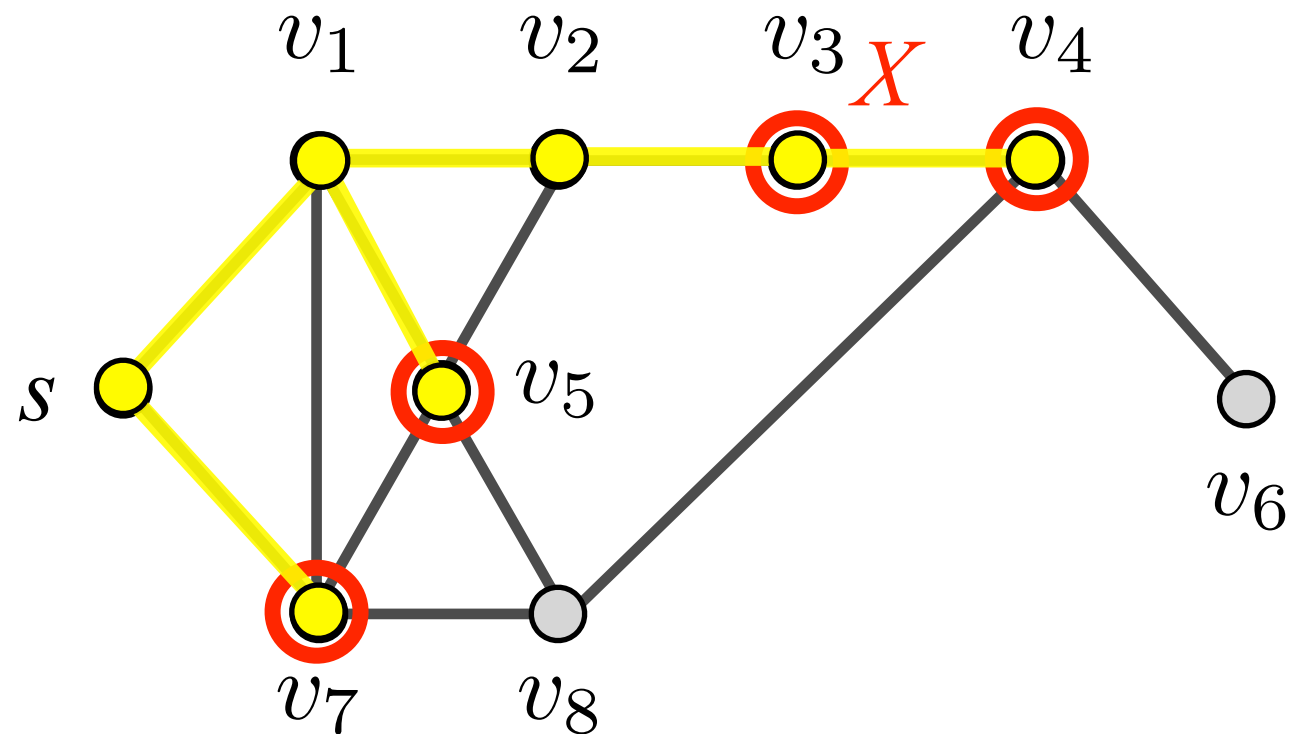
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

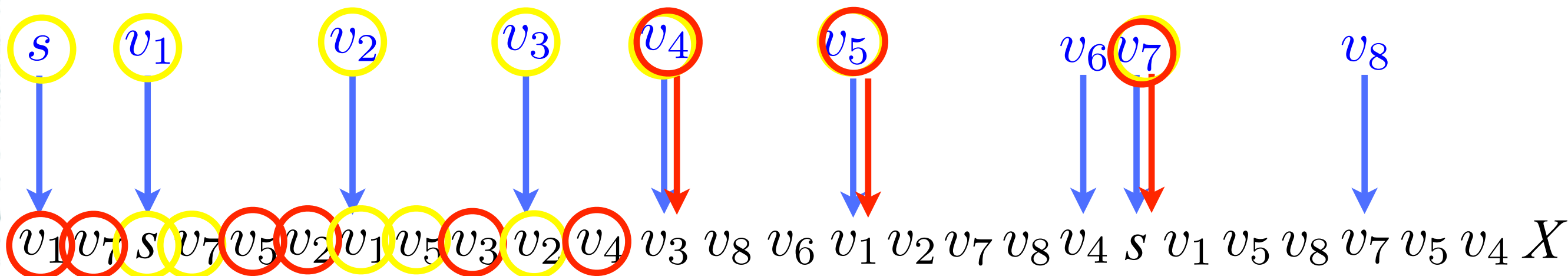
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

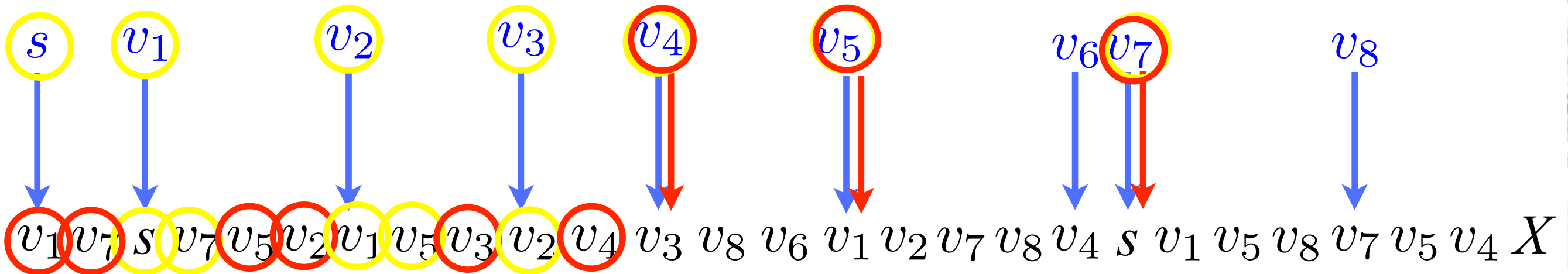
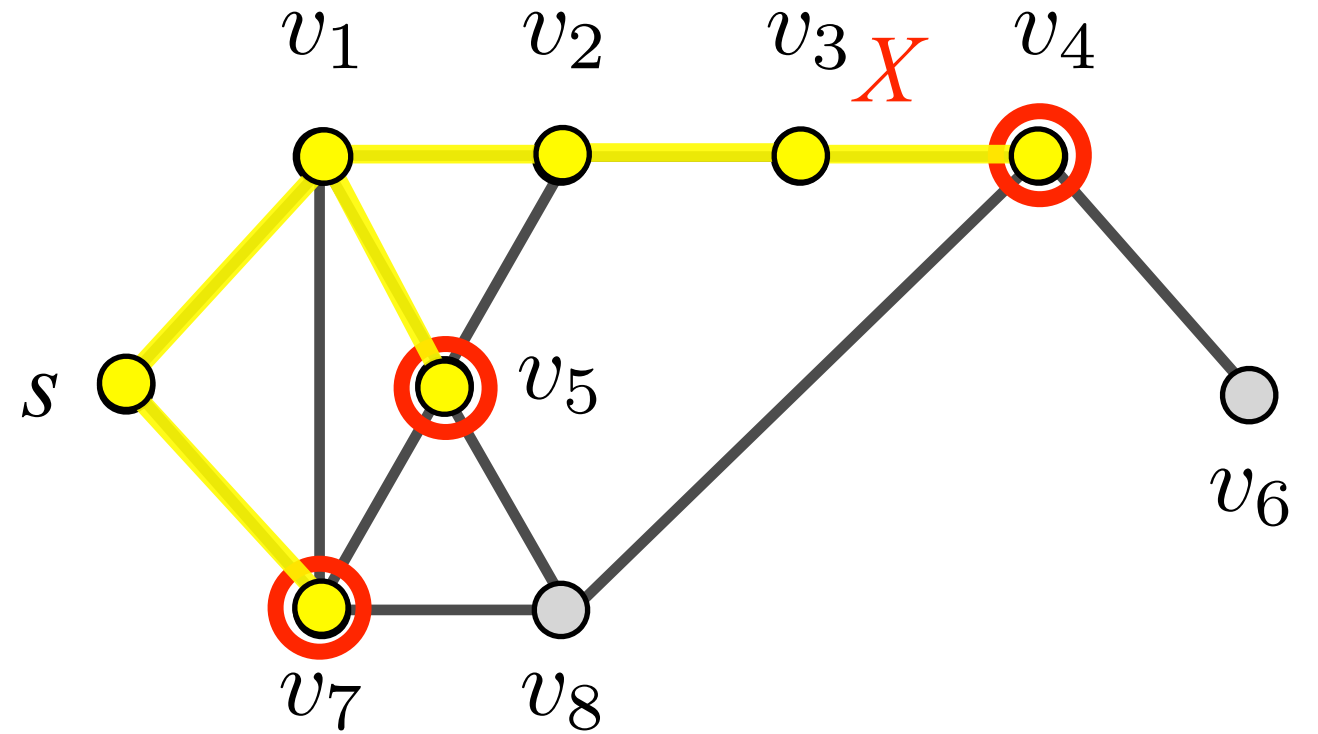
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

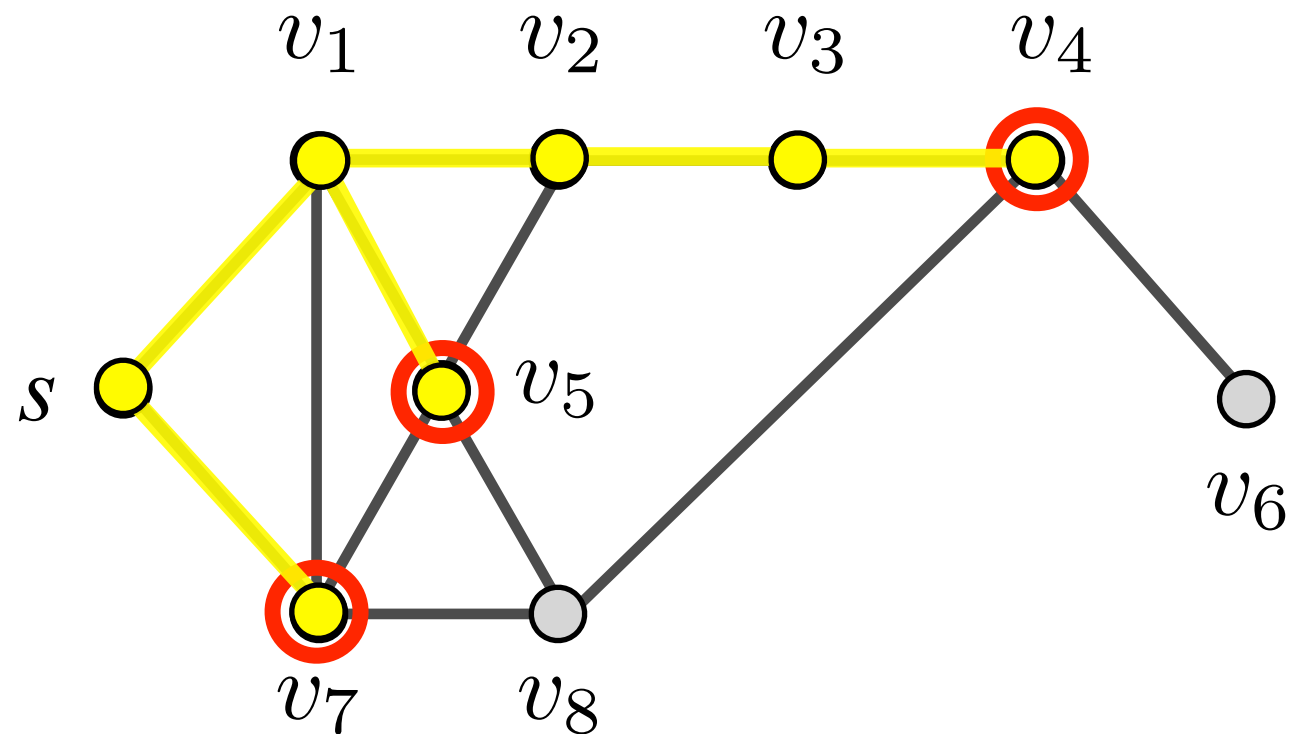
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

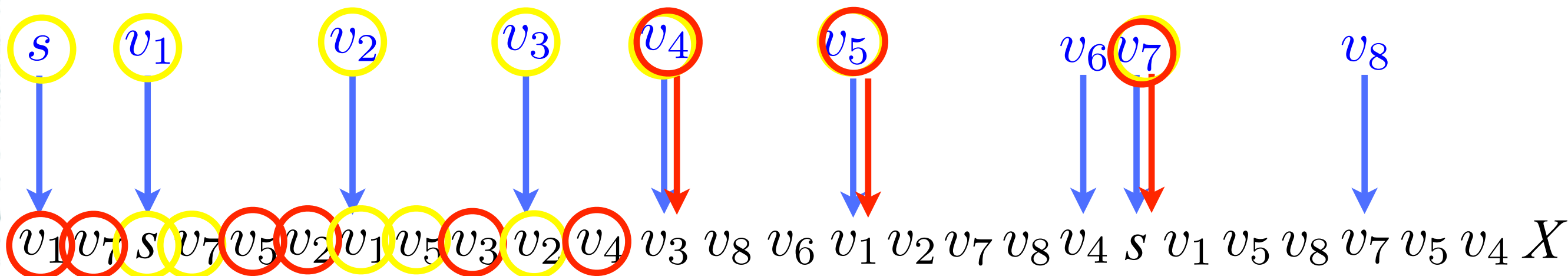
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

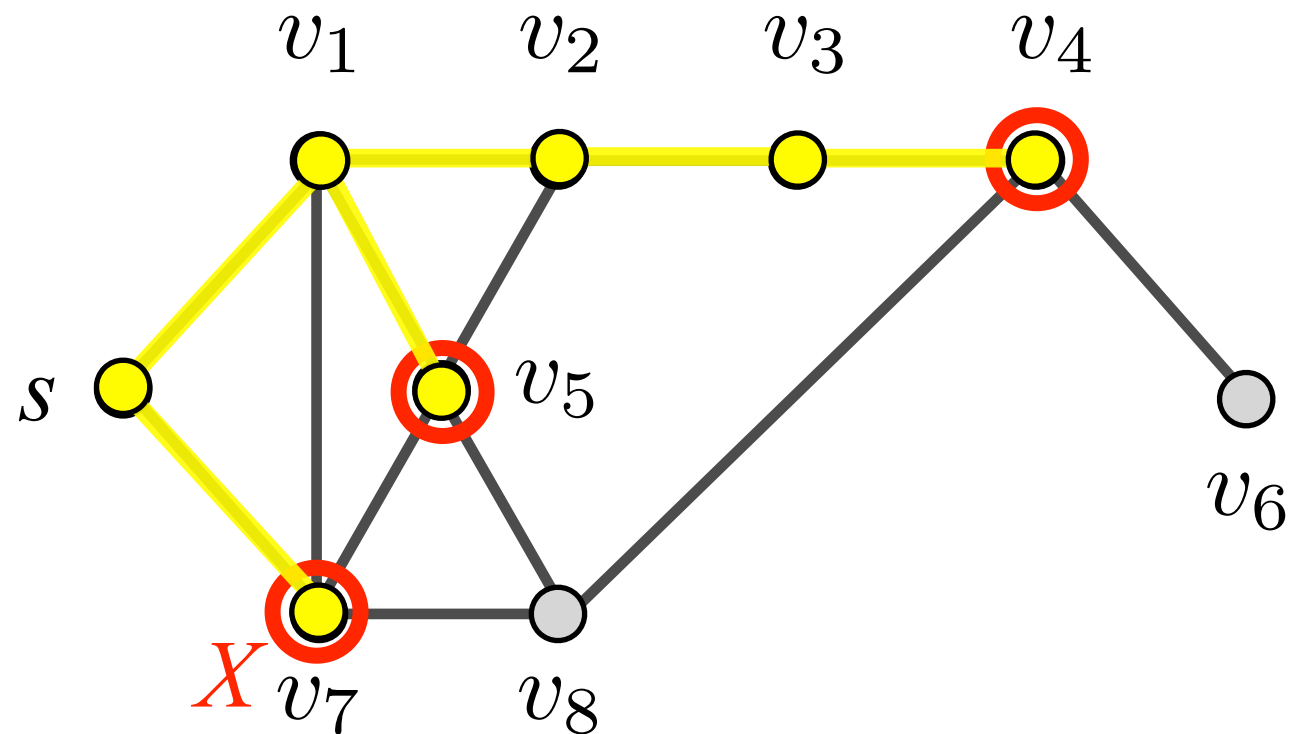
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

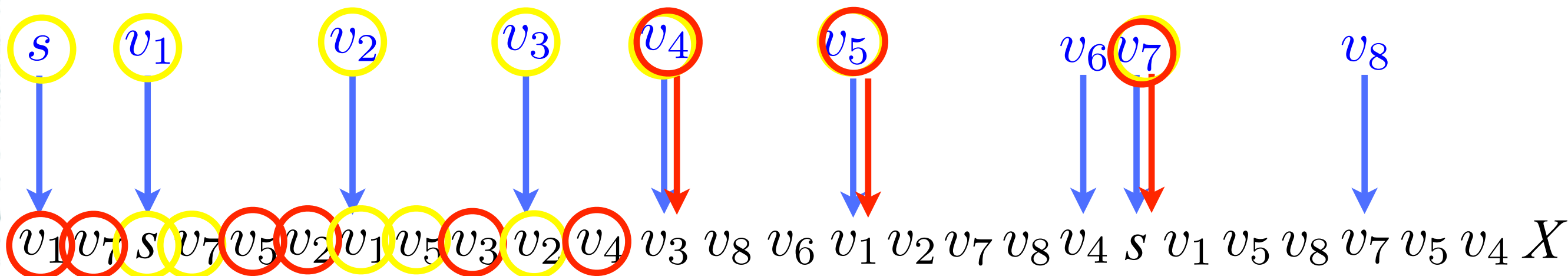
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

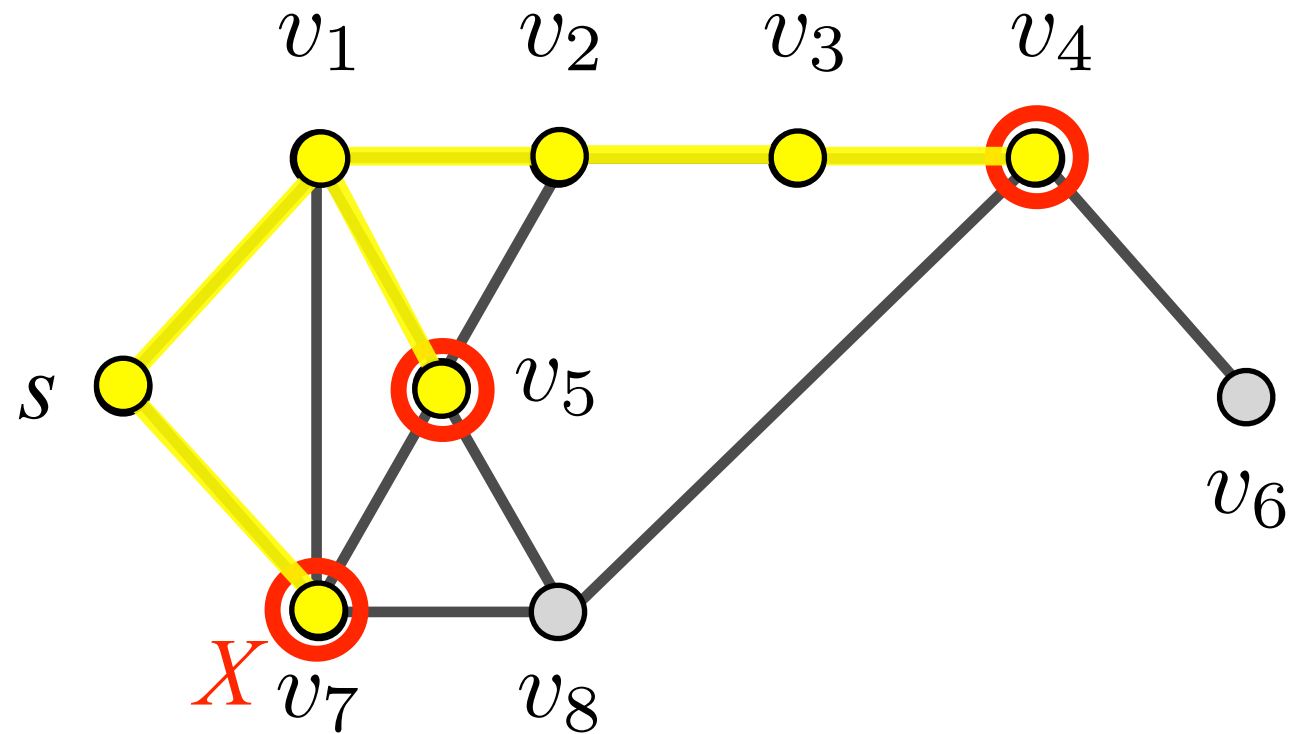
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

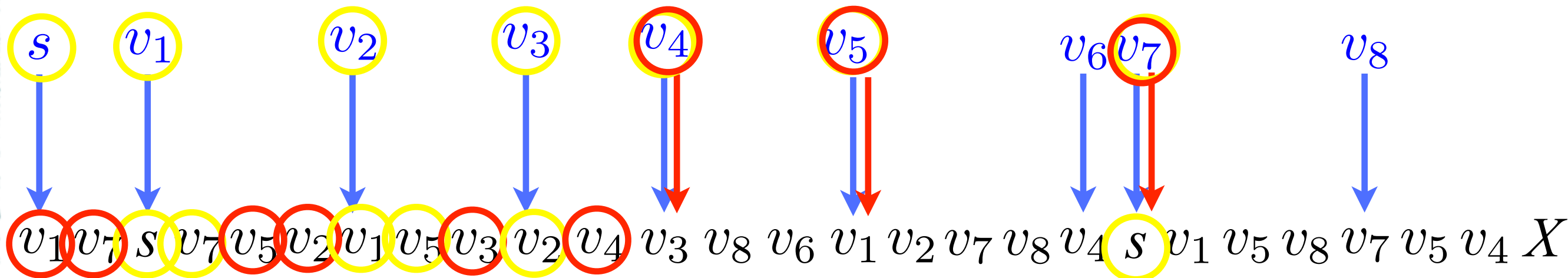
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

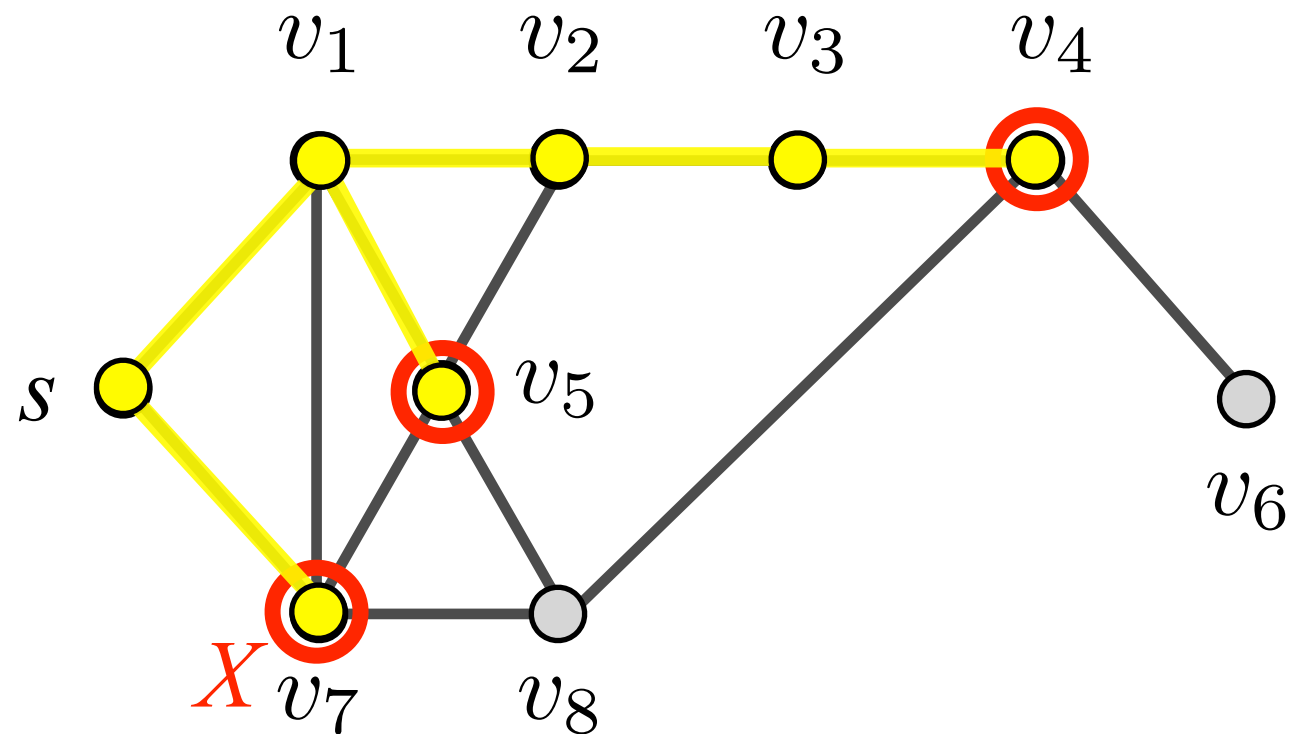
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

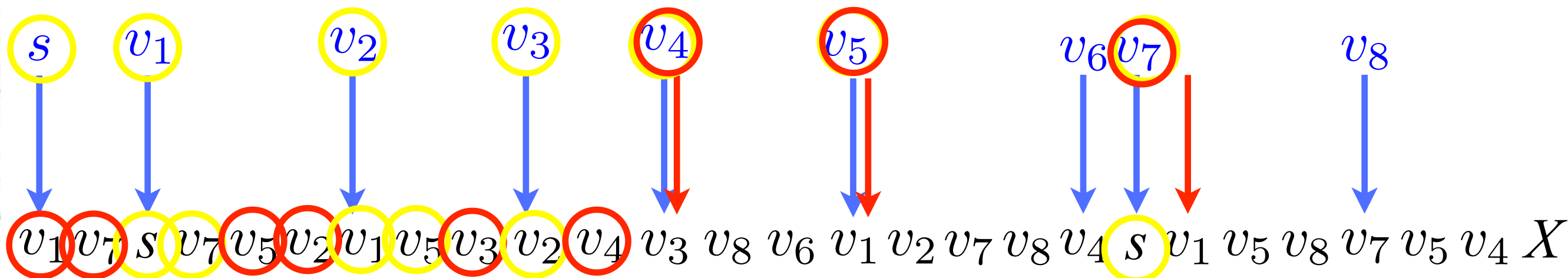
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

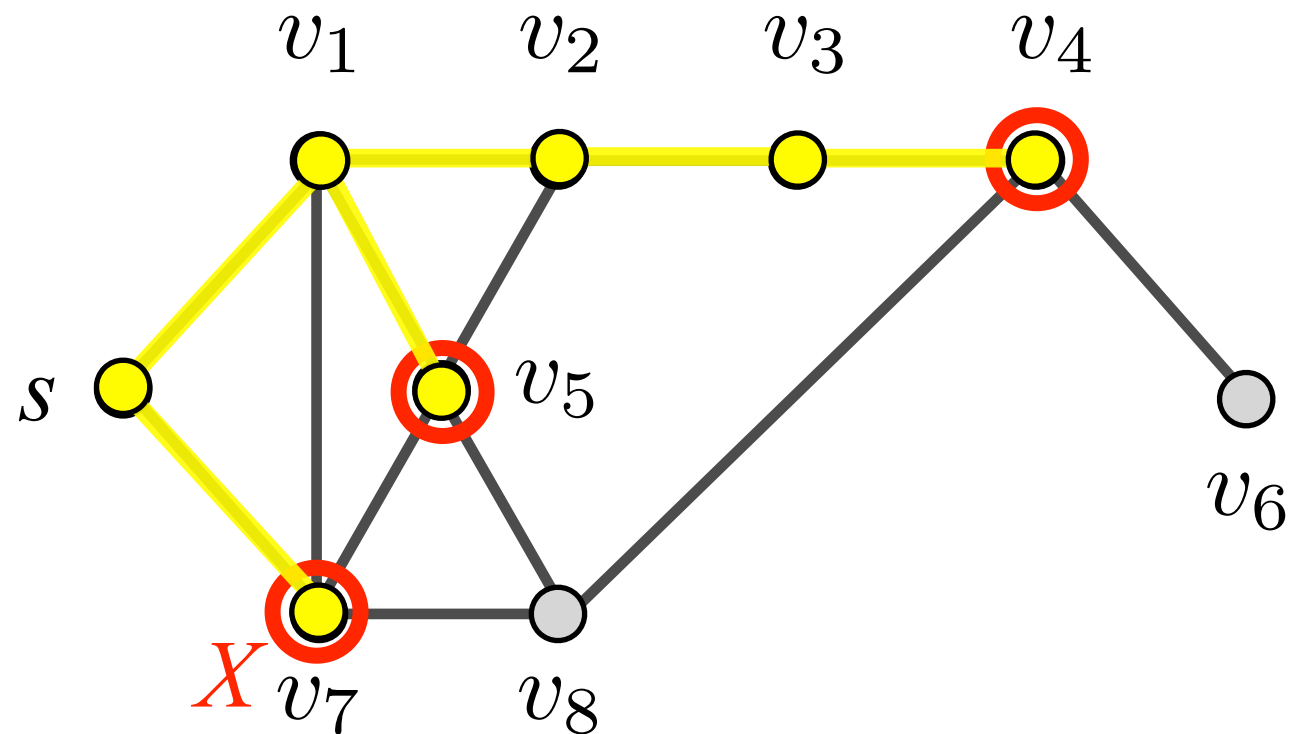
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

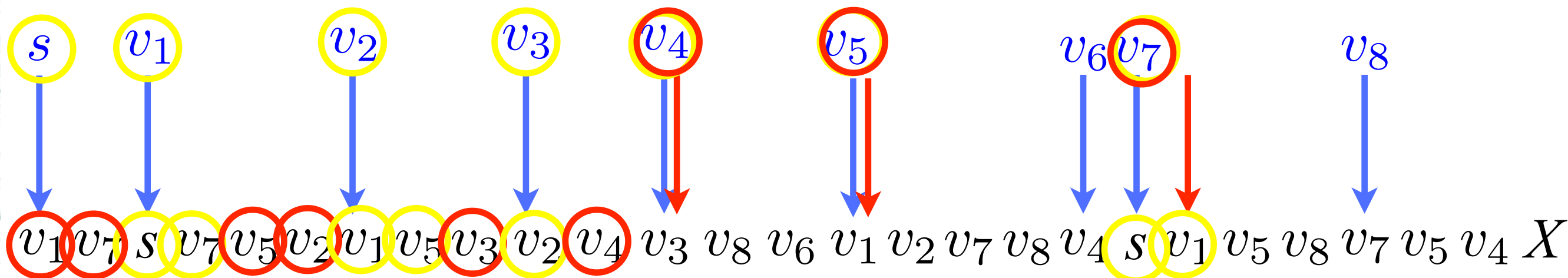
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

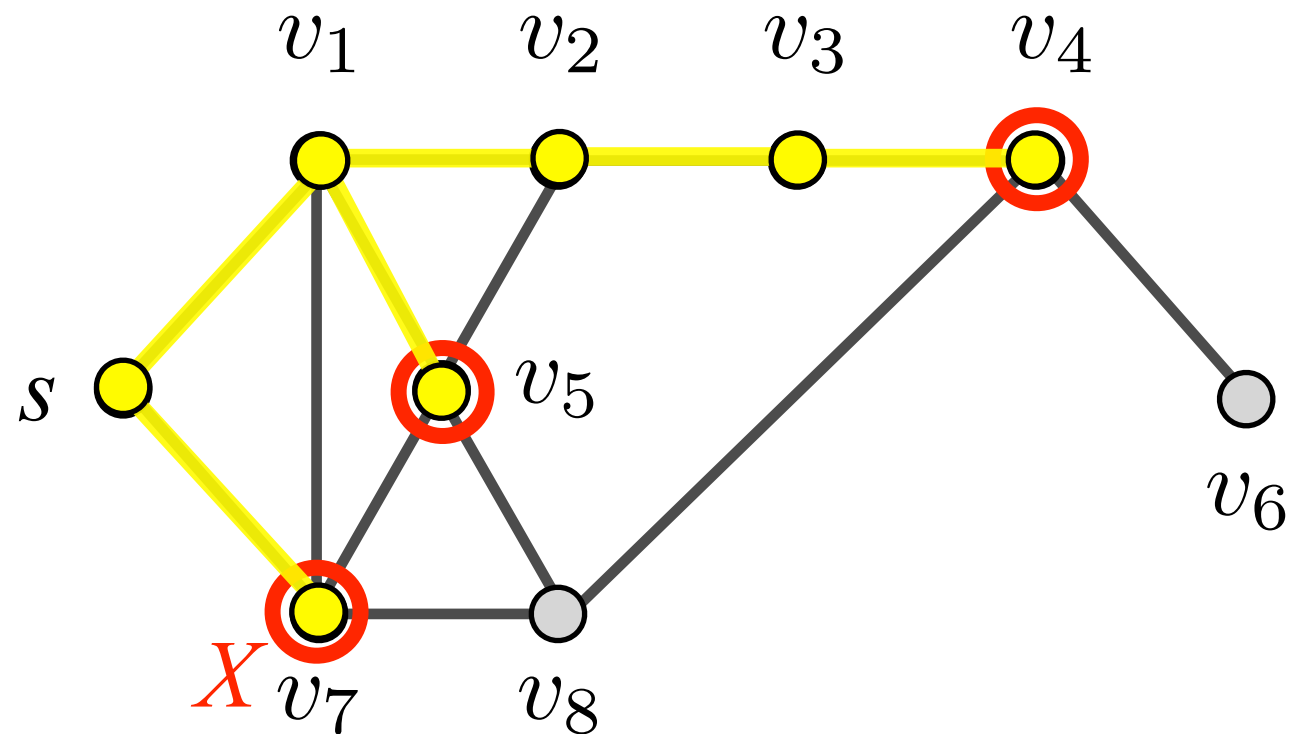
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

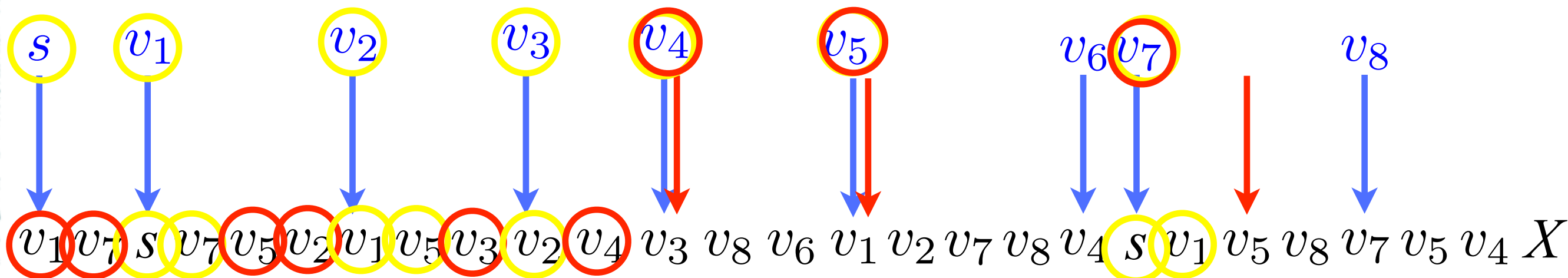
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

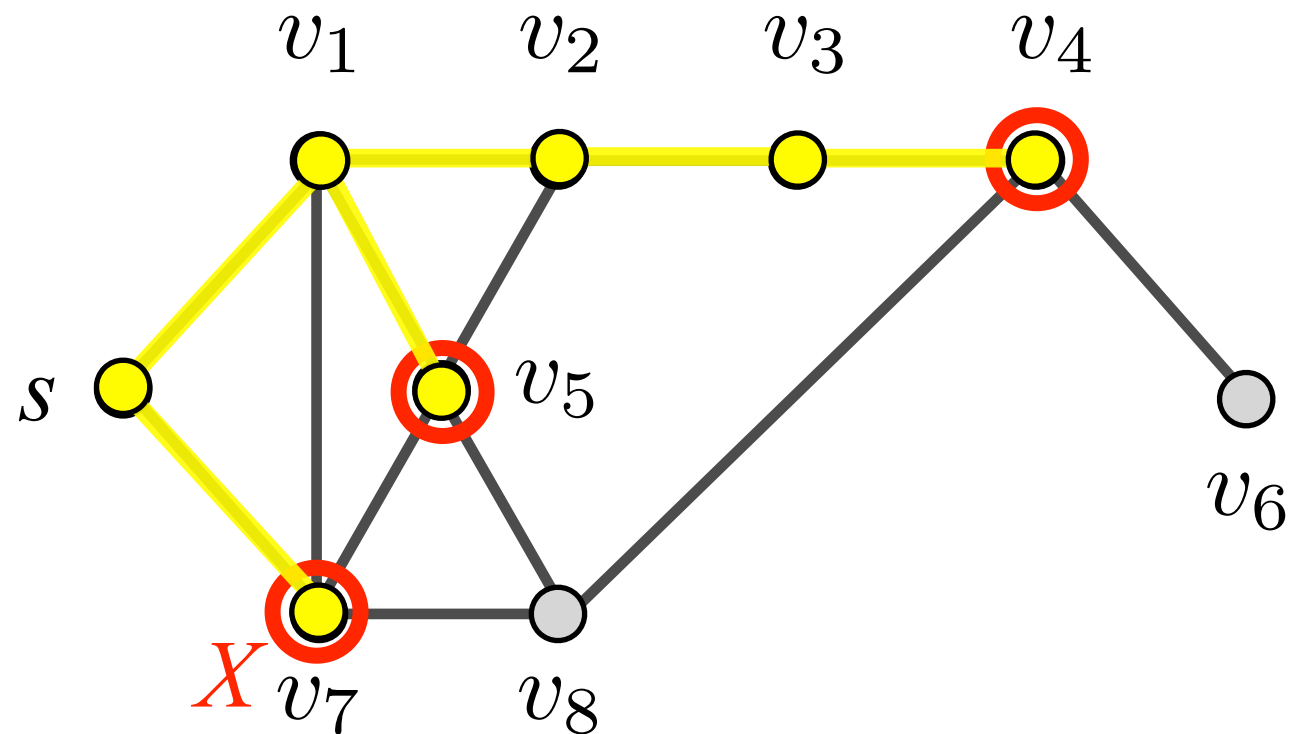
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

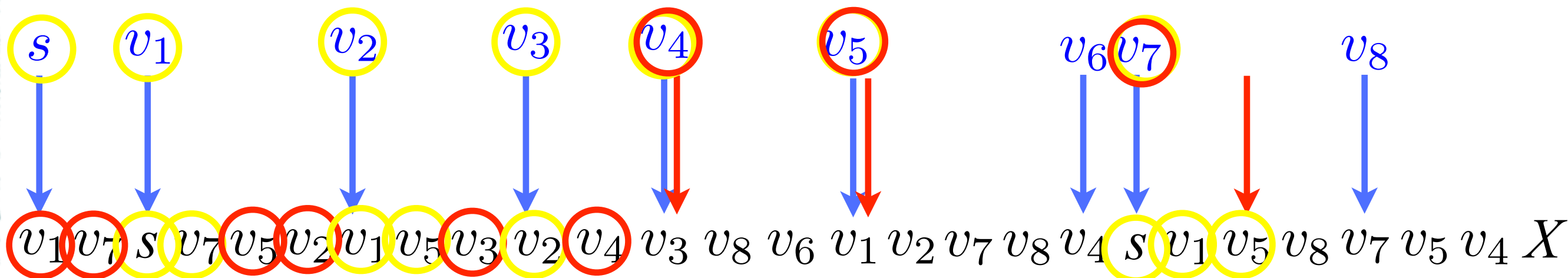
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

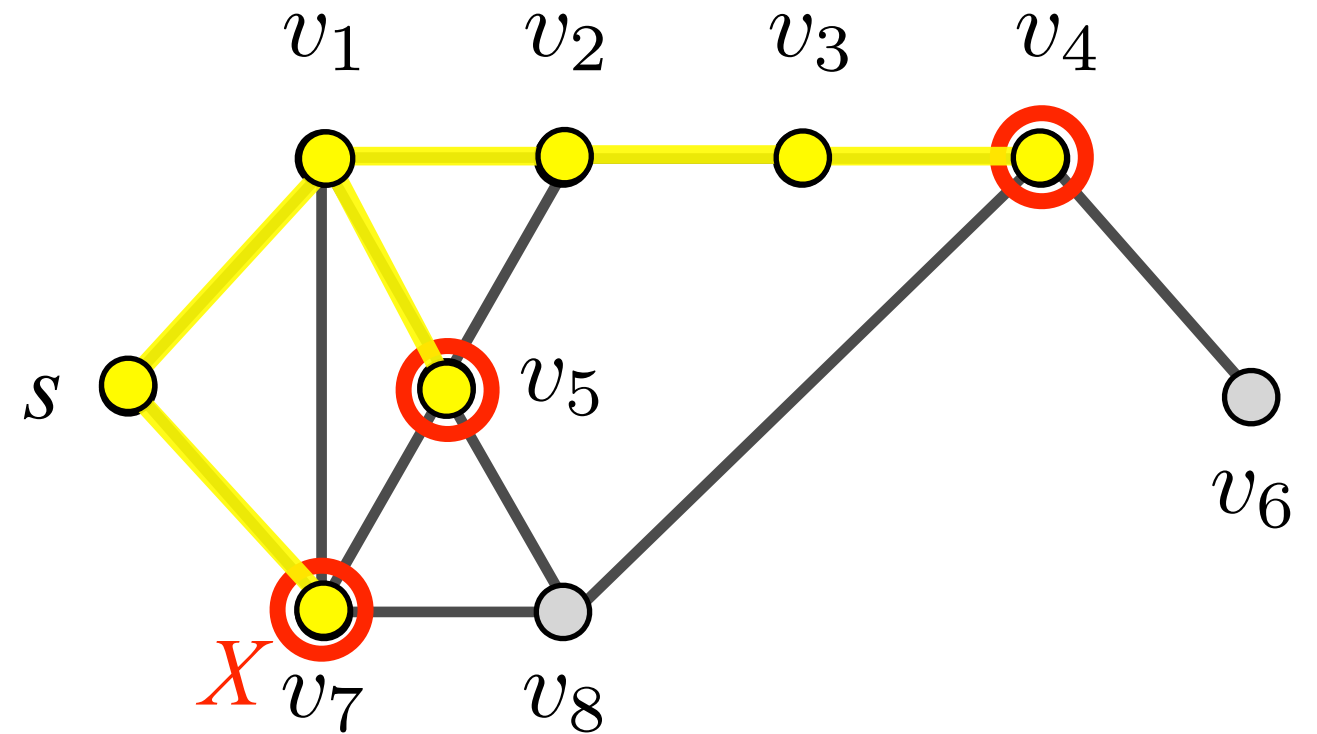
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

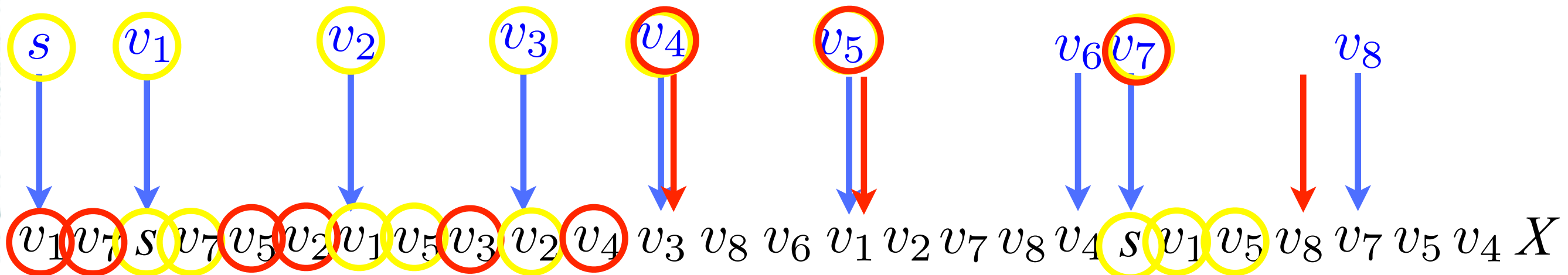
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

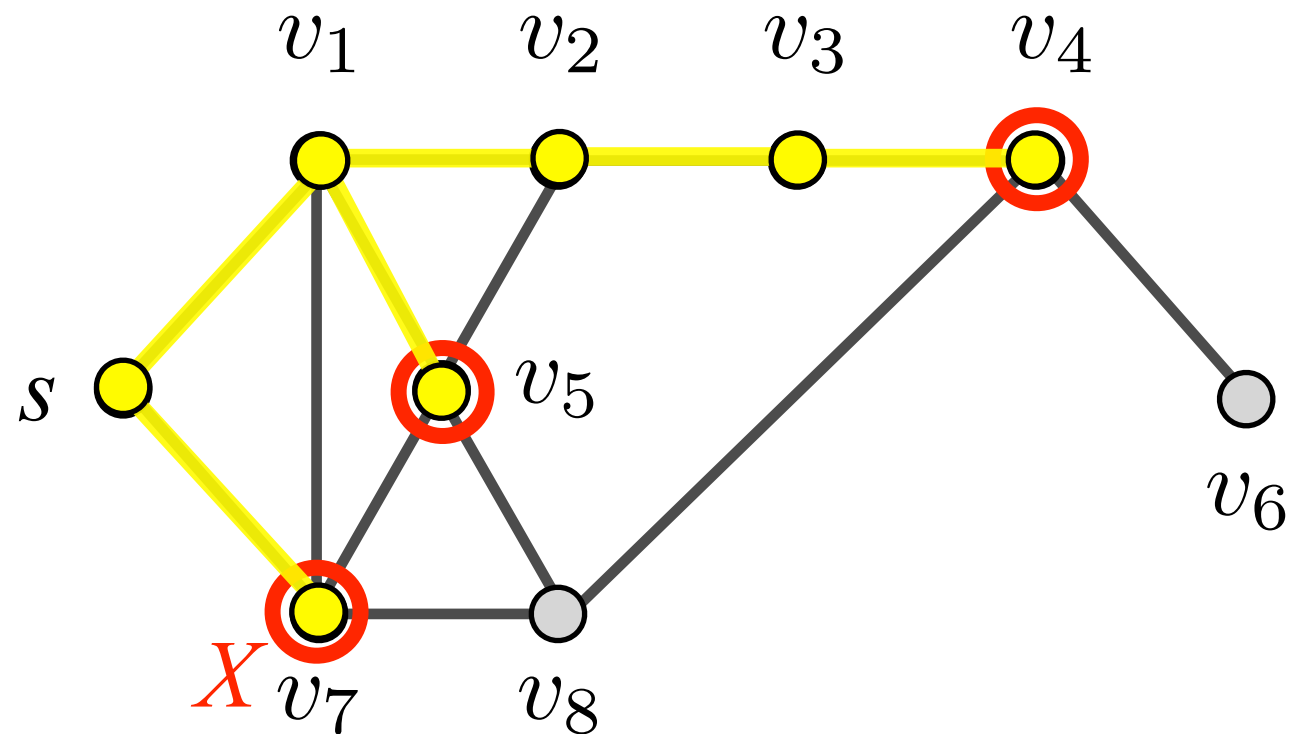
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

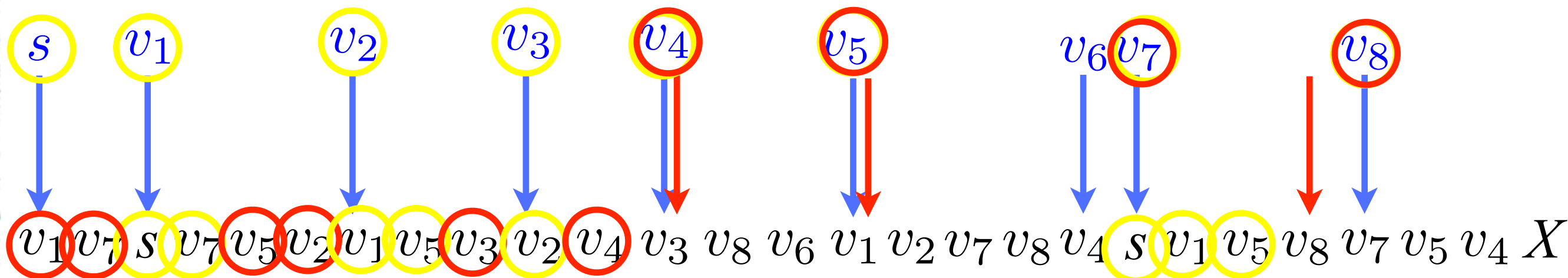
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

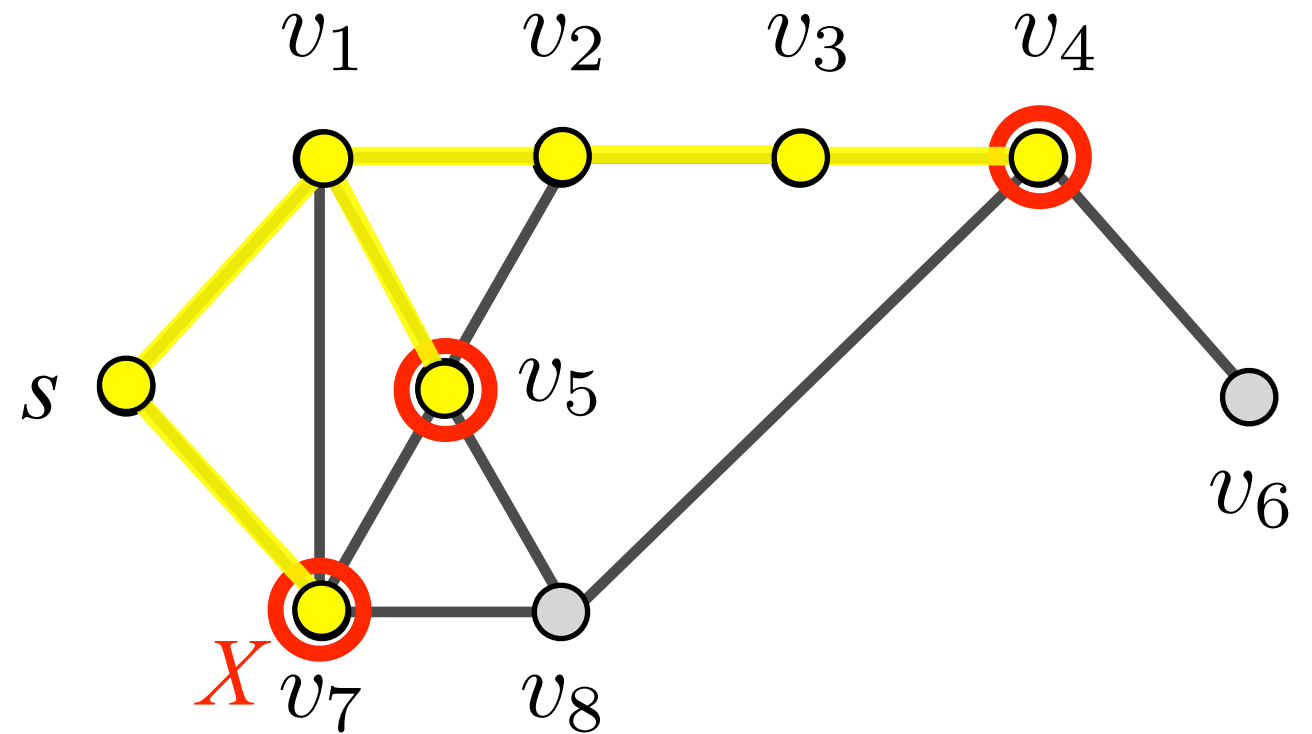
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

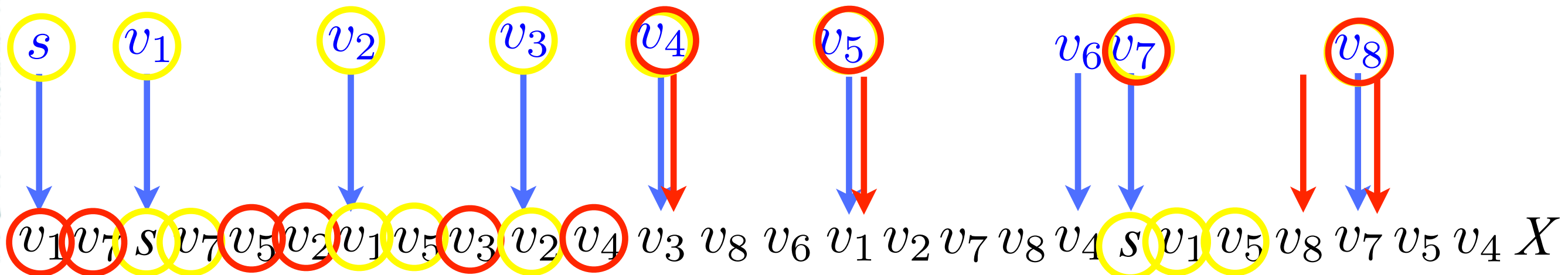
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

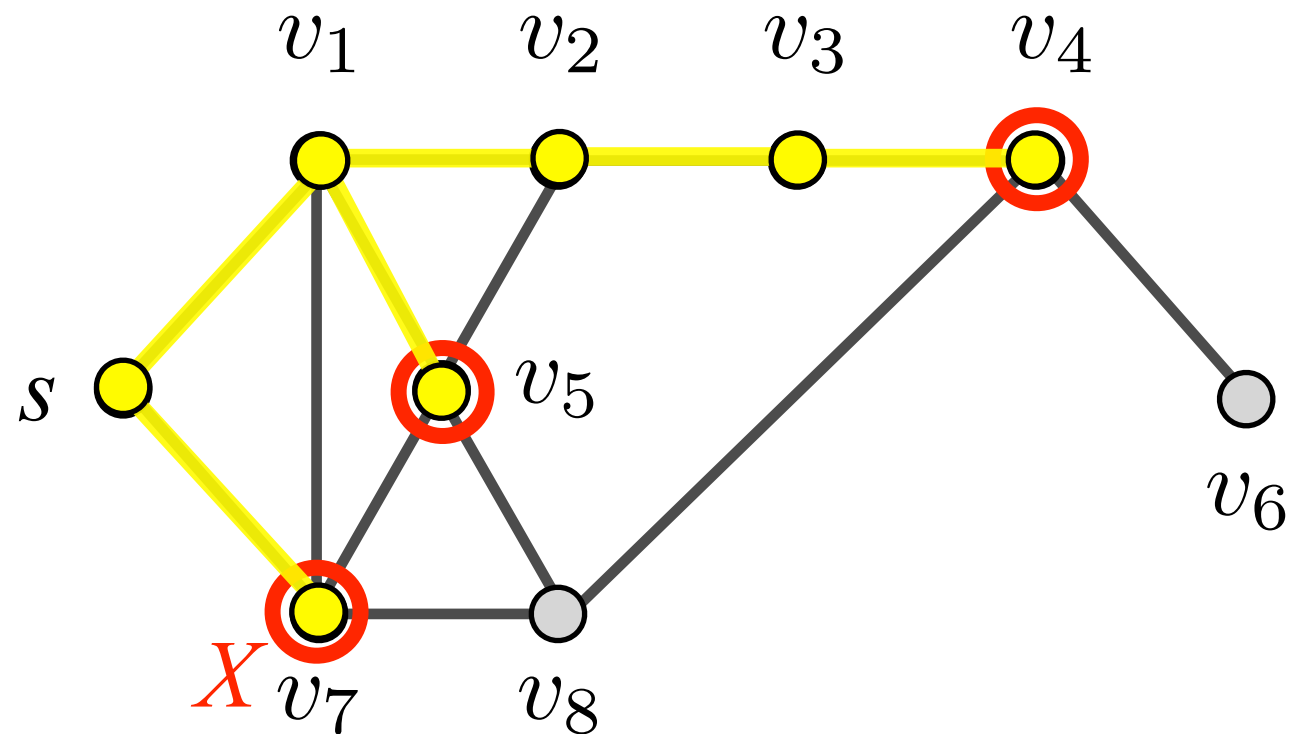
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

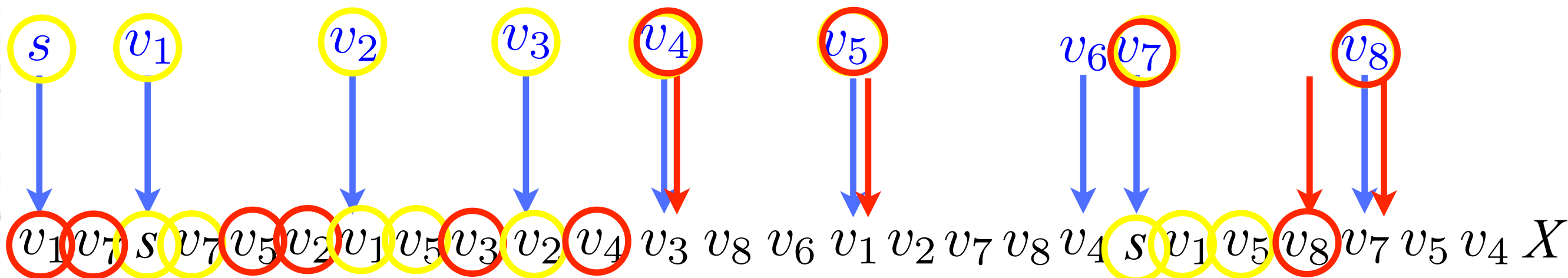
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

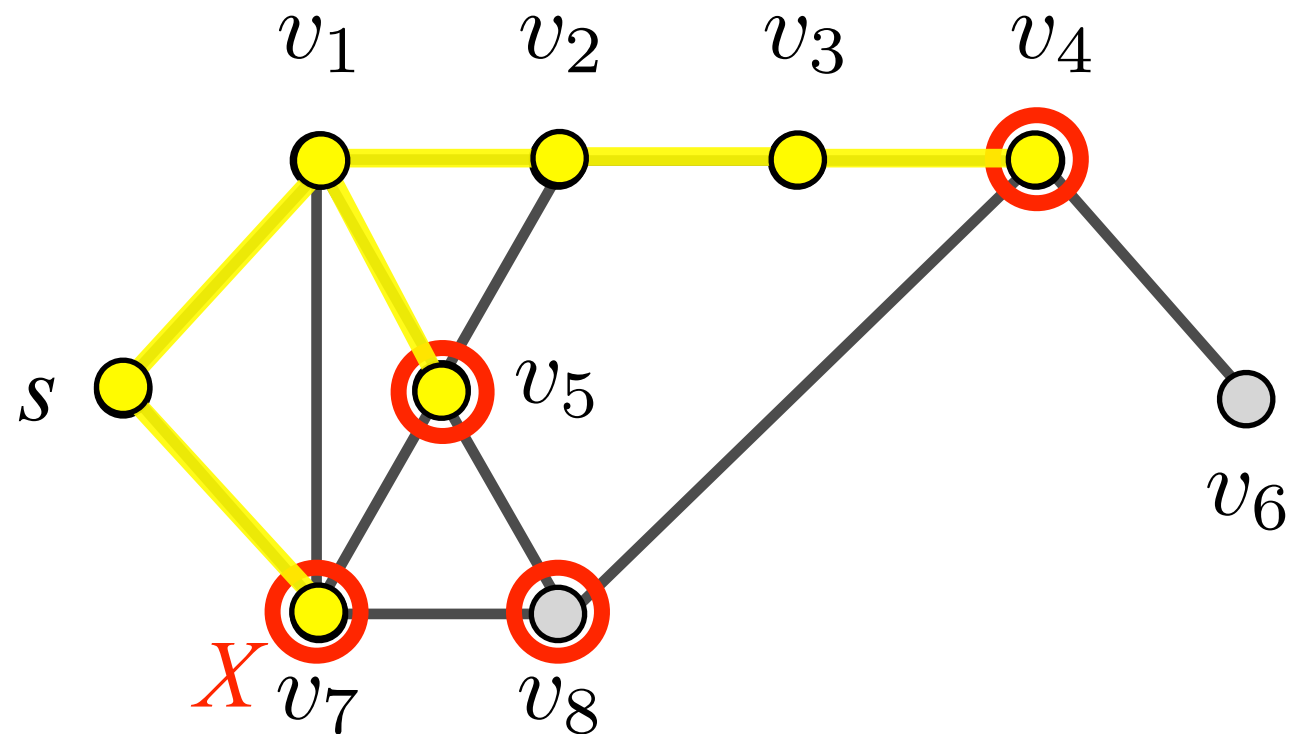
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

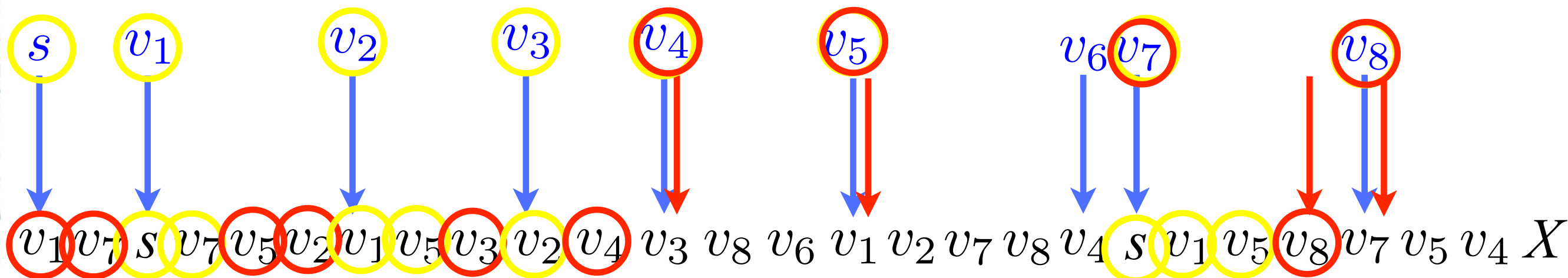
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

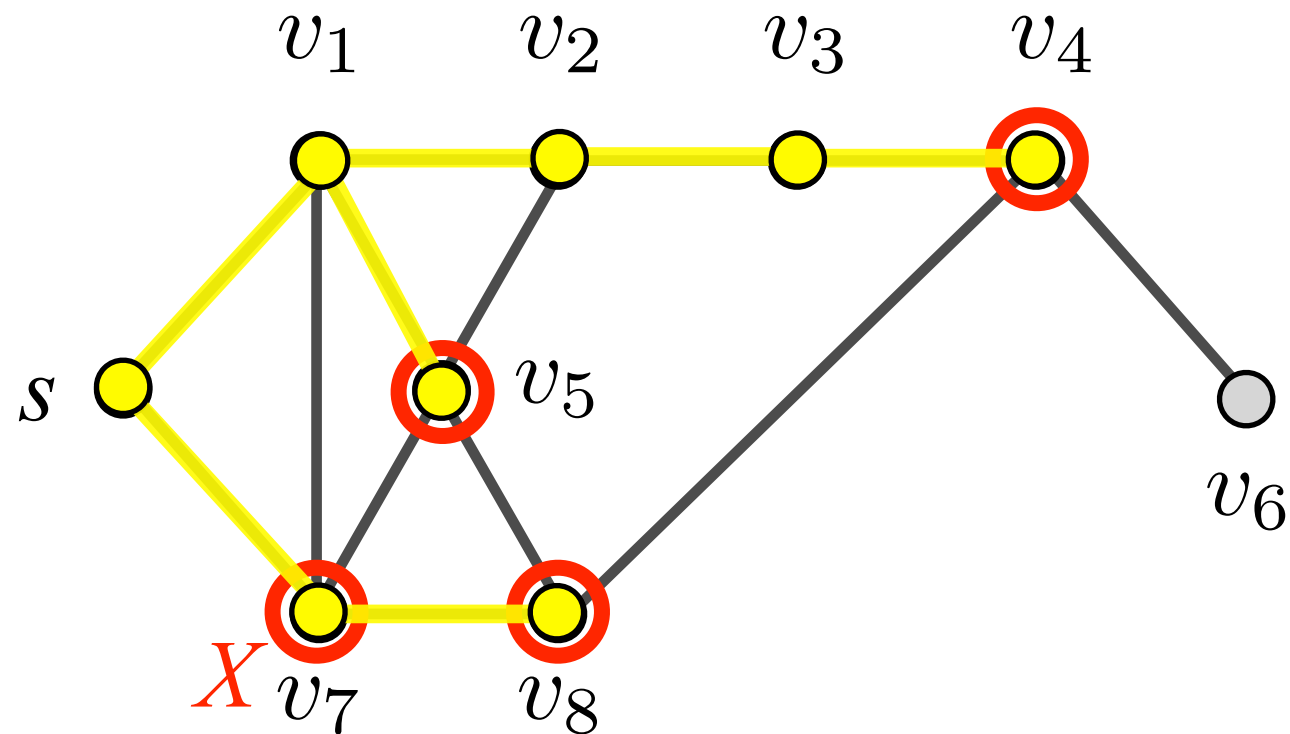
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

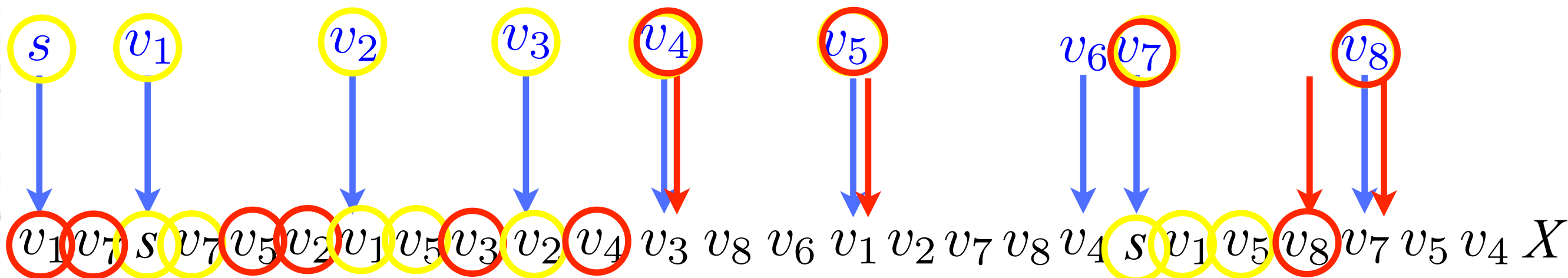
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

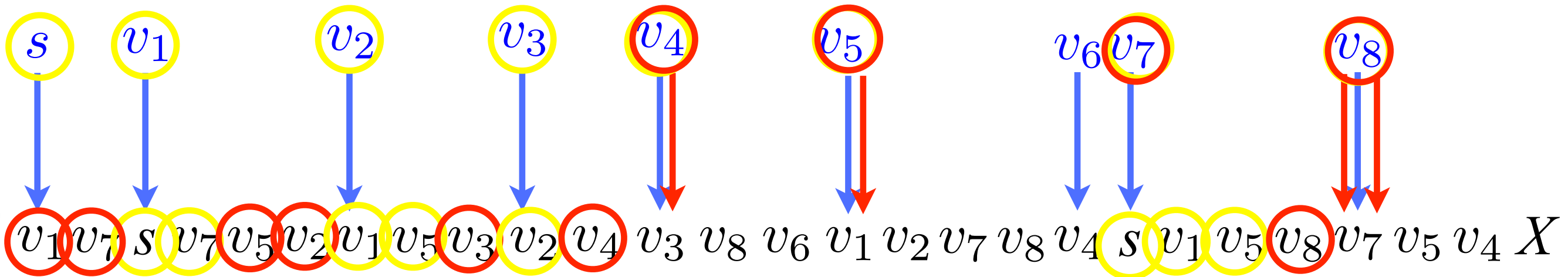
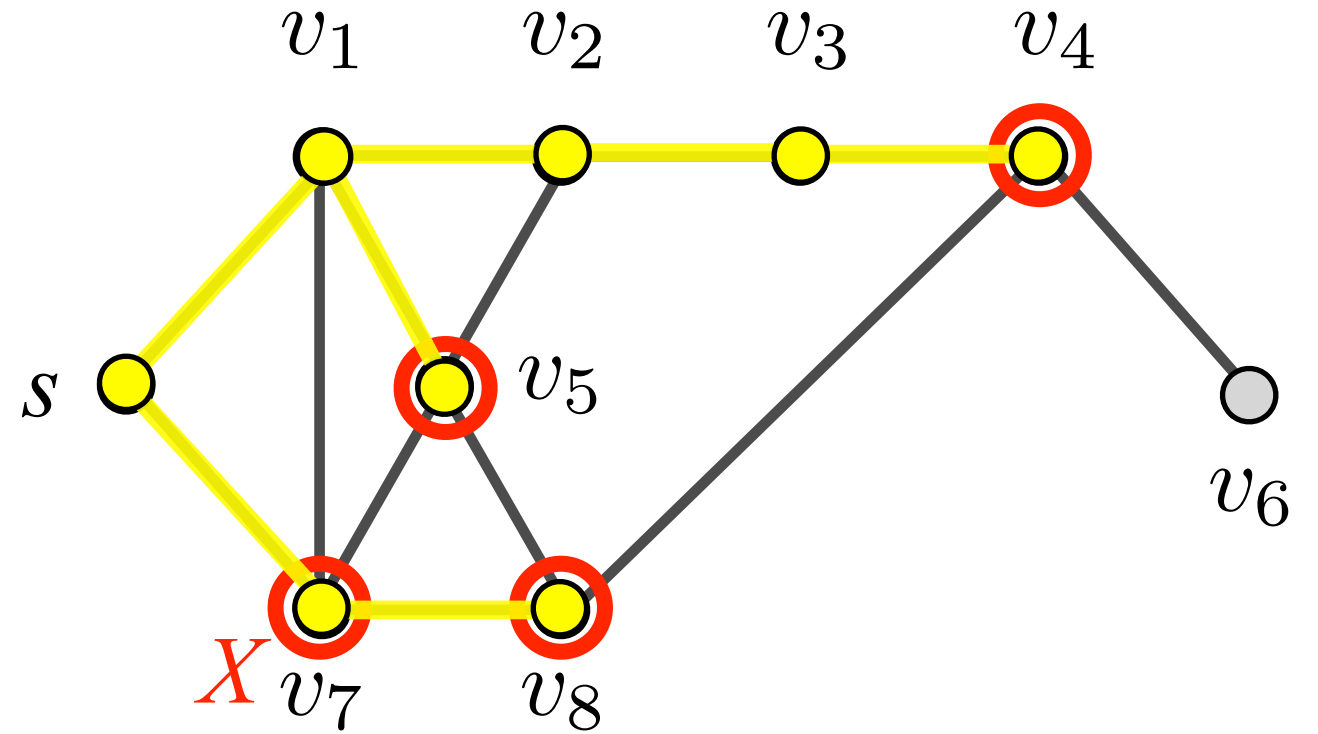
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

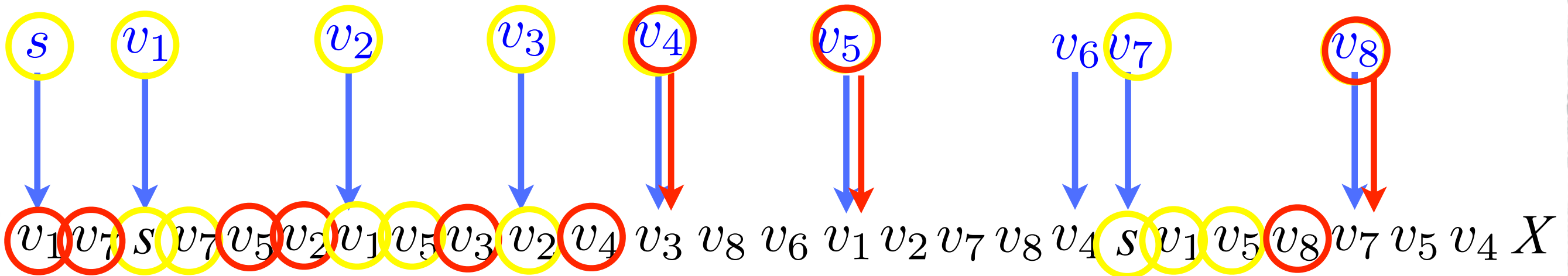
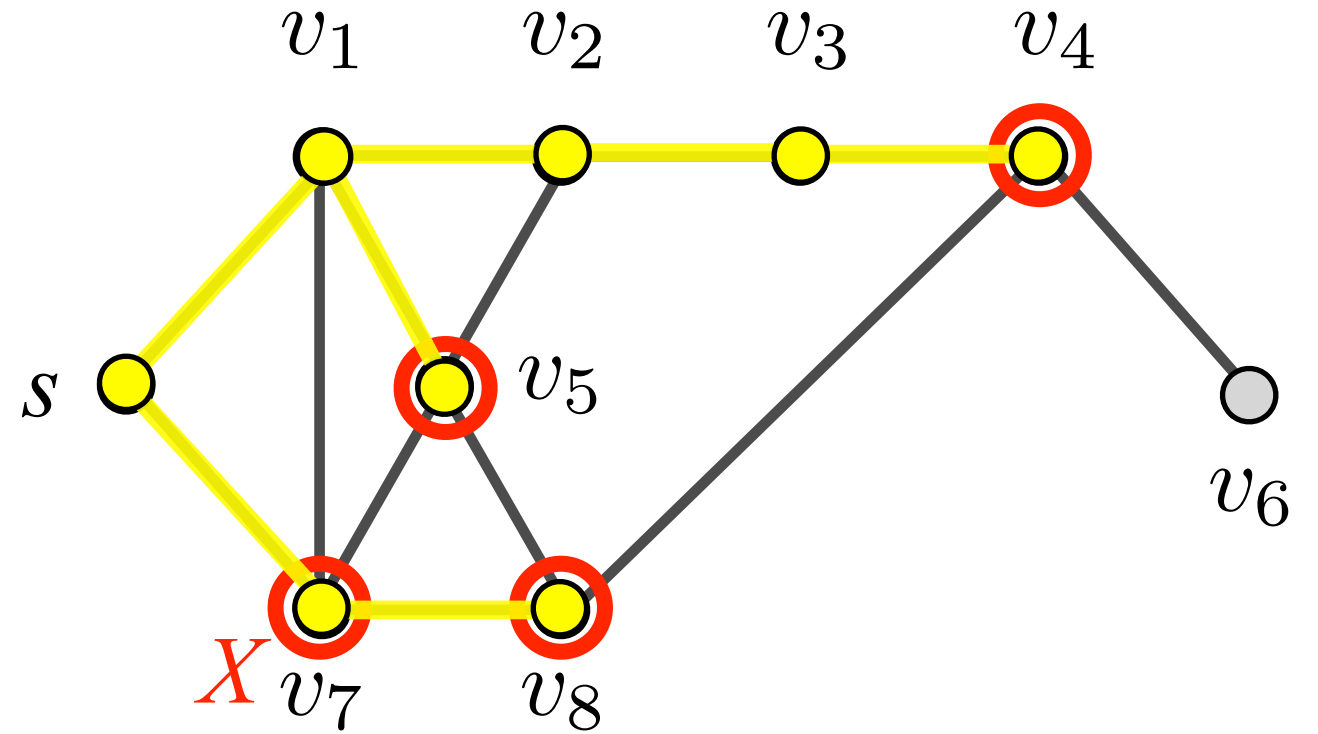
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

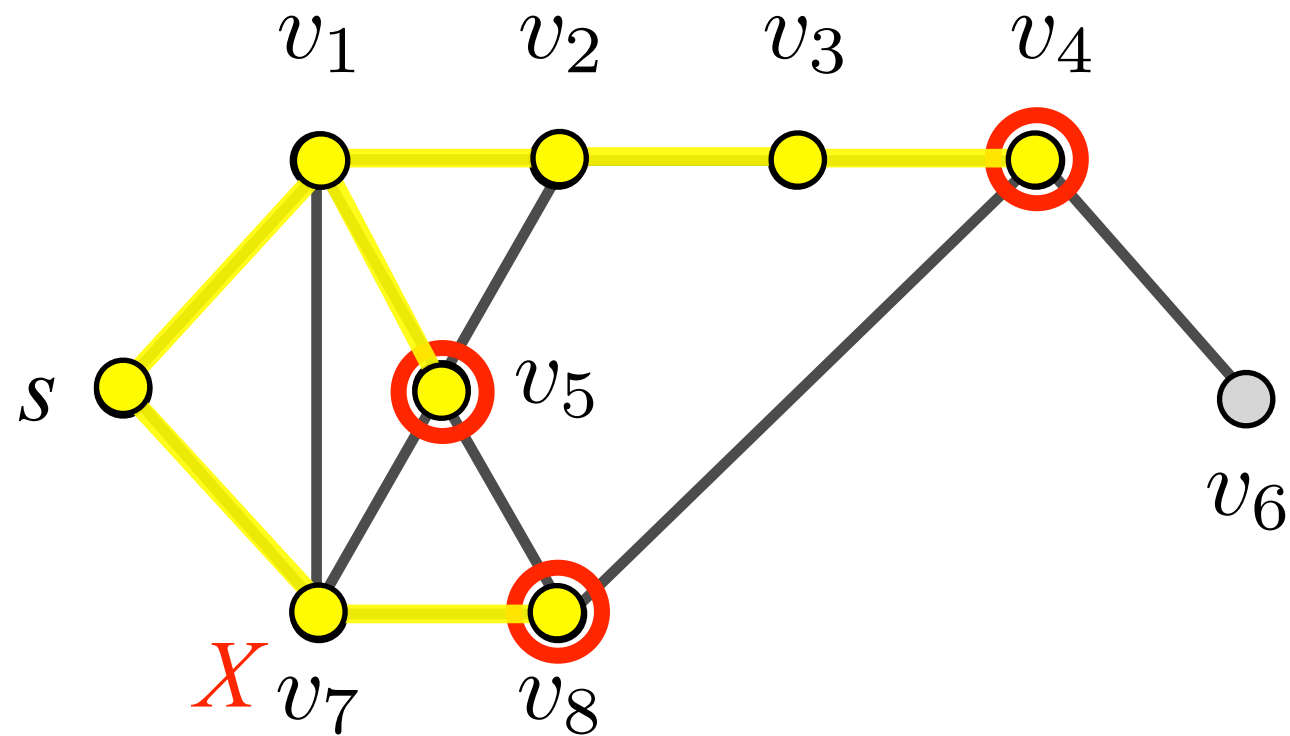
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

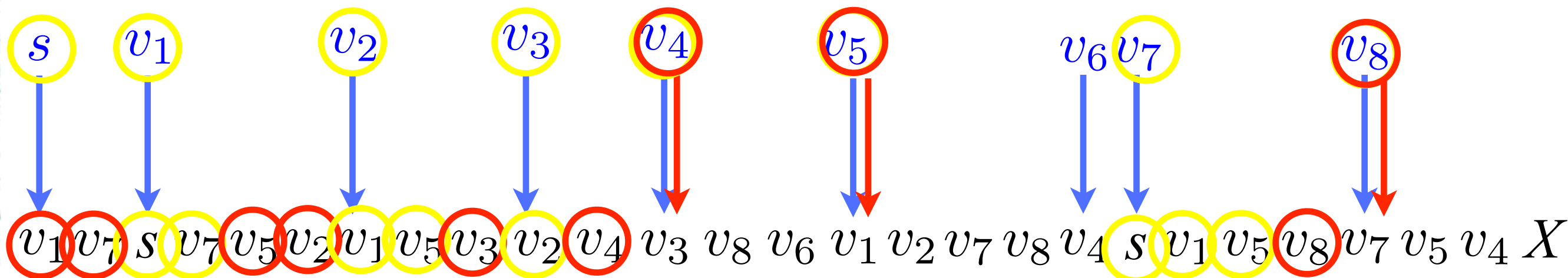
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

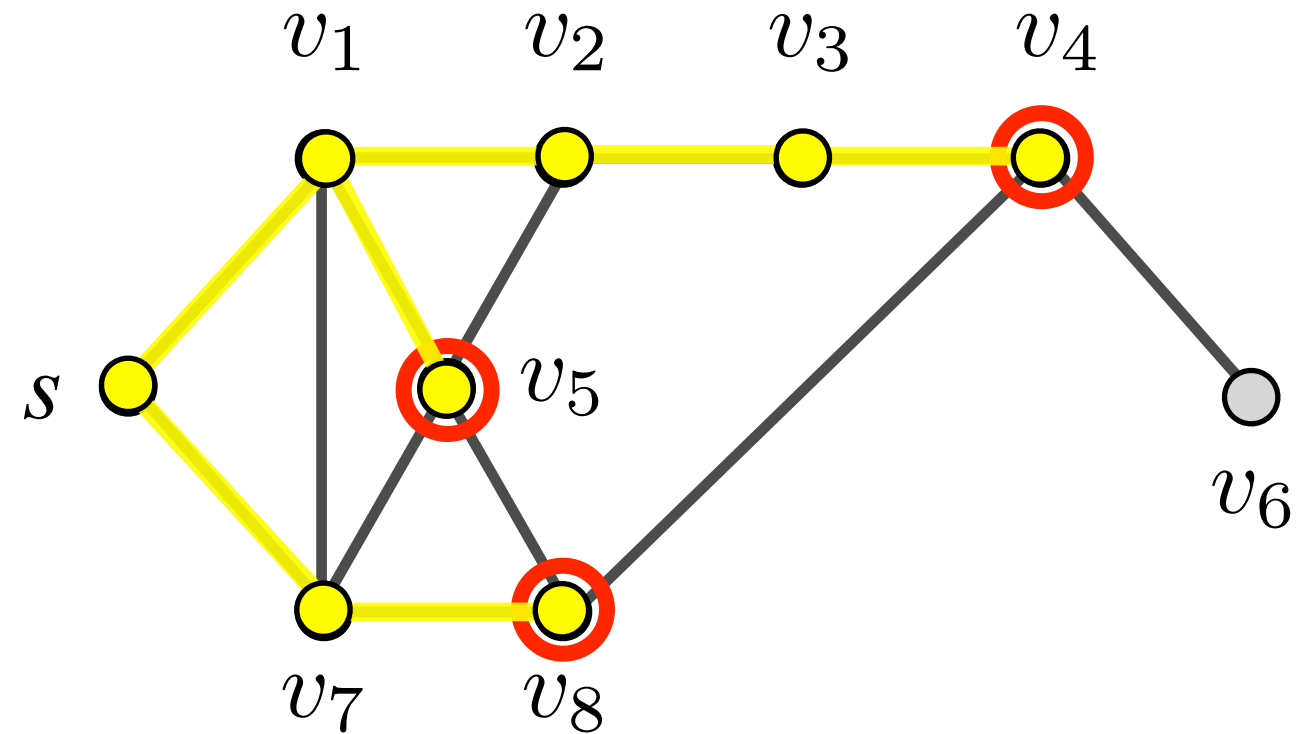
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

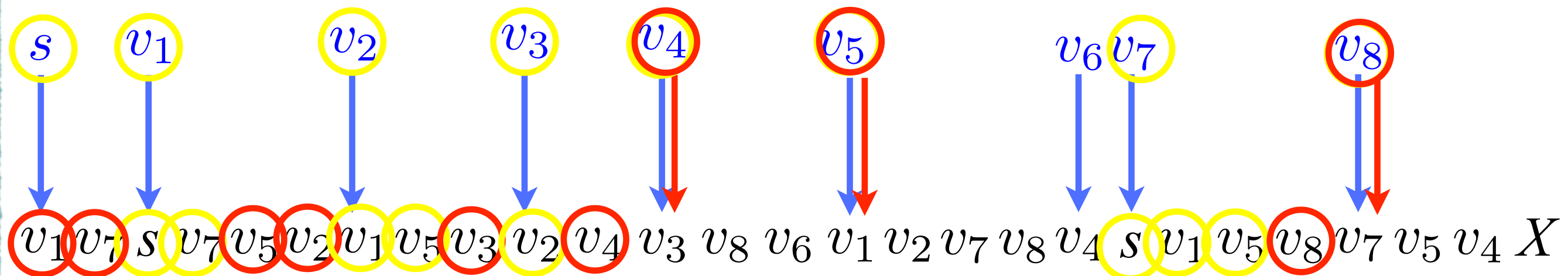
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V,E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R:=\{s\}$, $Y:=\{s\}$, $T:=\emptyset$

```
WHILE (R≠∅) DO {
```

2.1. Wähle $v \in \mathbb{R}$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

```
2.3. ELSE {
```

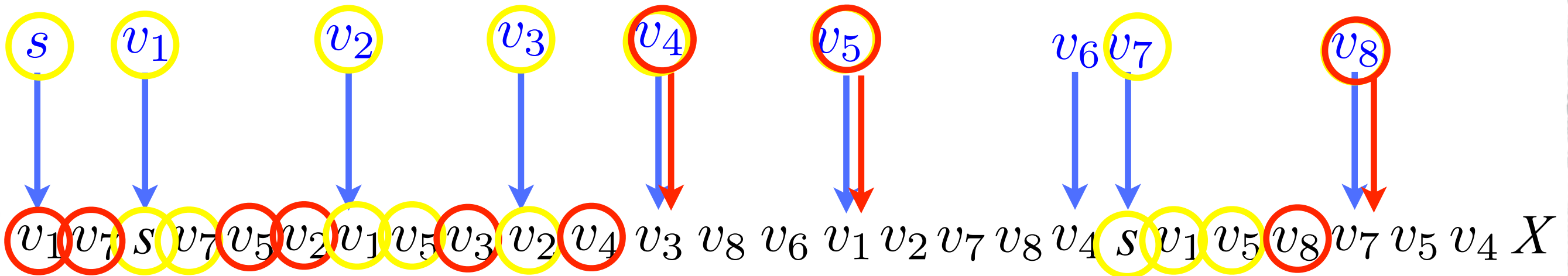
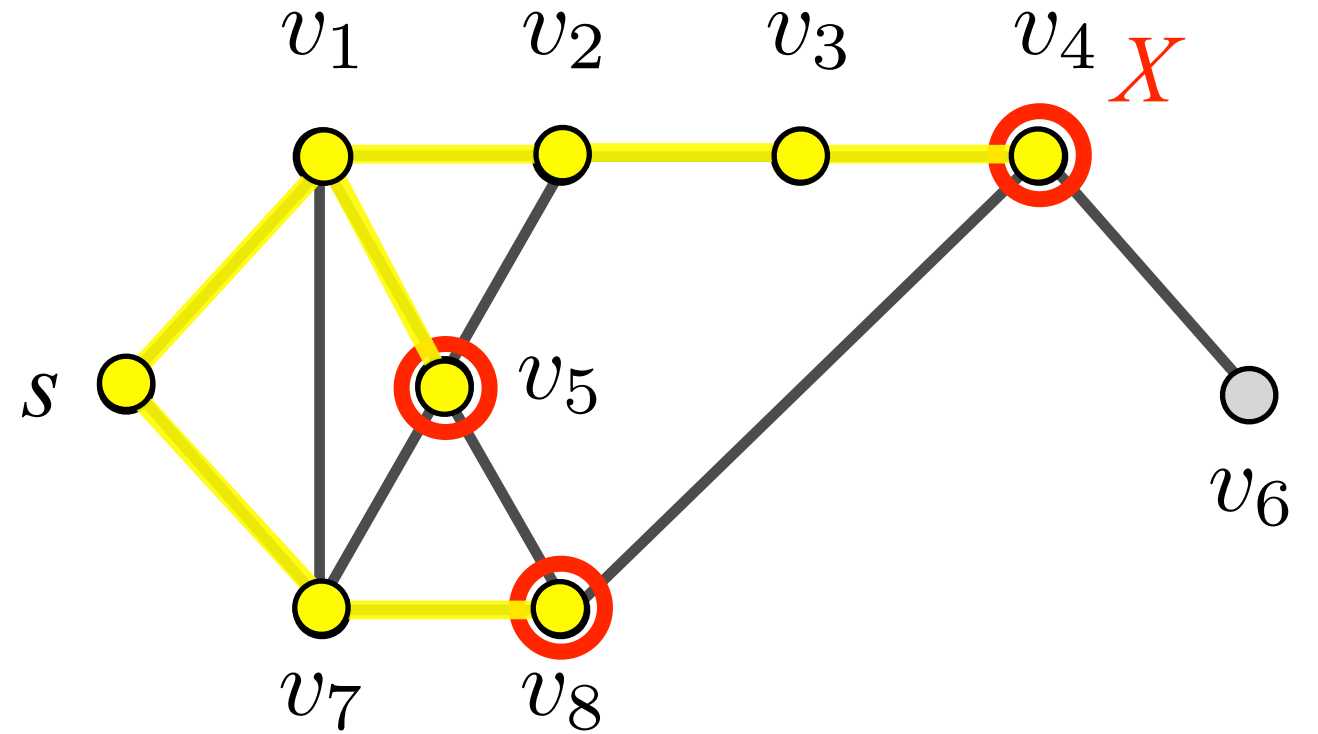
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

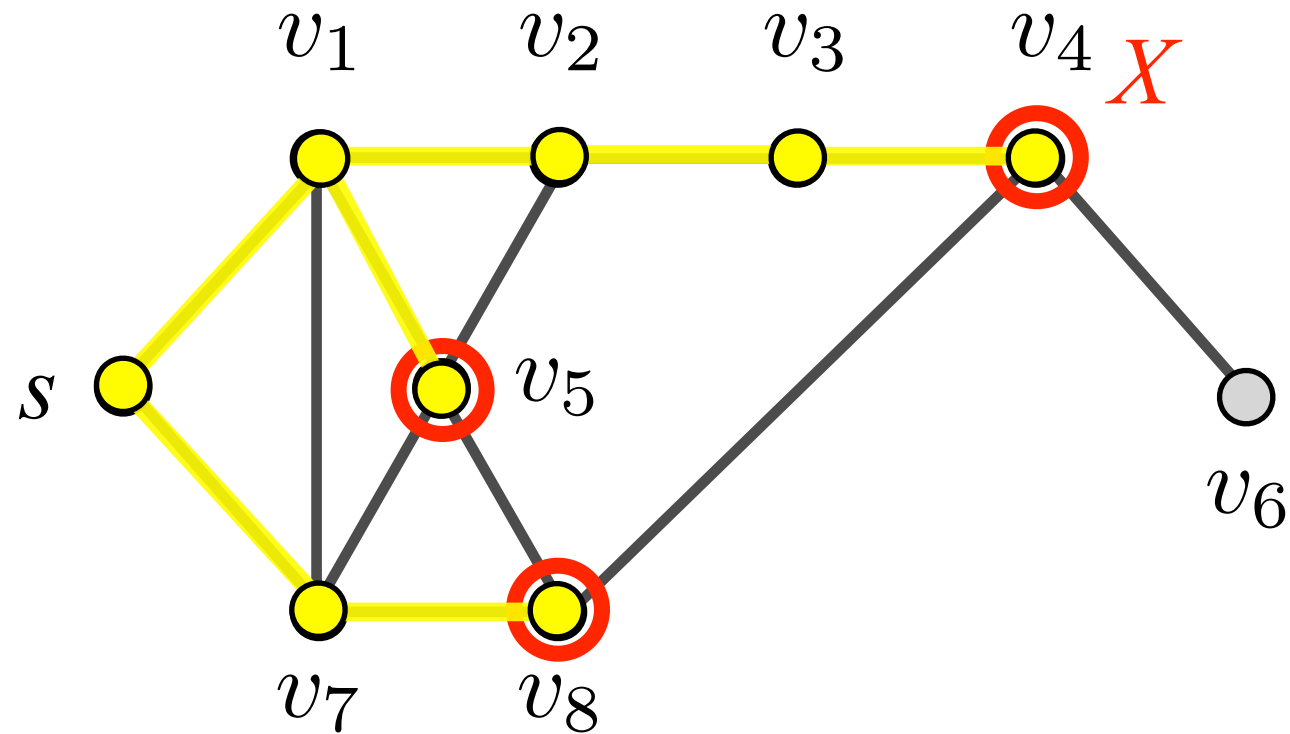
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

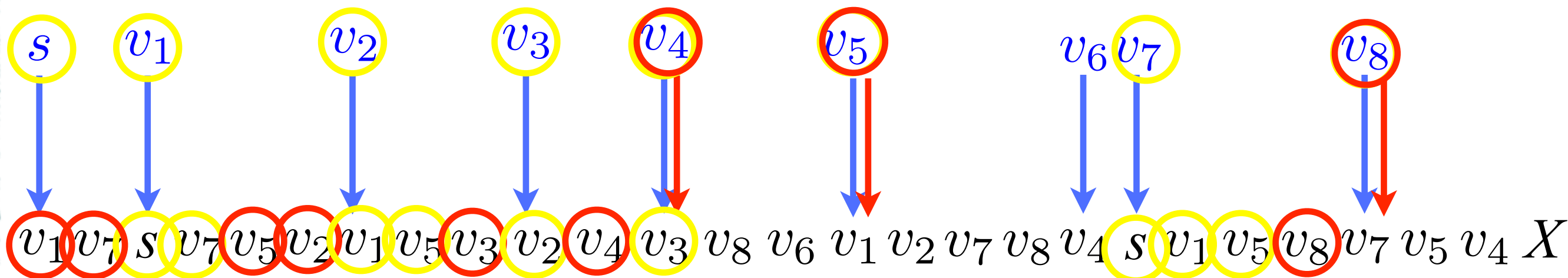
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

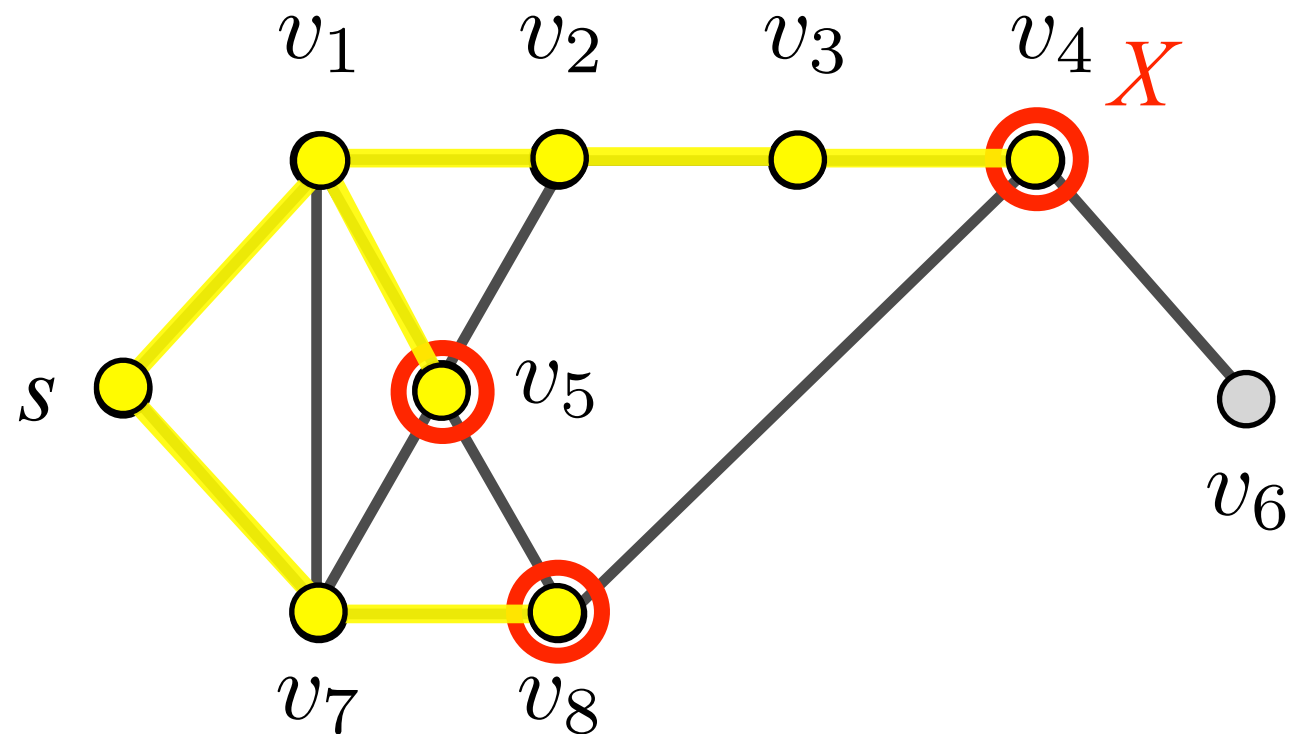
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

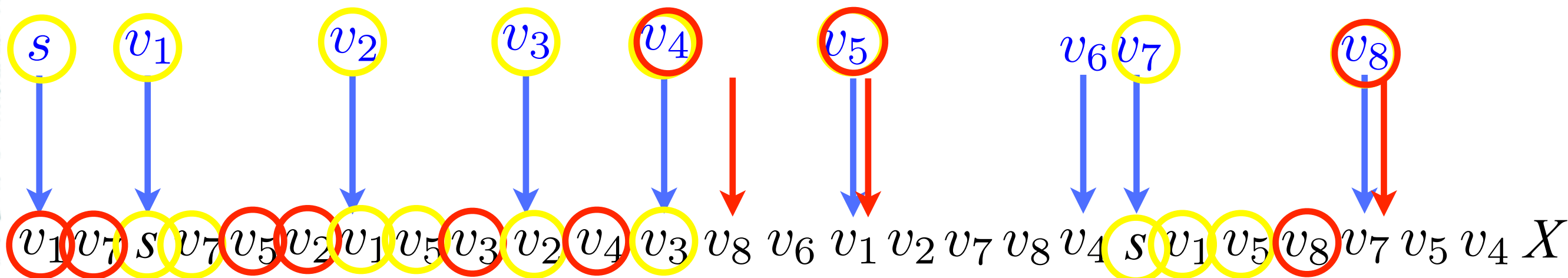
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

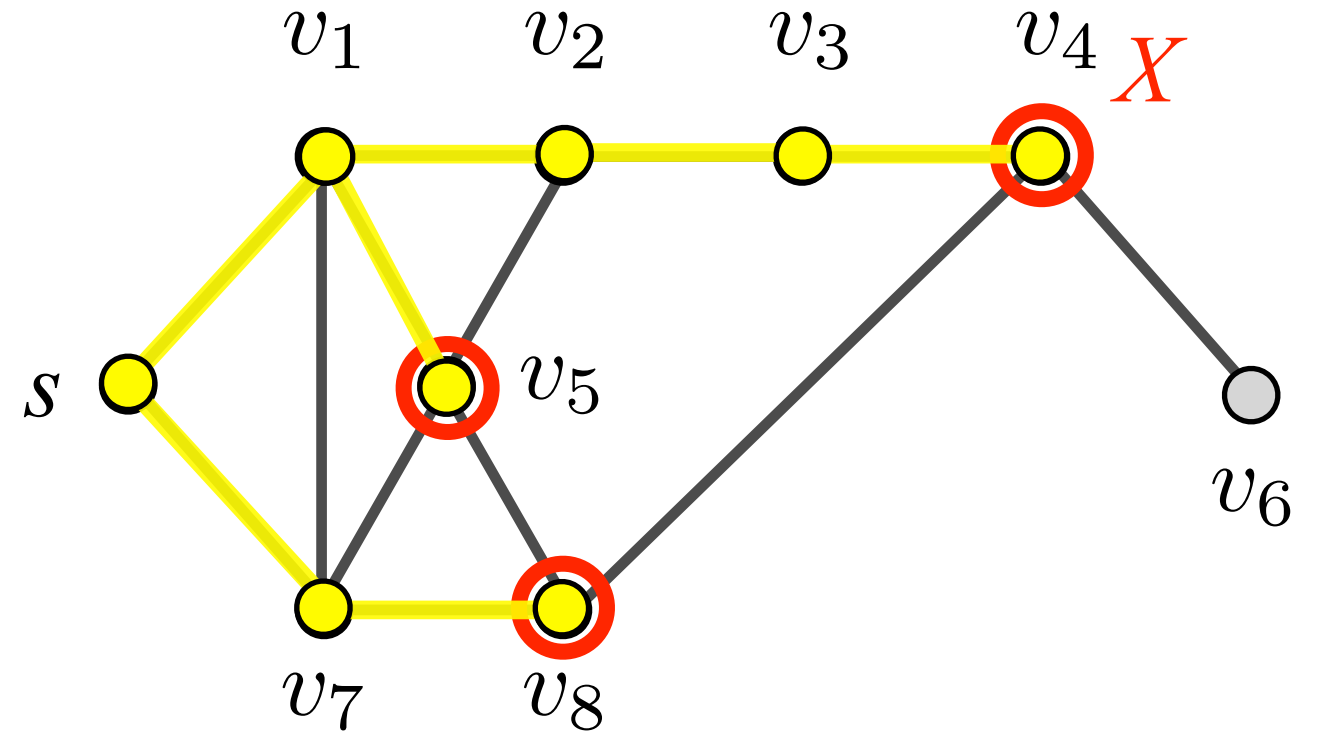
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

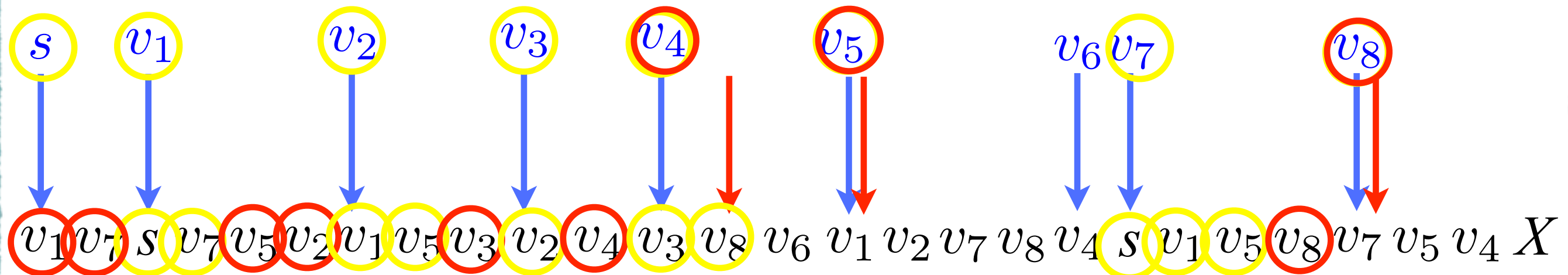
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

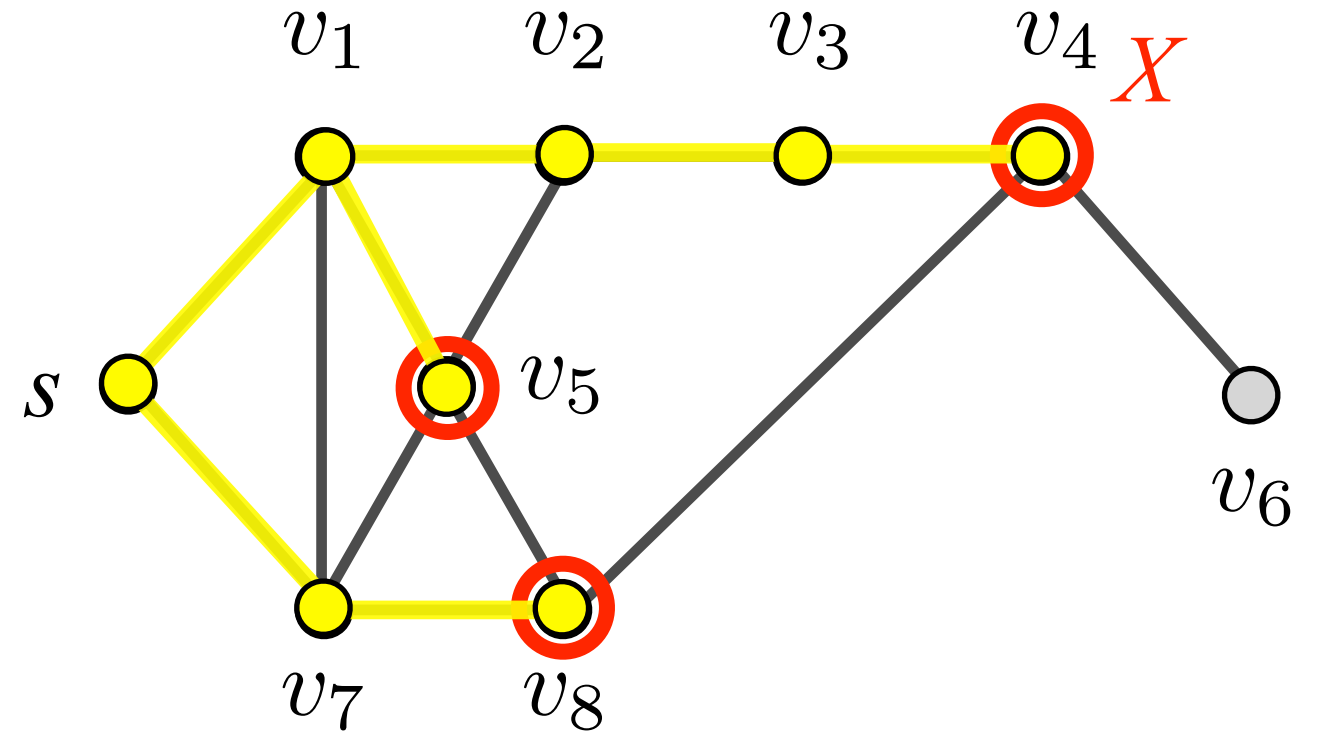
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

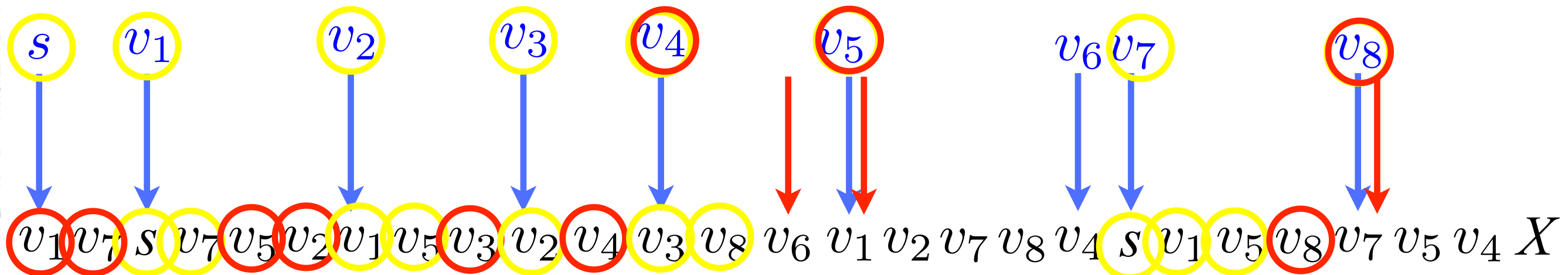
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

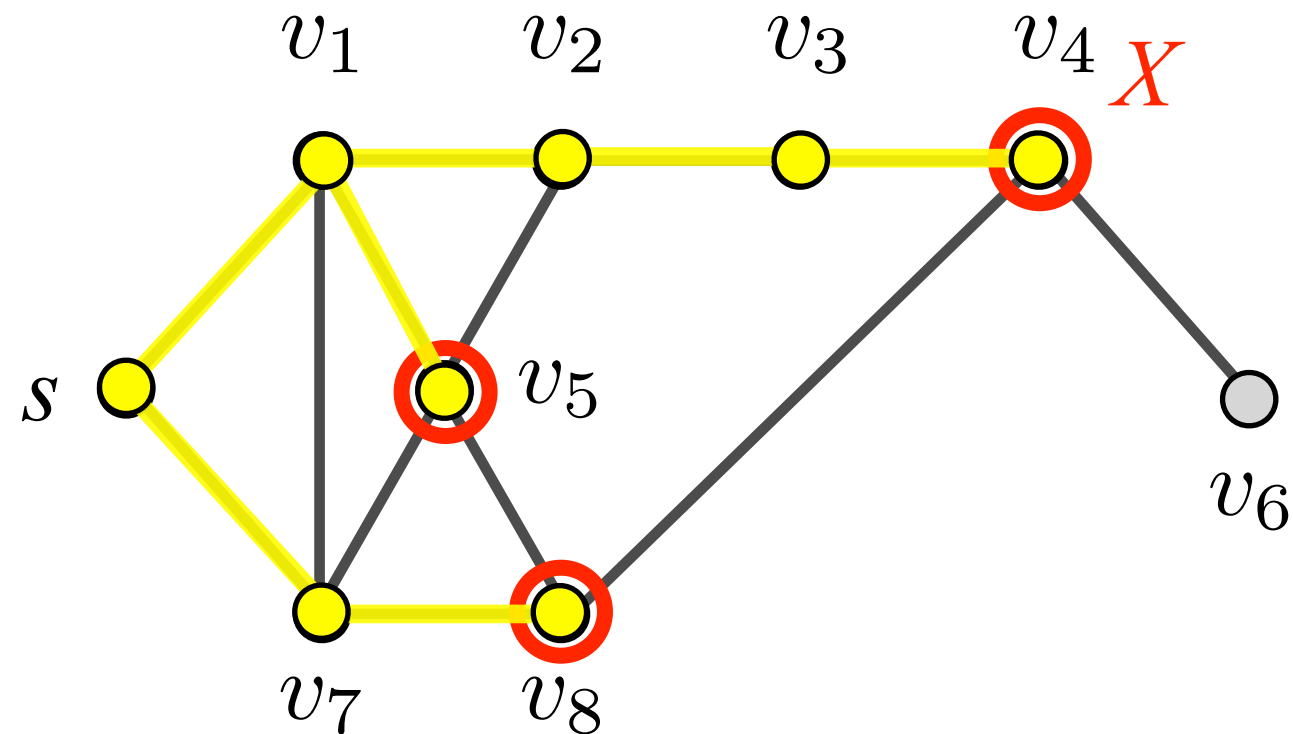
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

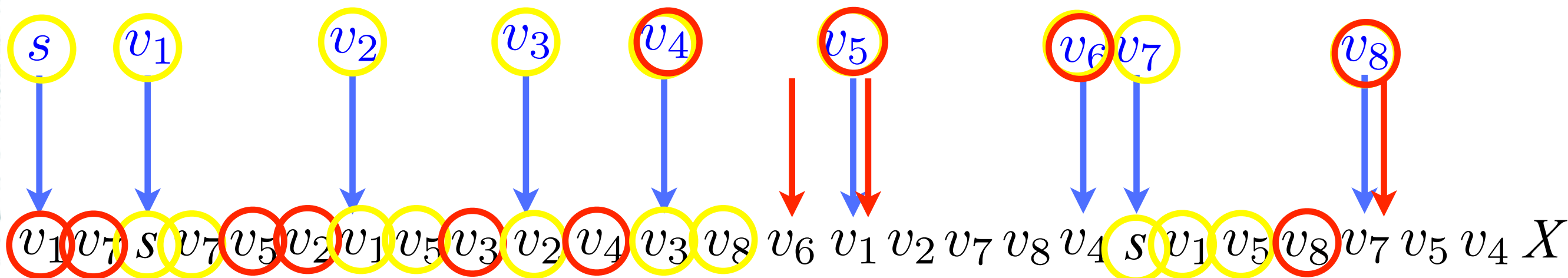
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

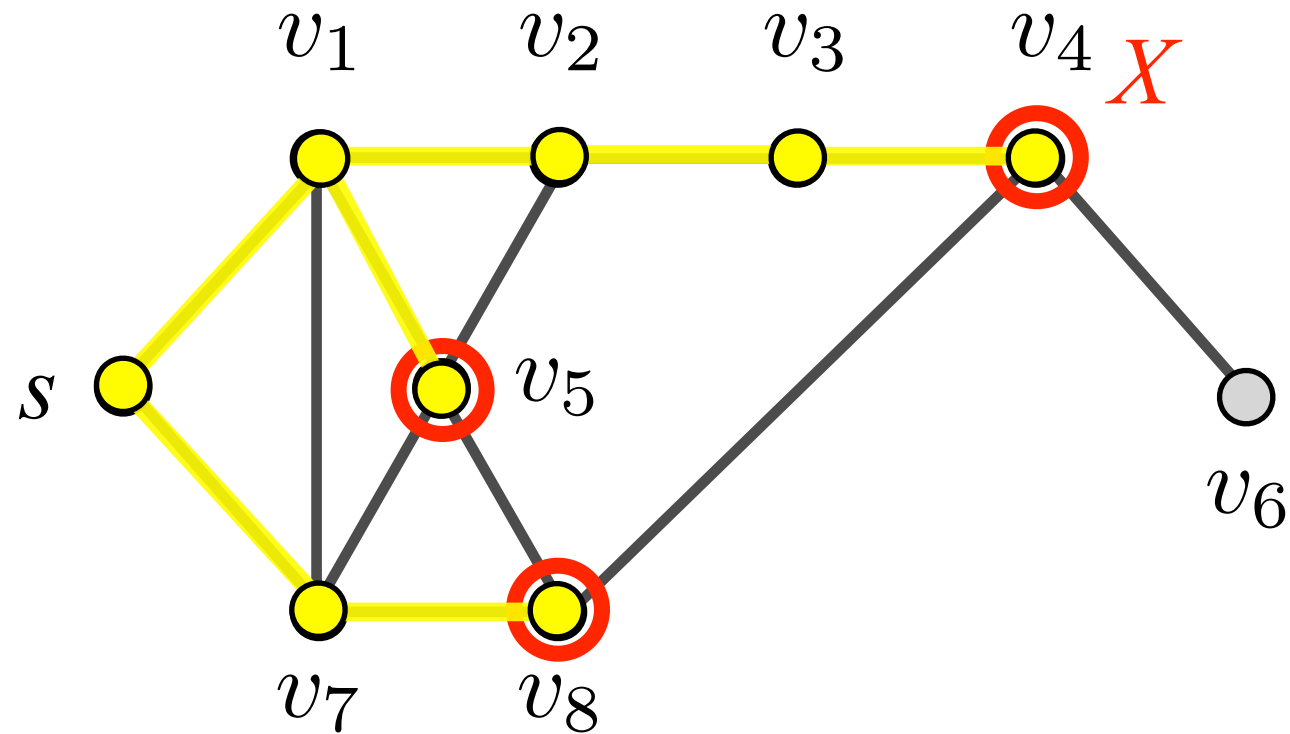
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

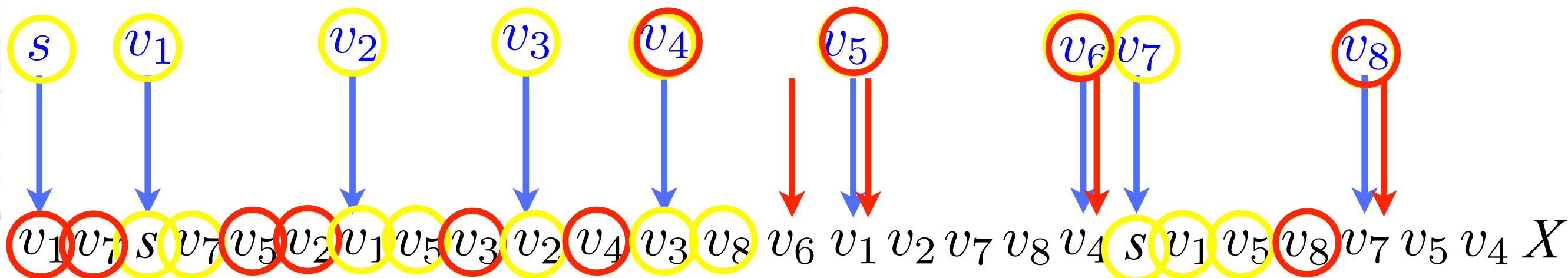
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

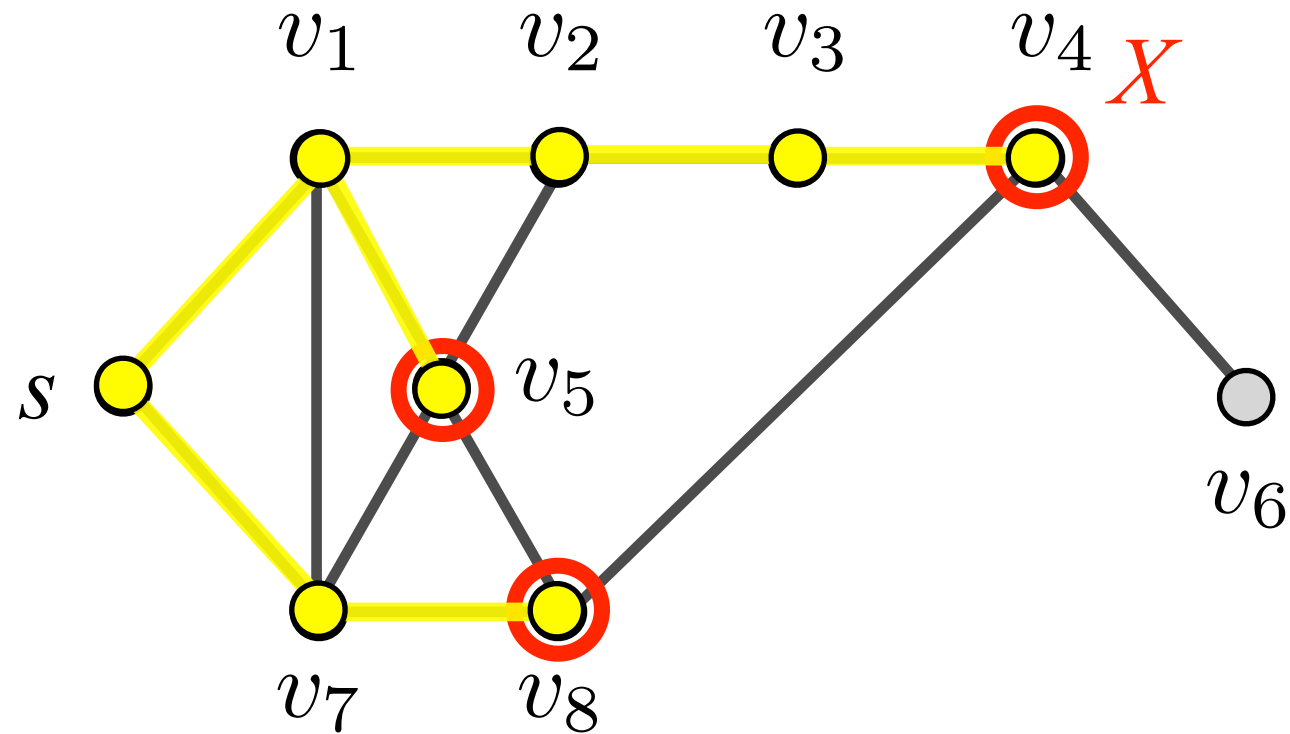
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

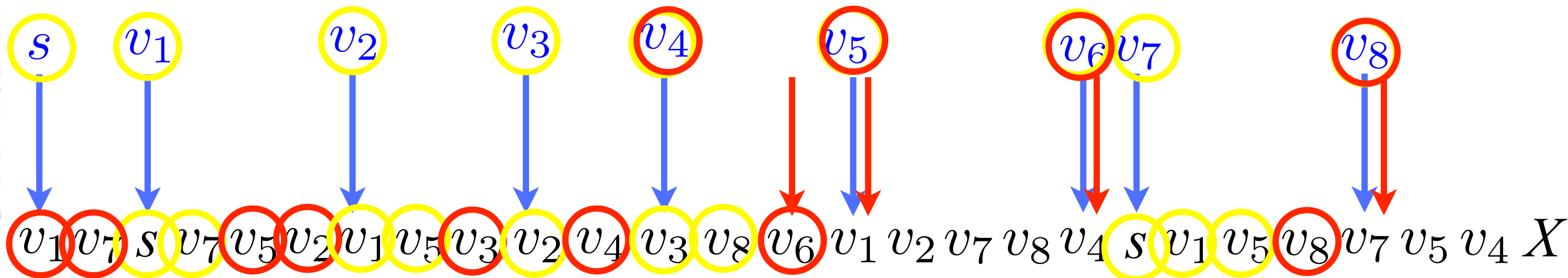
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

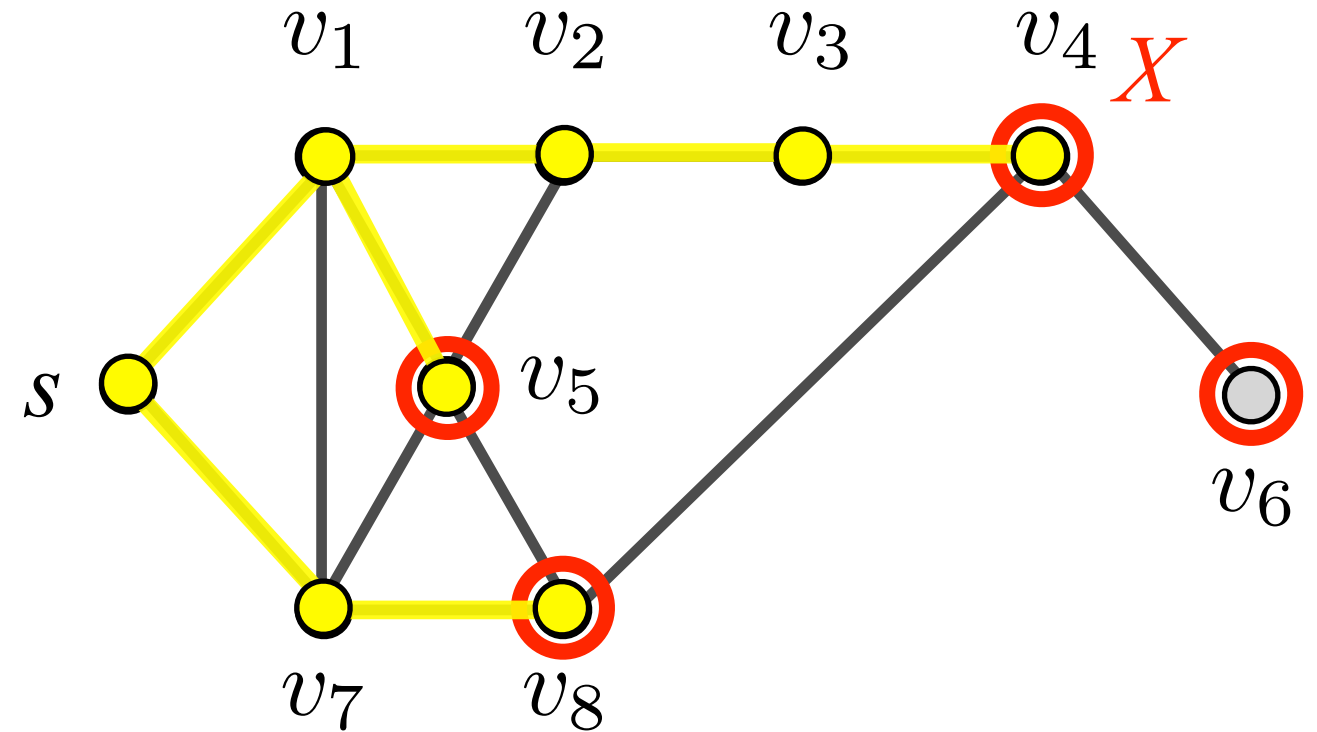
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

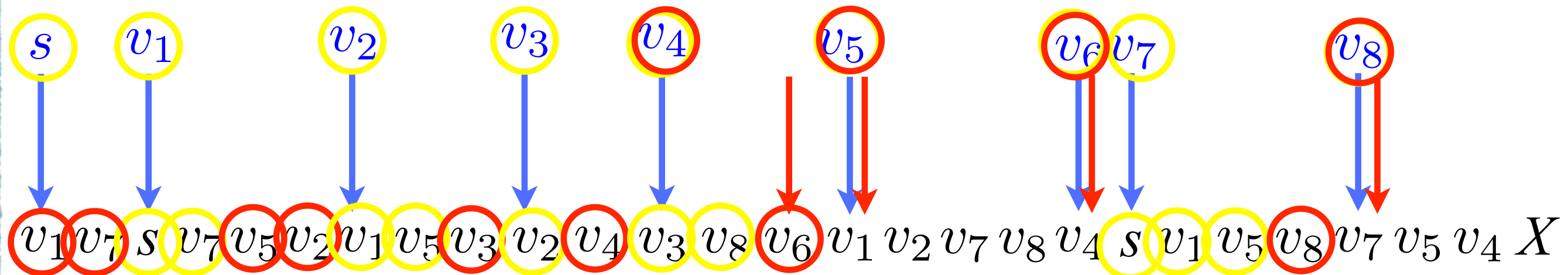
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

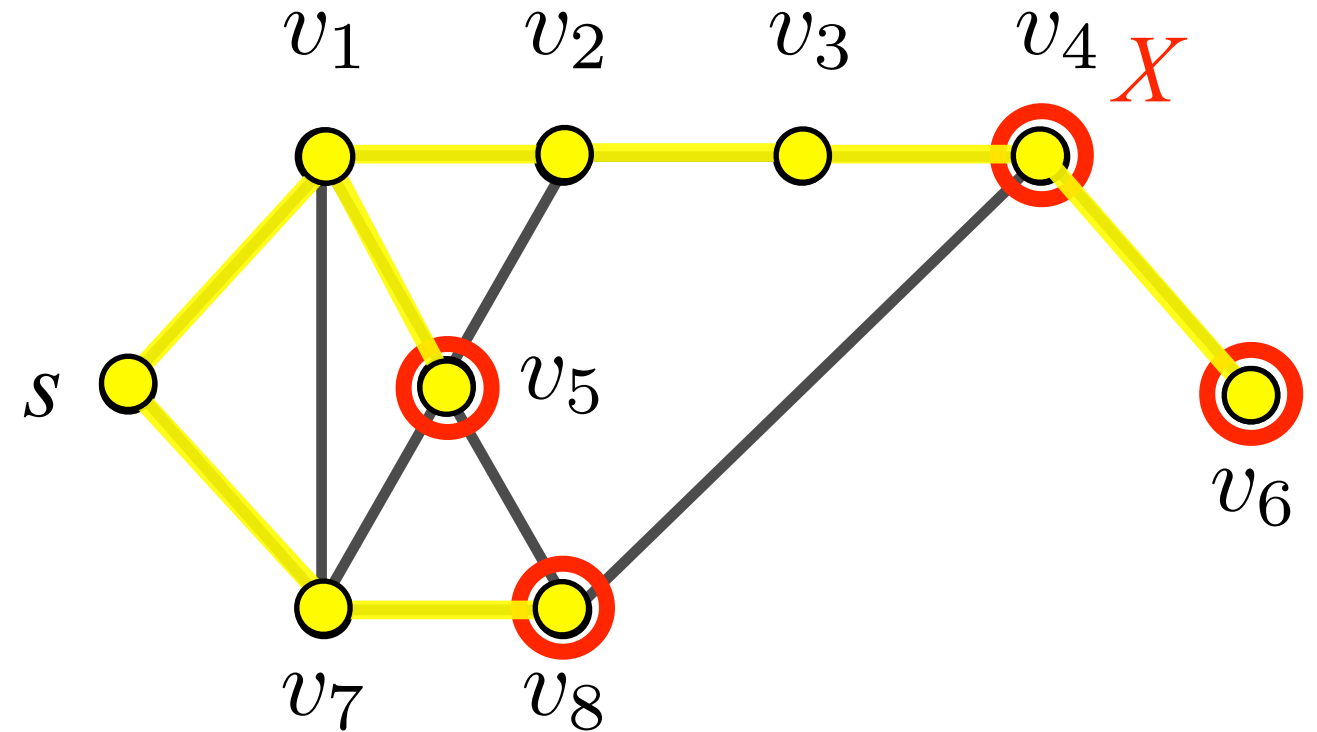
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

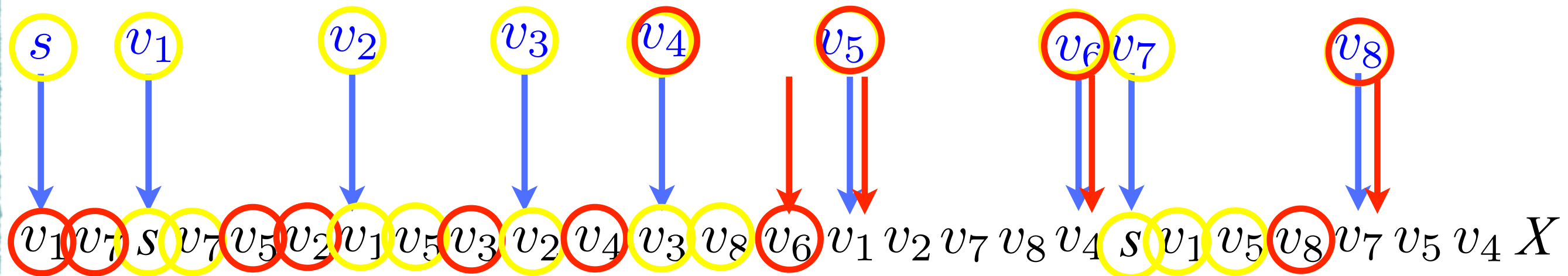
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

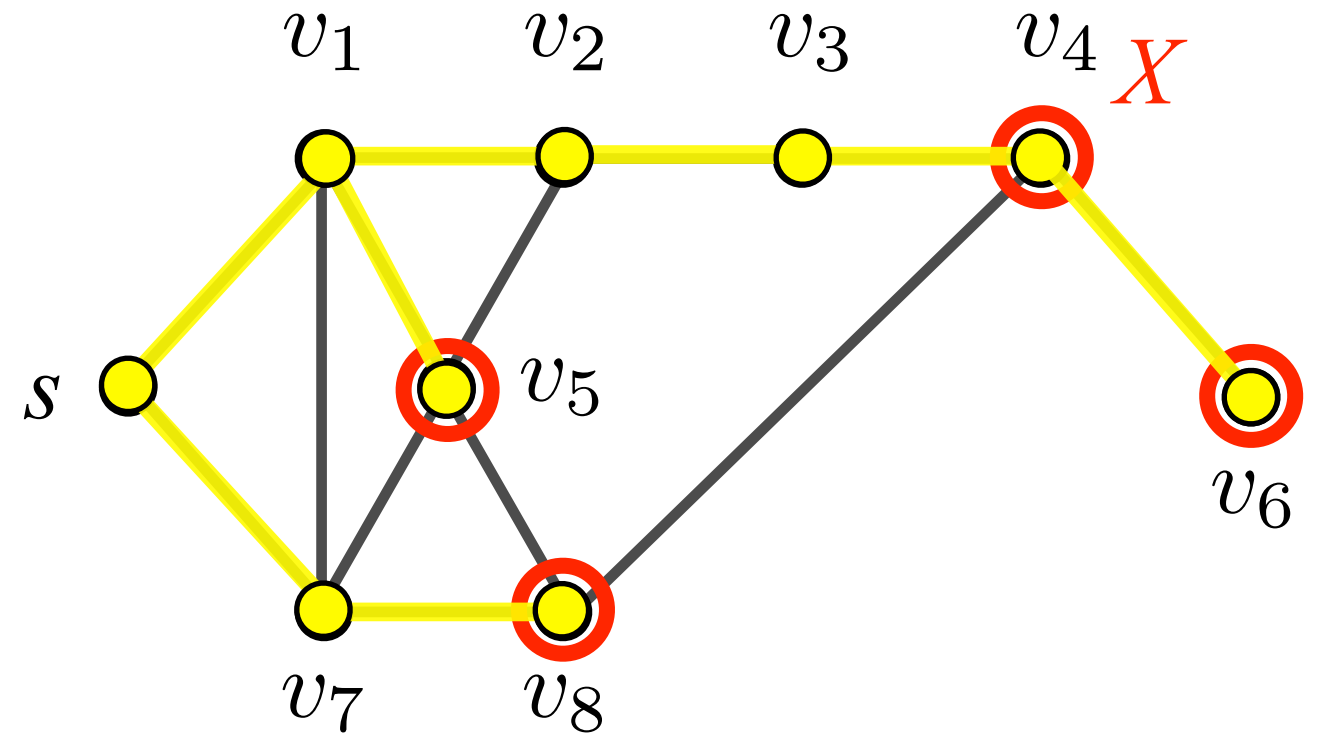
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

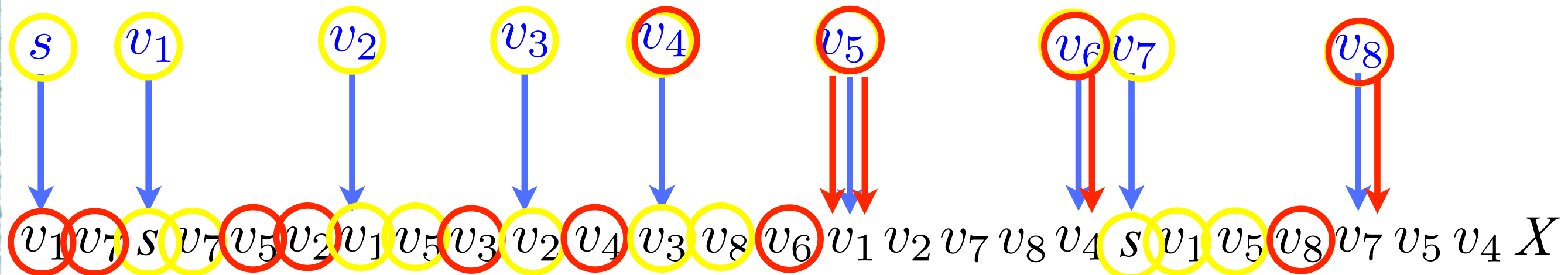
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

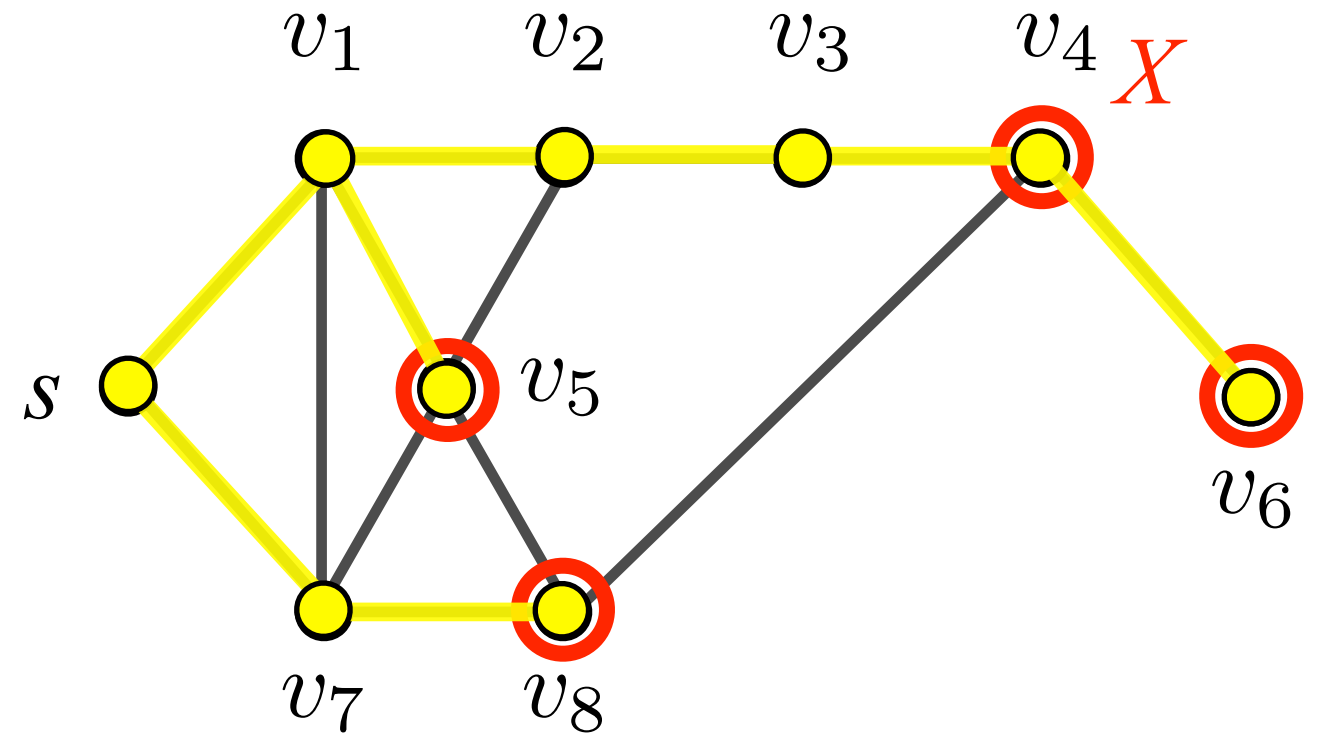
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

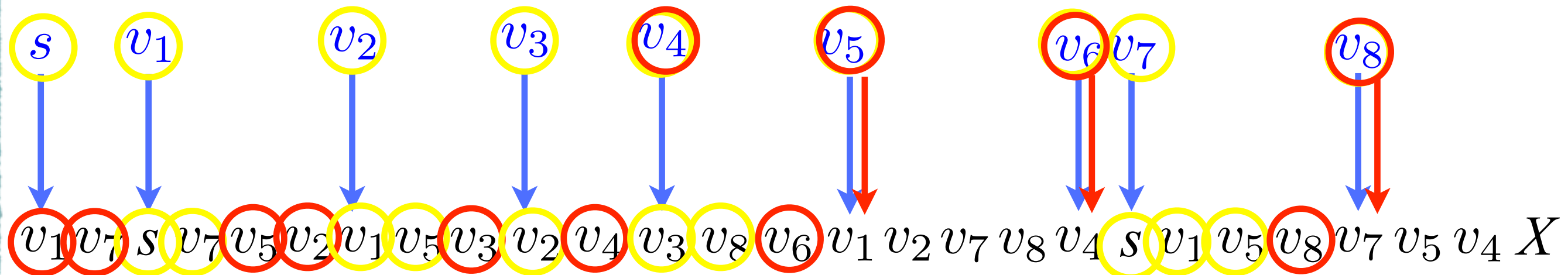
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

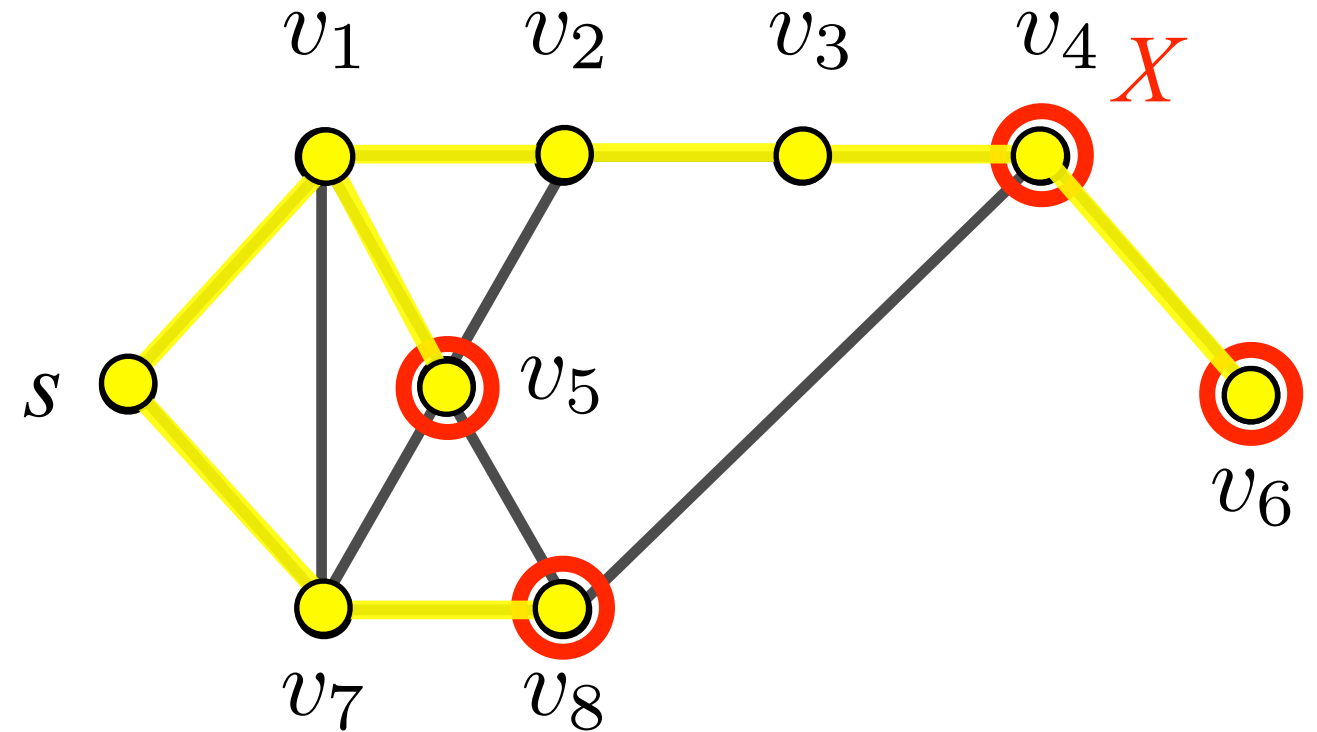
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

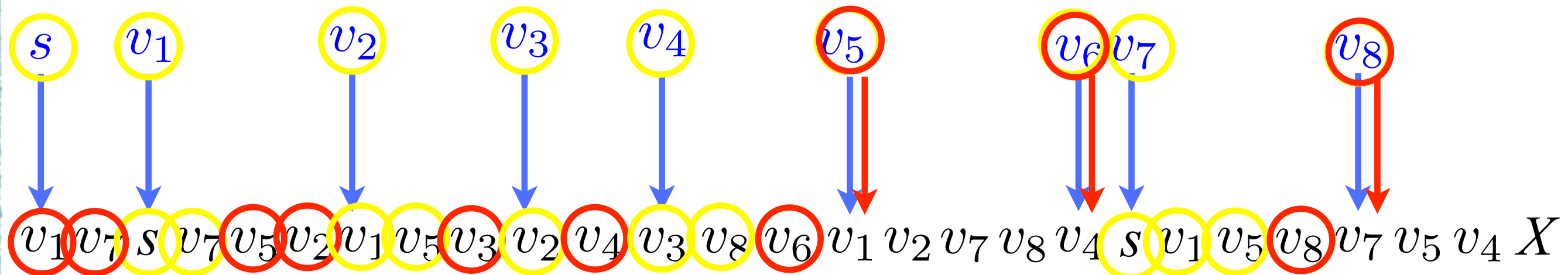
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

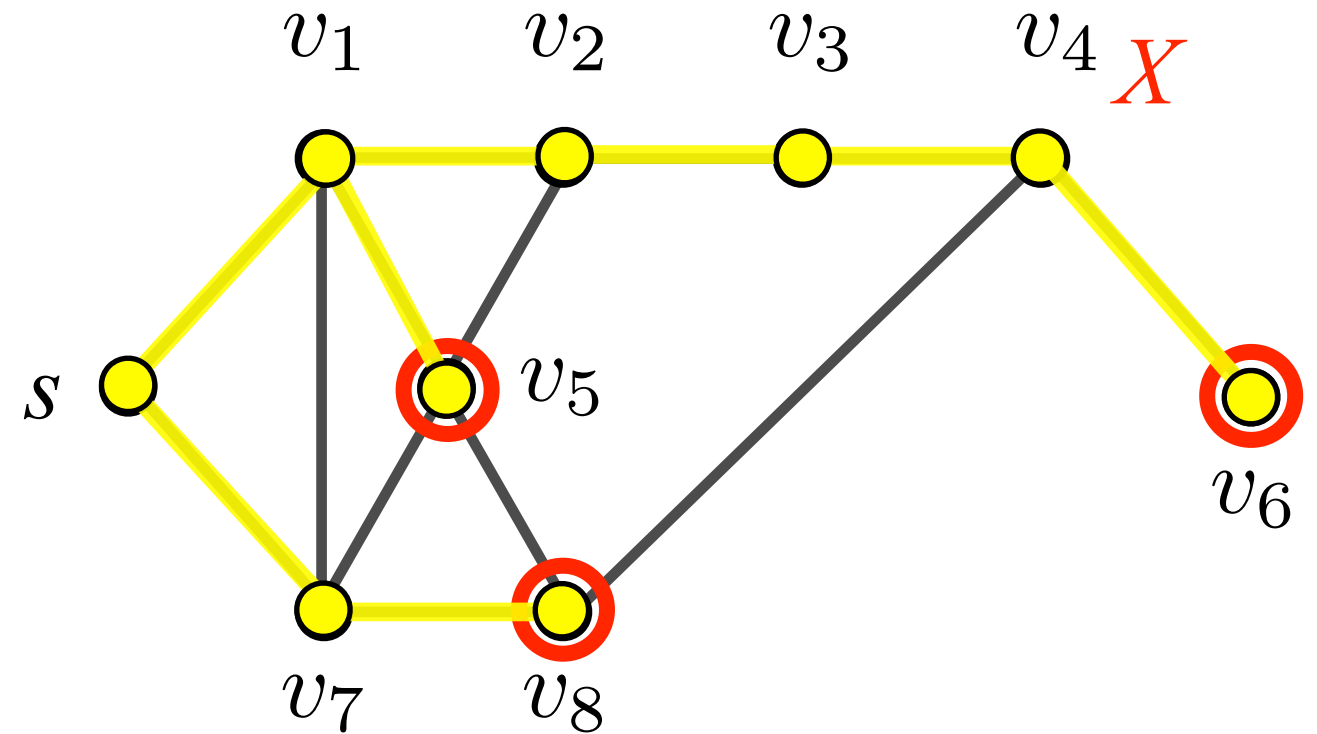
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

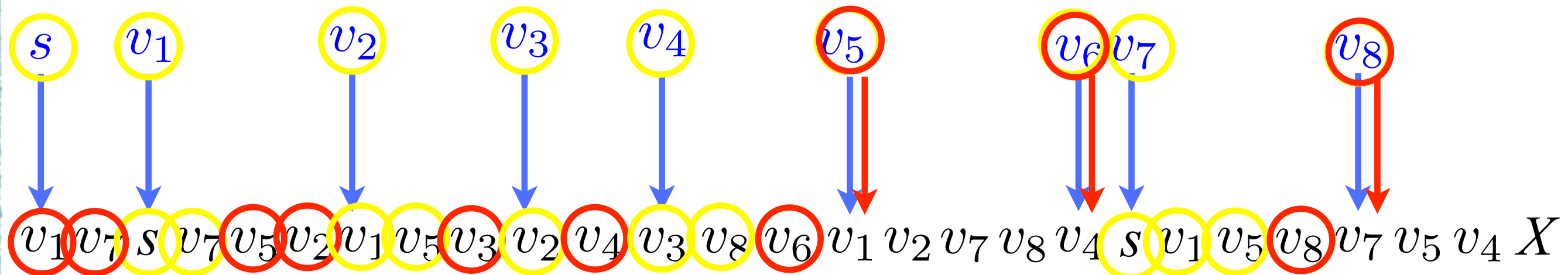
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

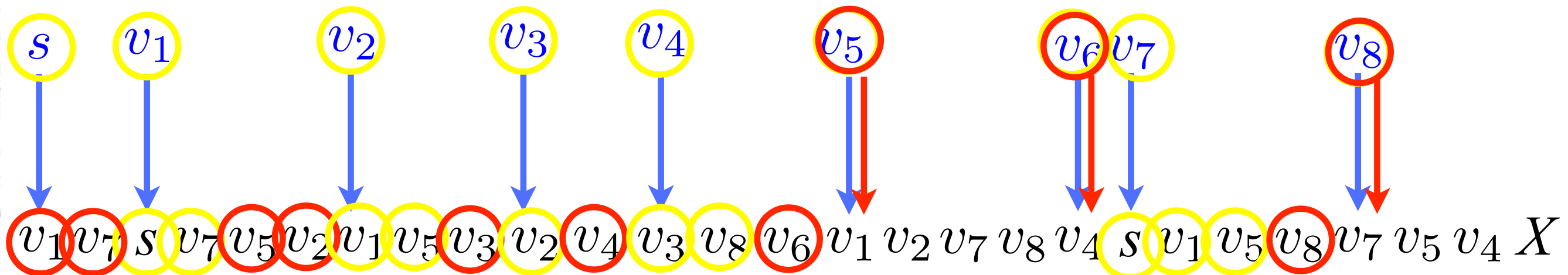
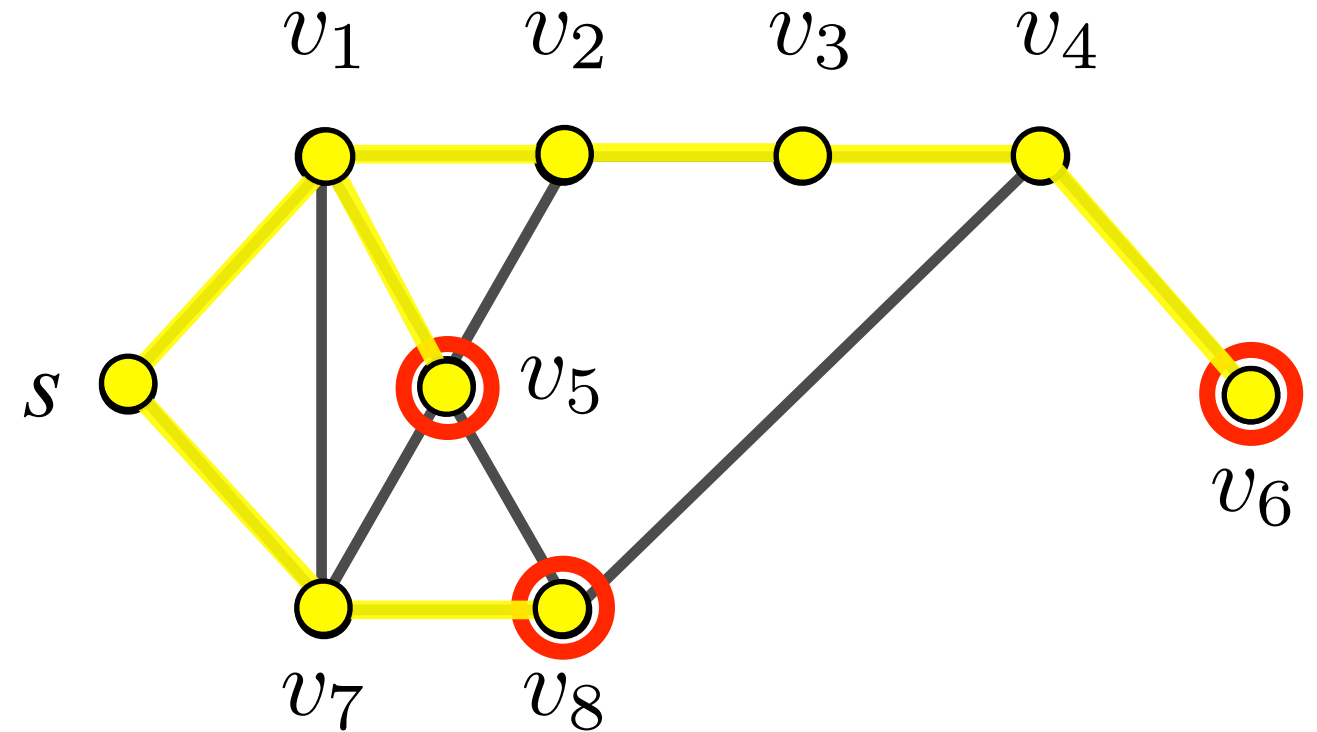
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

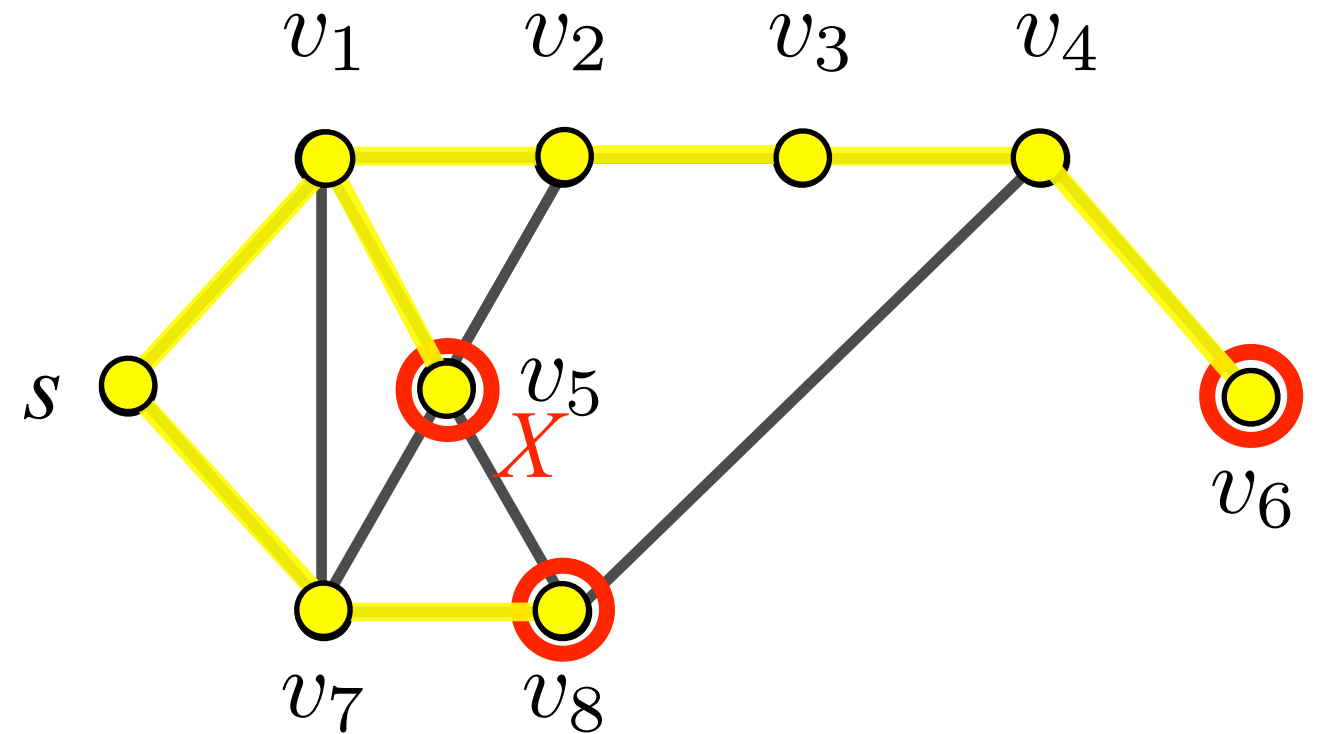
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

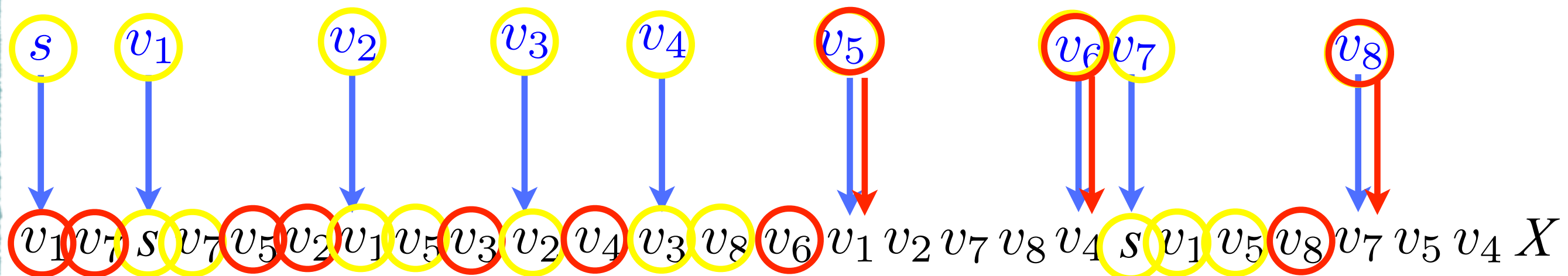
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

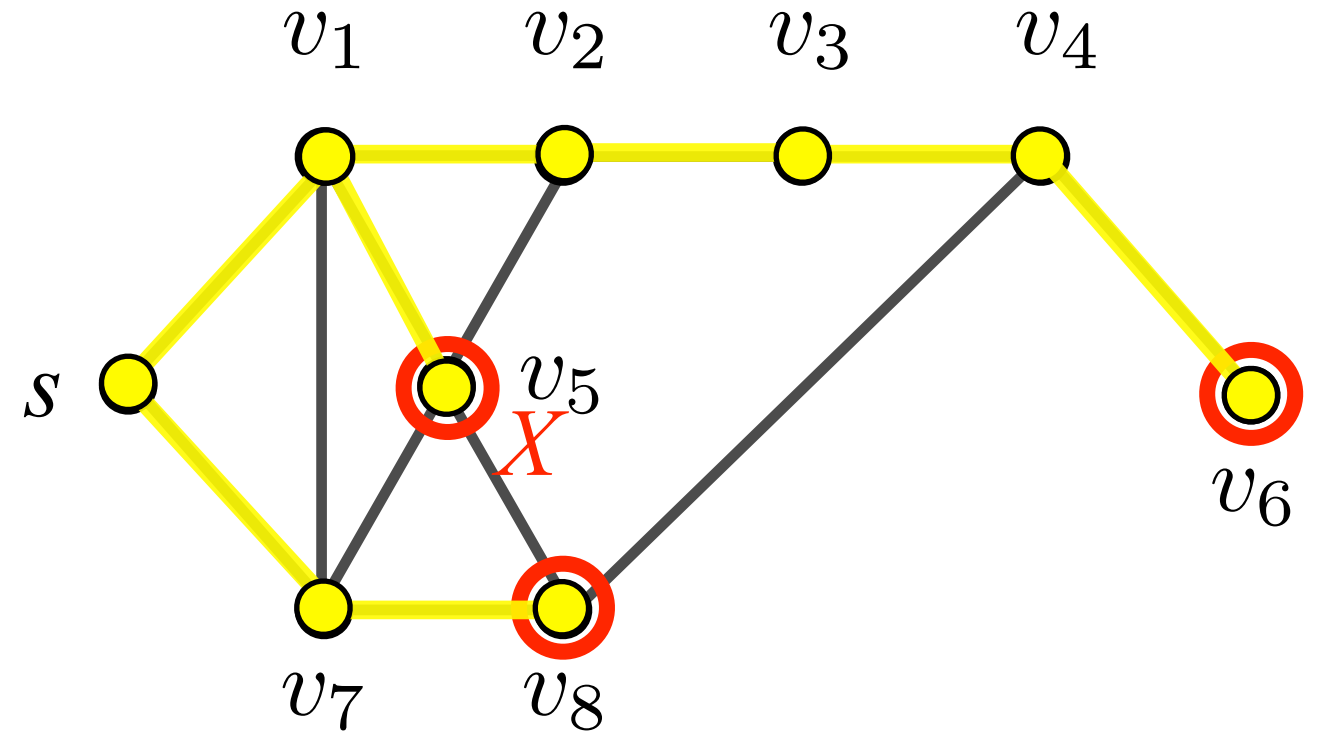
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

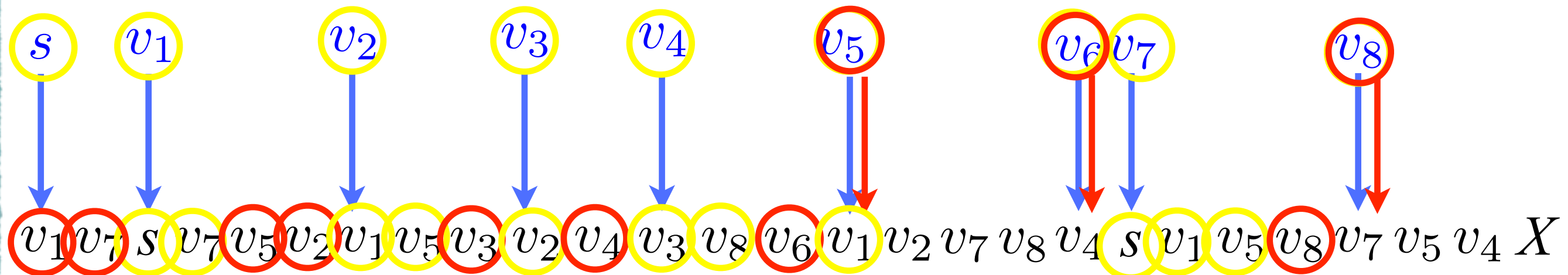
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

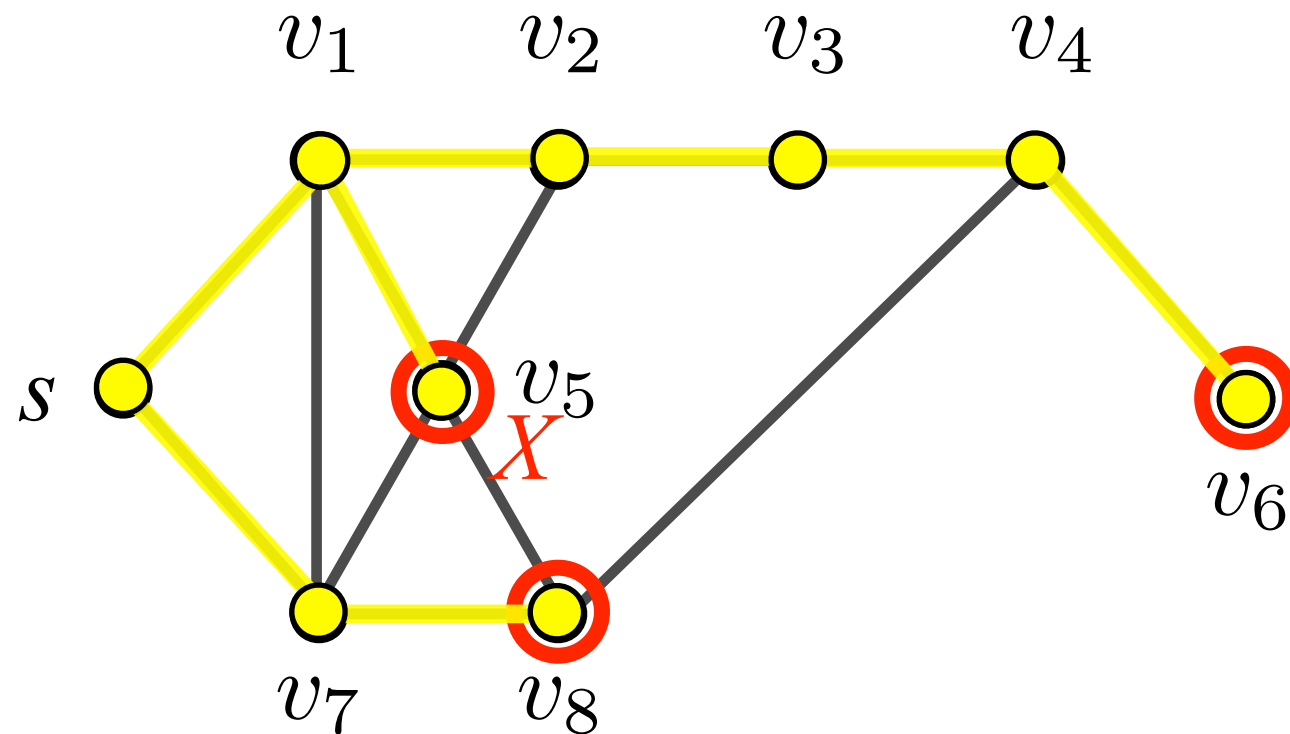
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

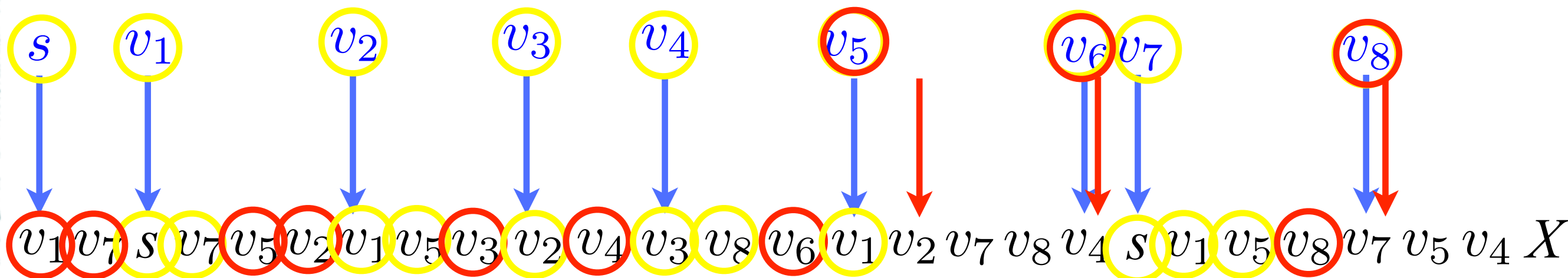
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

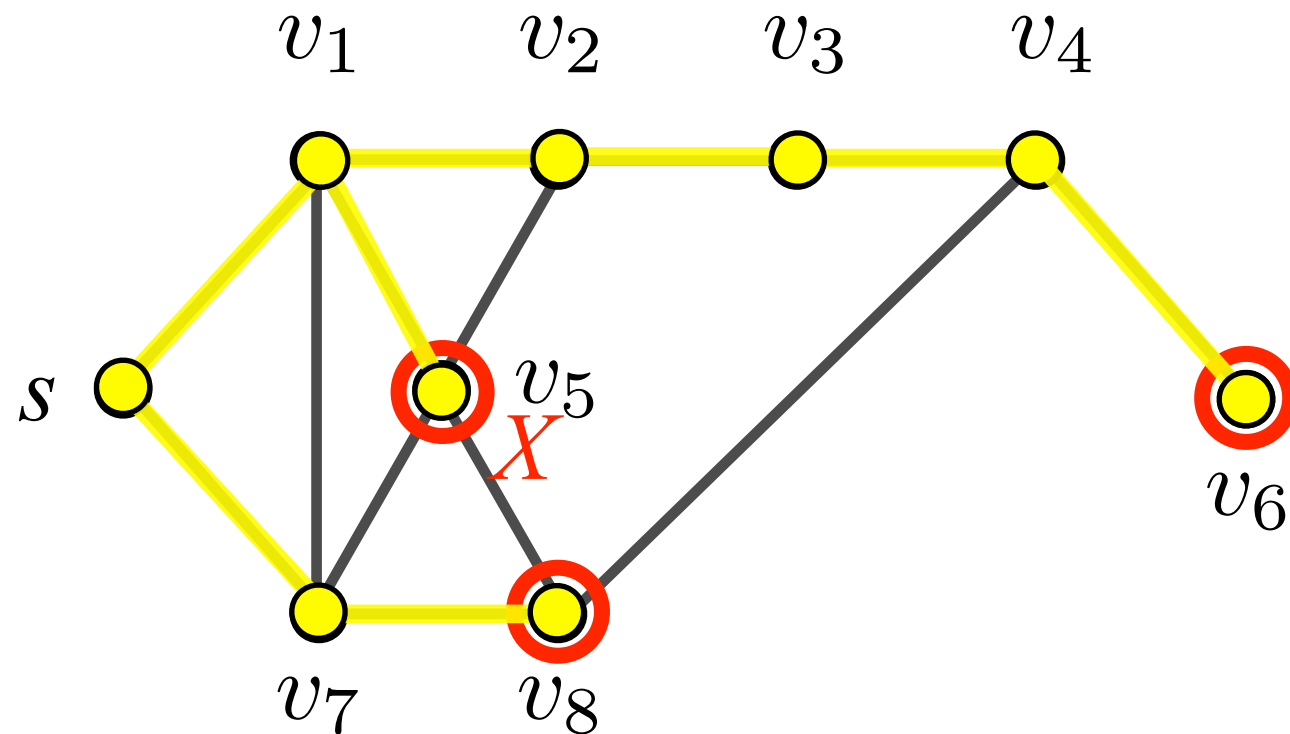
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

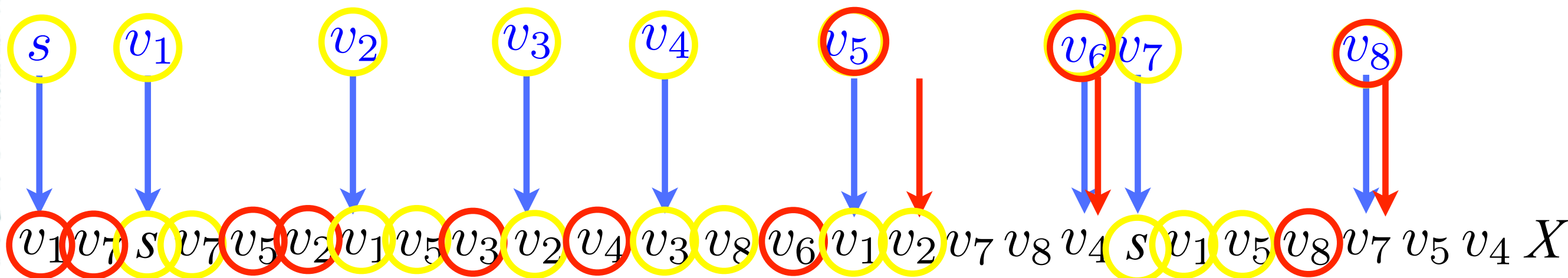
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

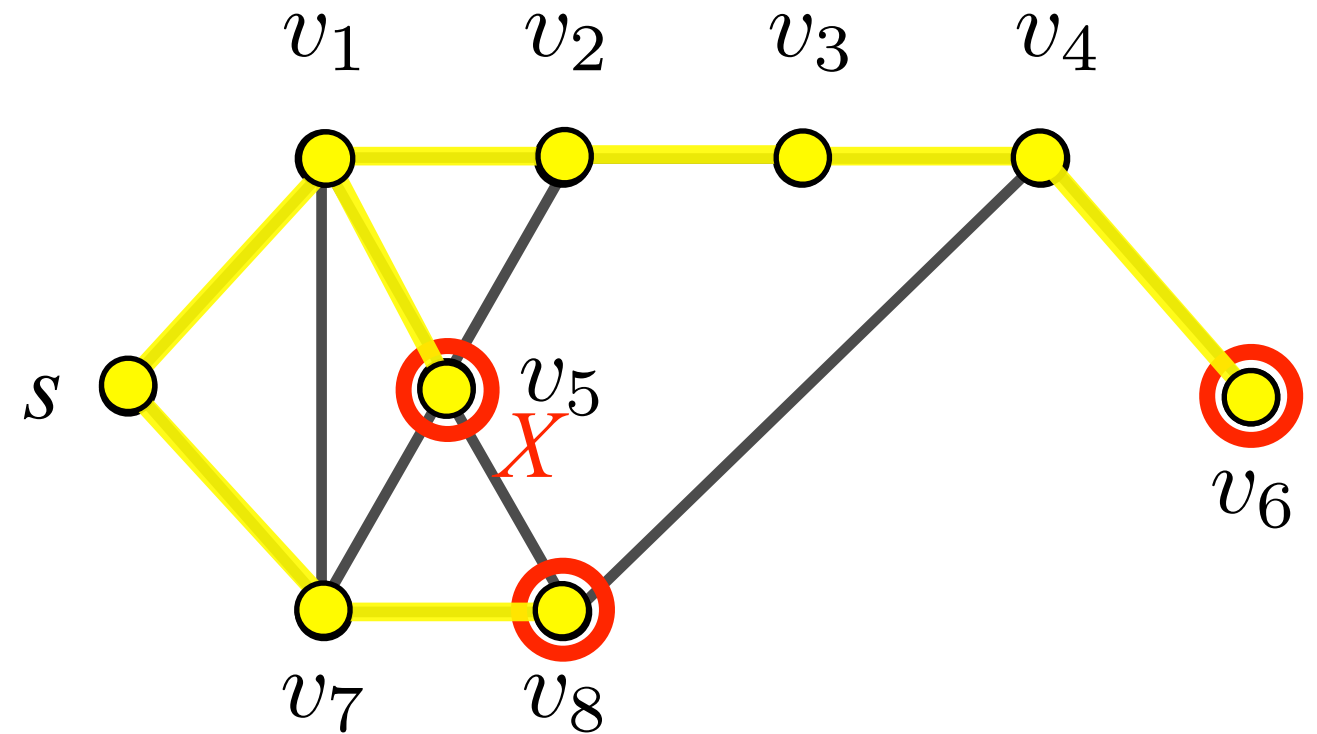
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

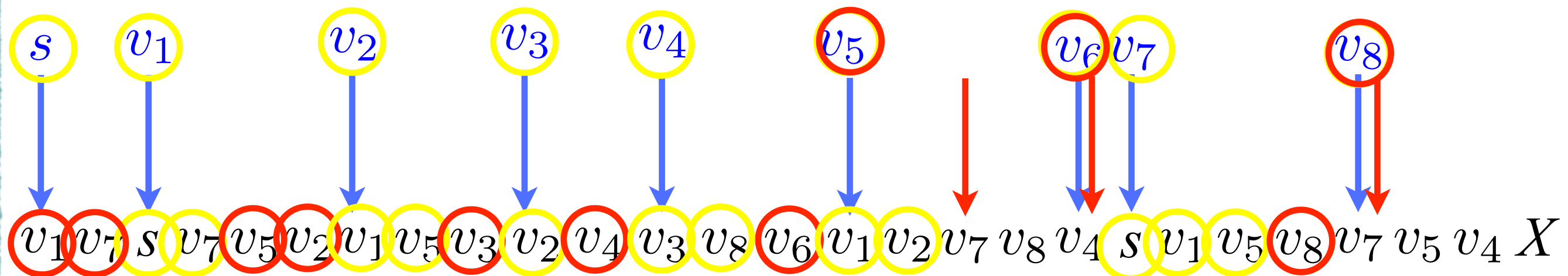
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

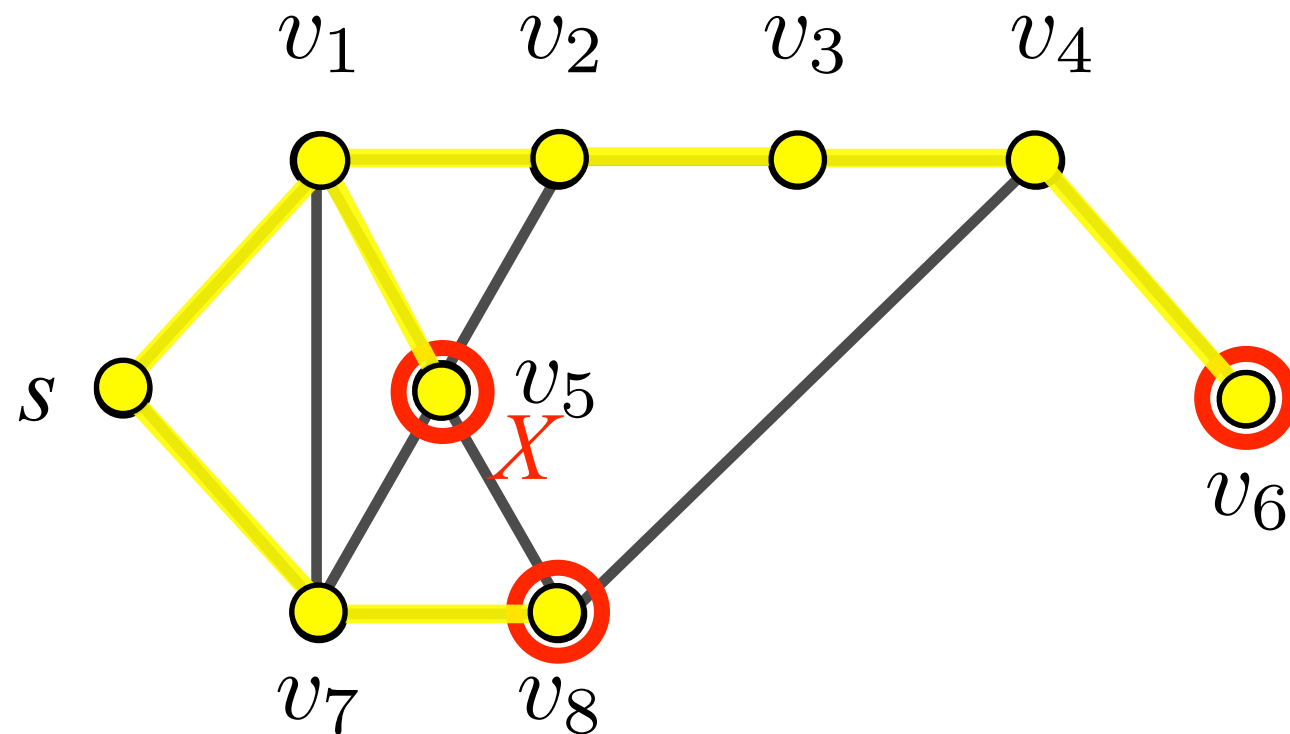
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

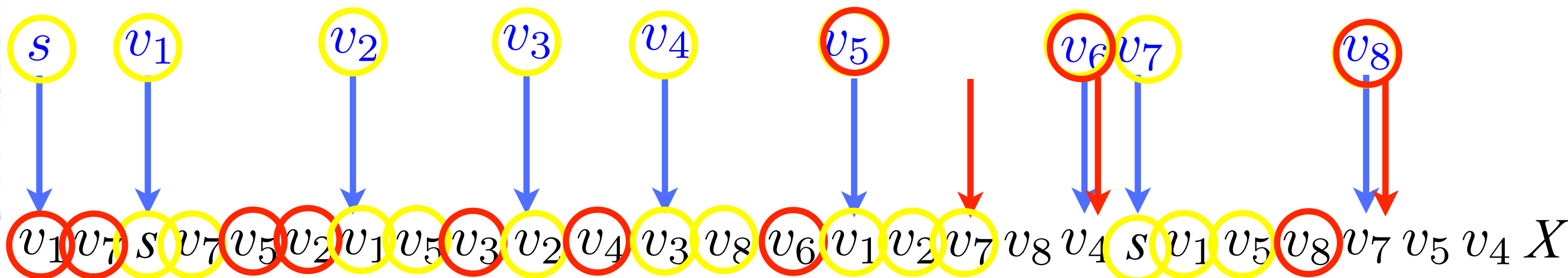
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

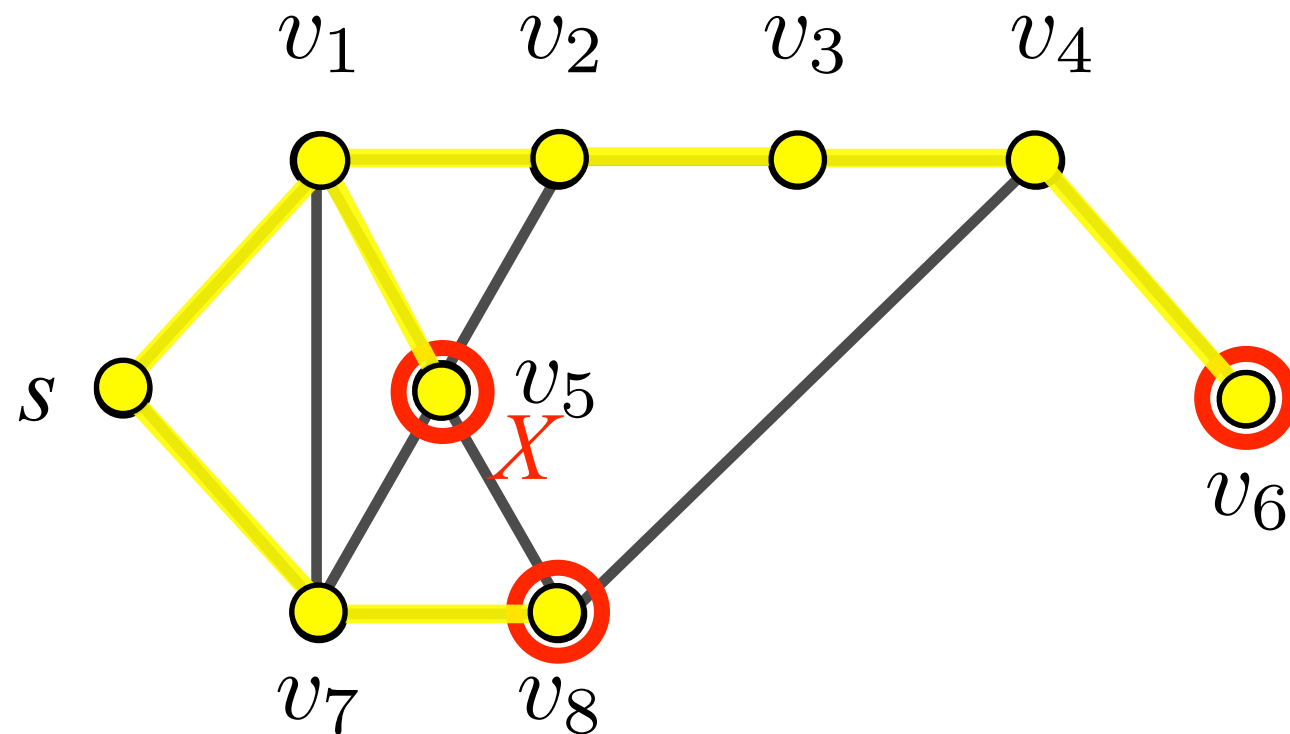
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

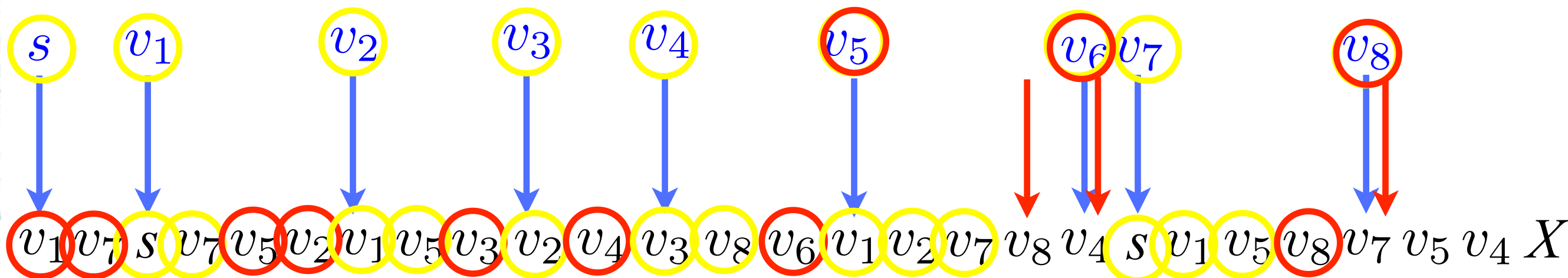
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

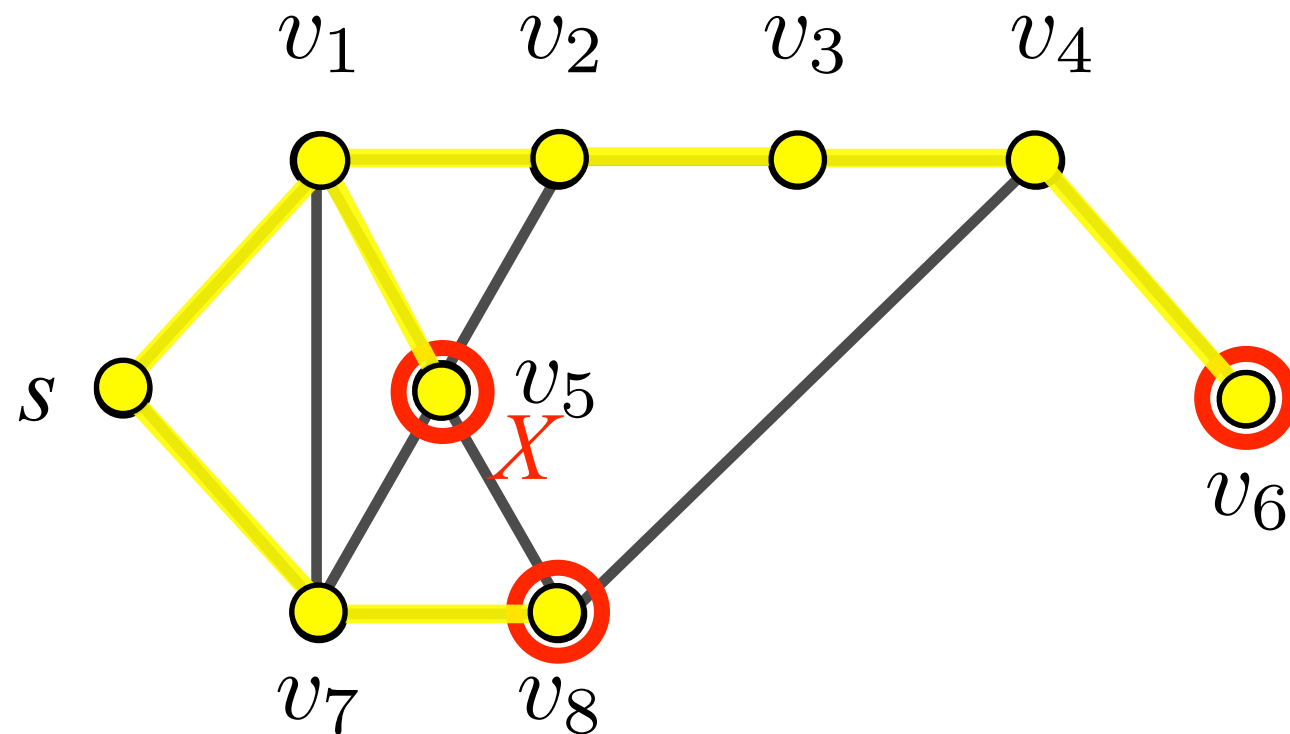
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

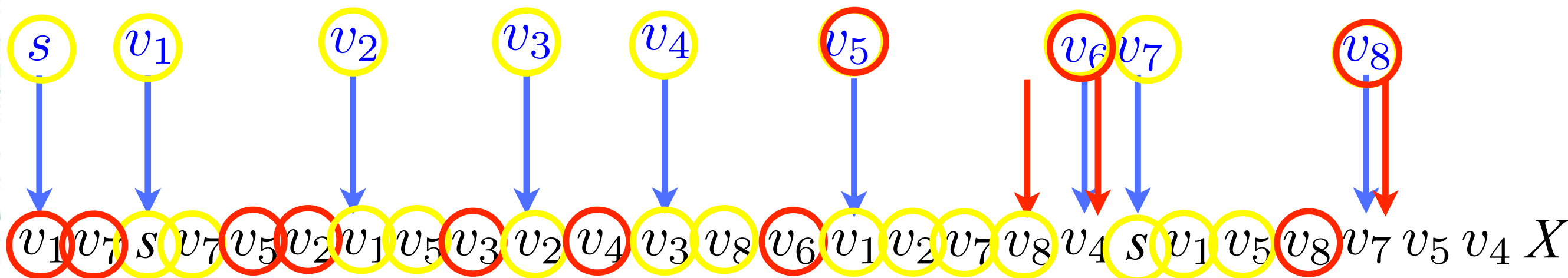
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

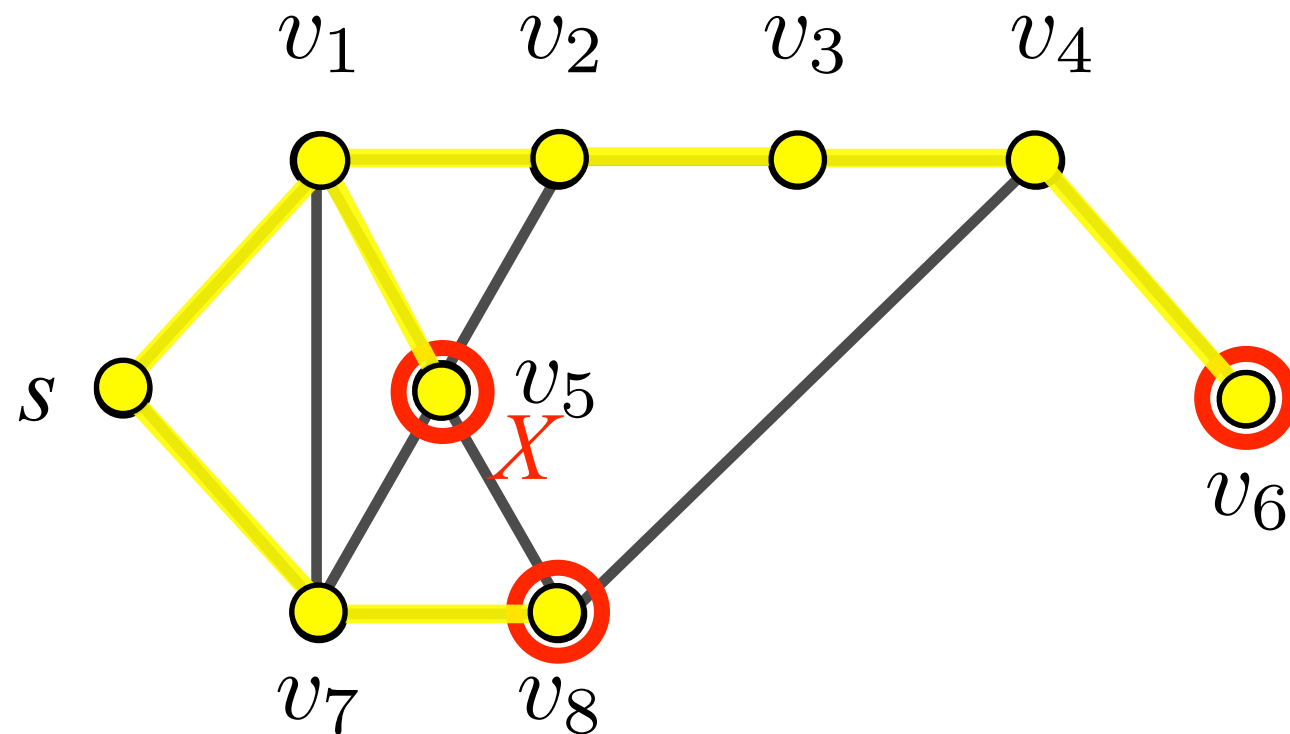
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

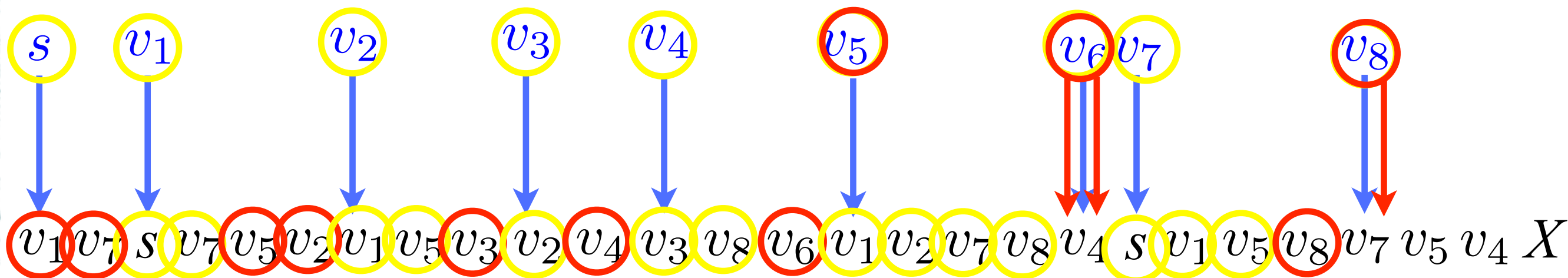
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

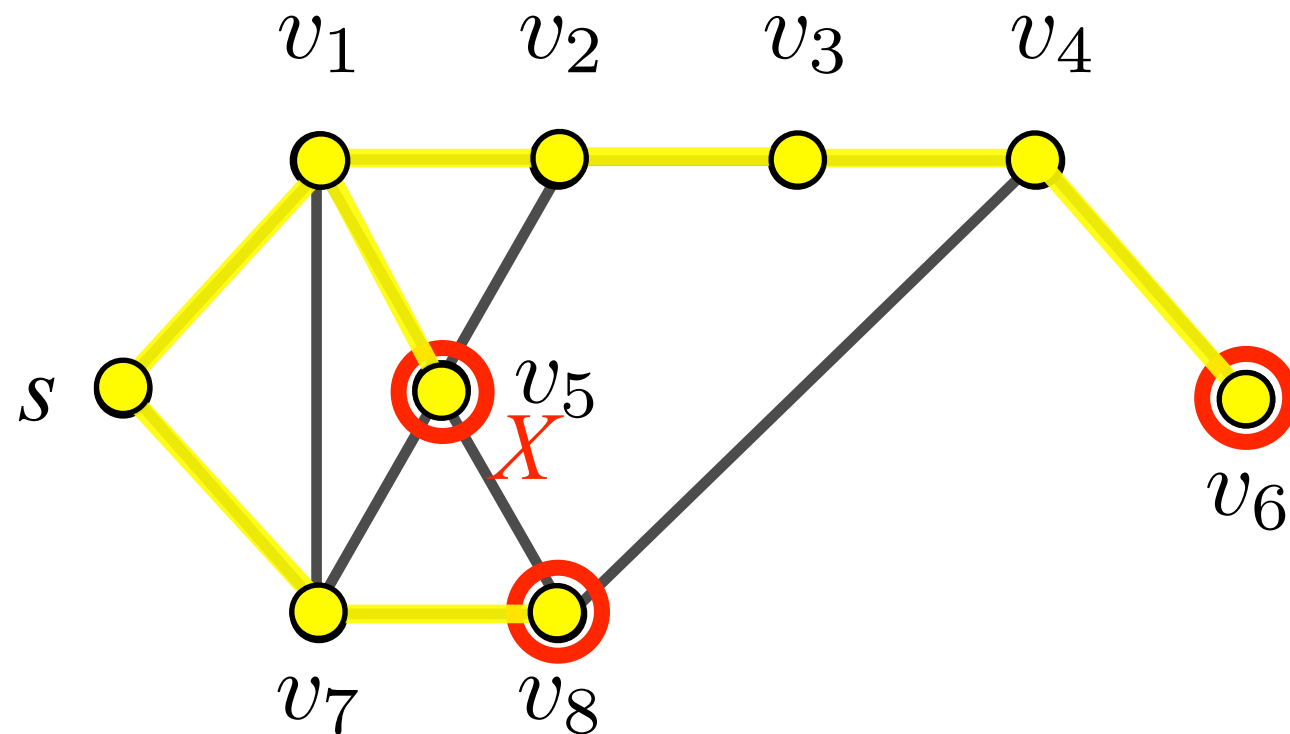
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

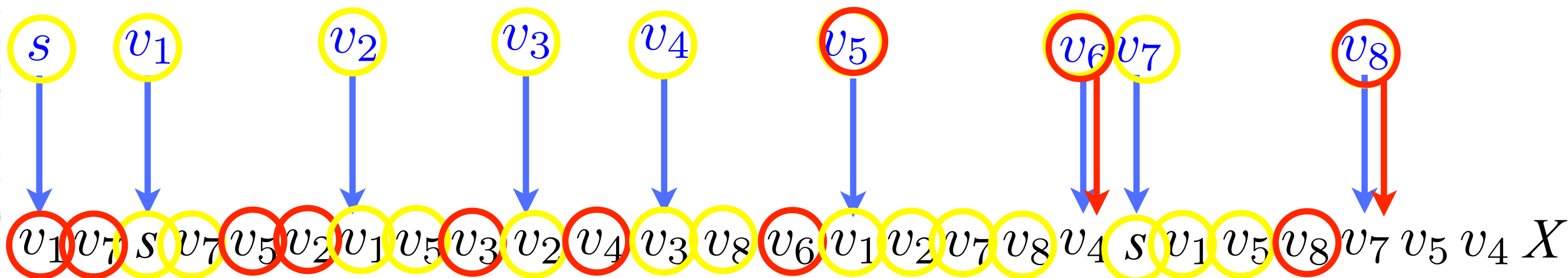
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

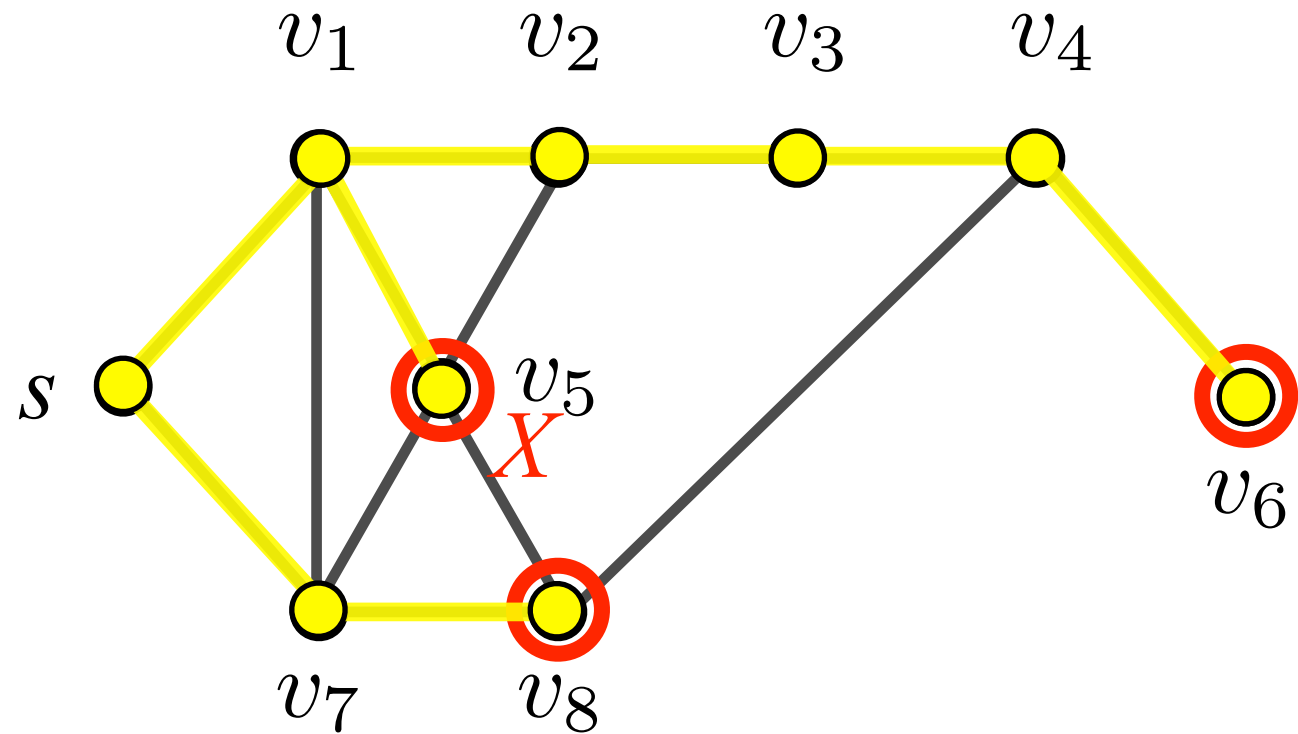
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

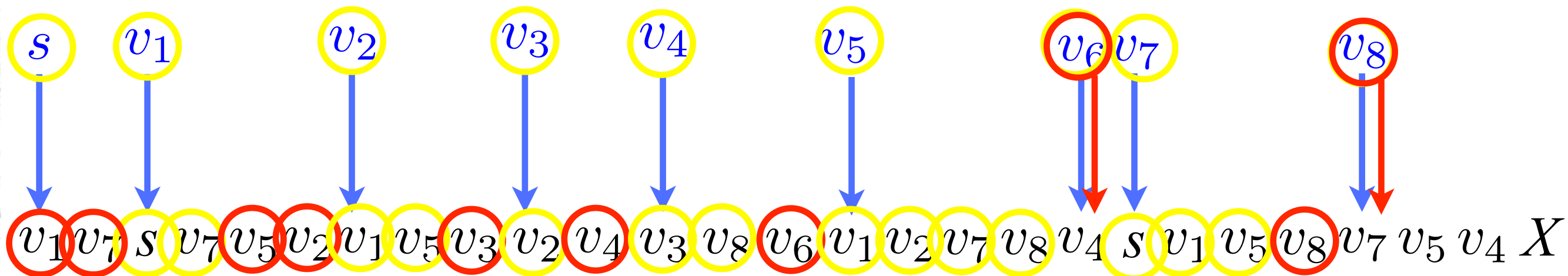
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

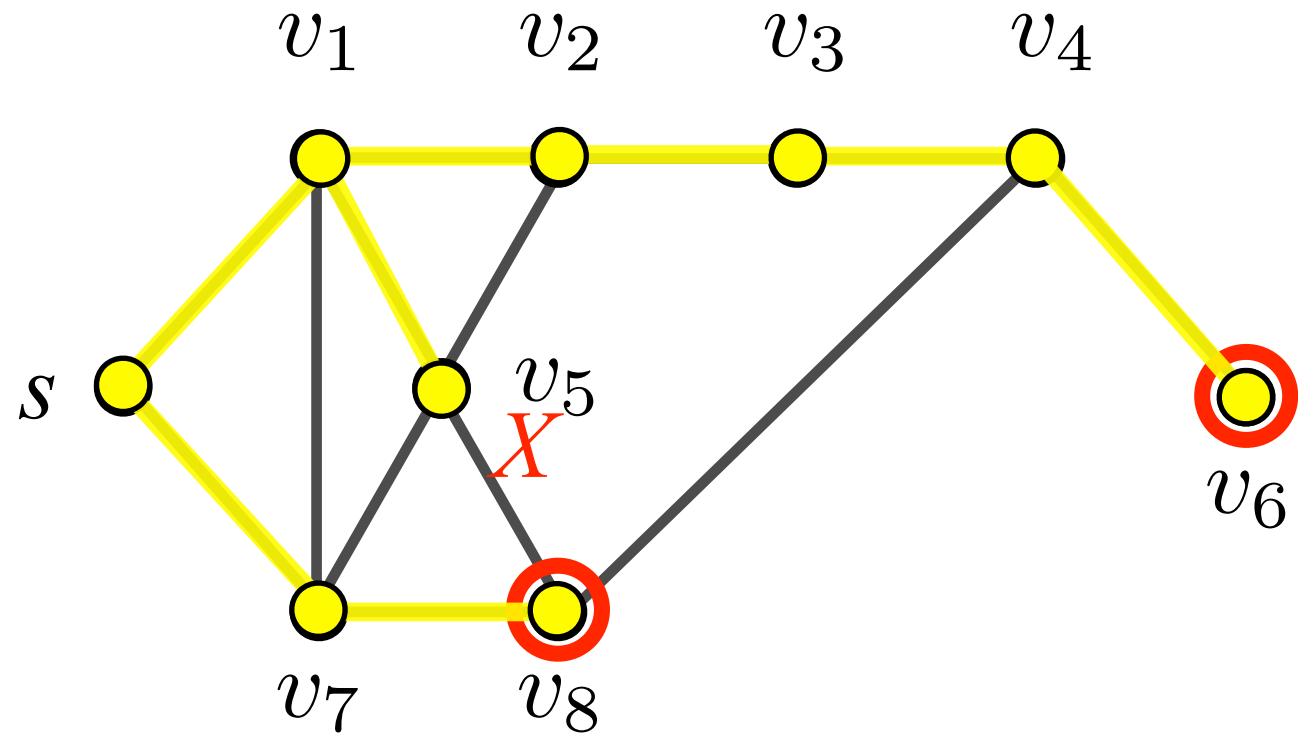
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

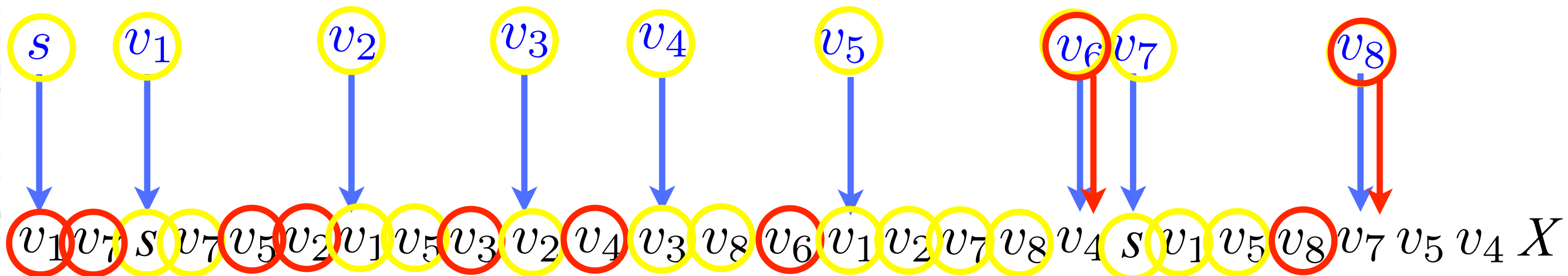
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

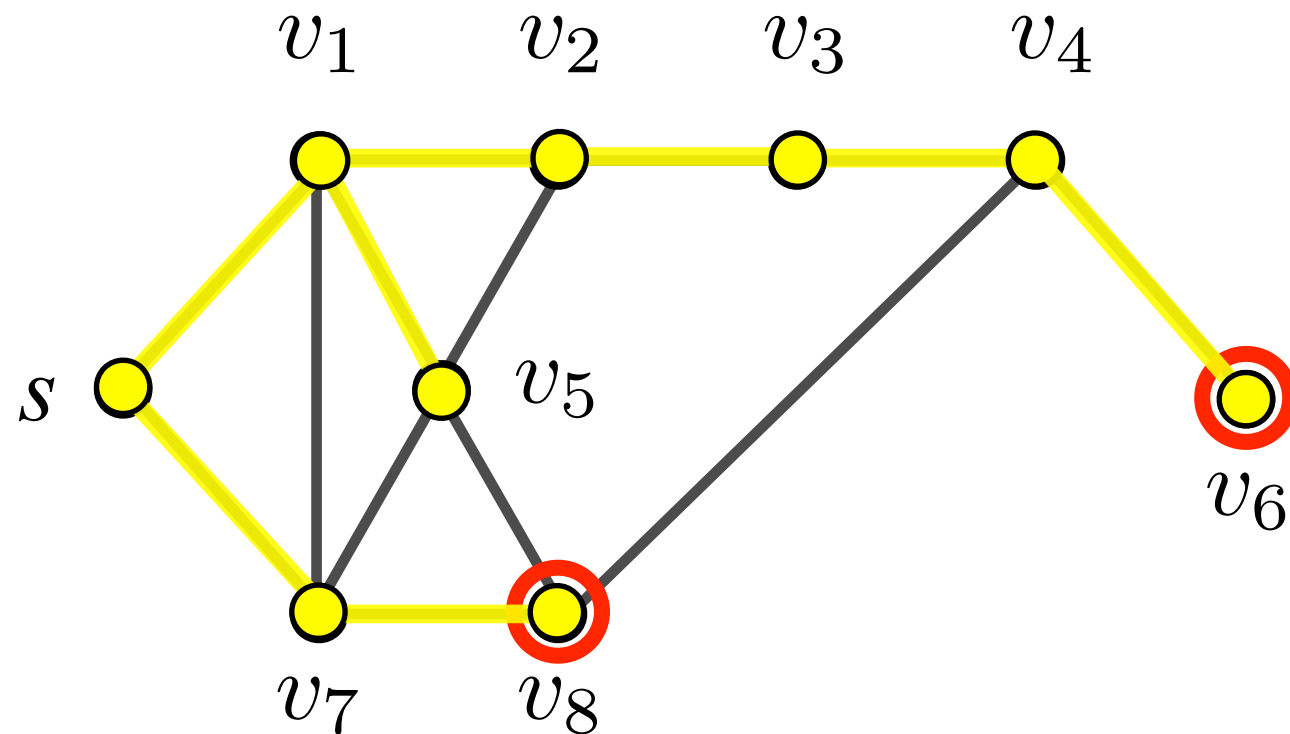
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

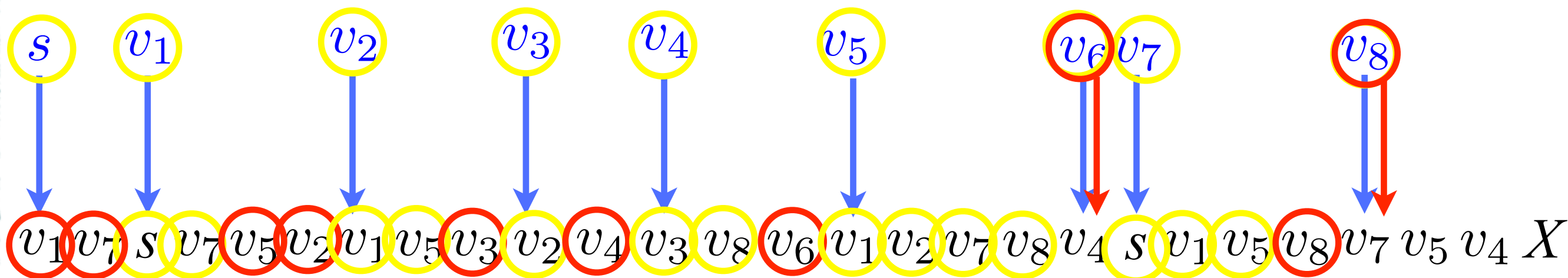
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

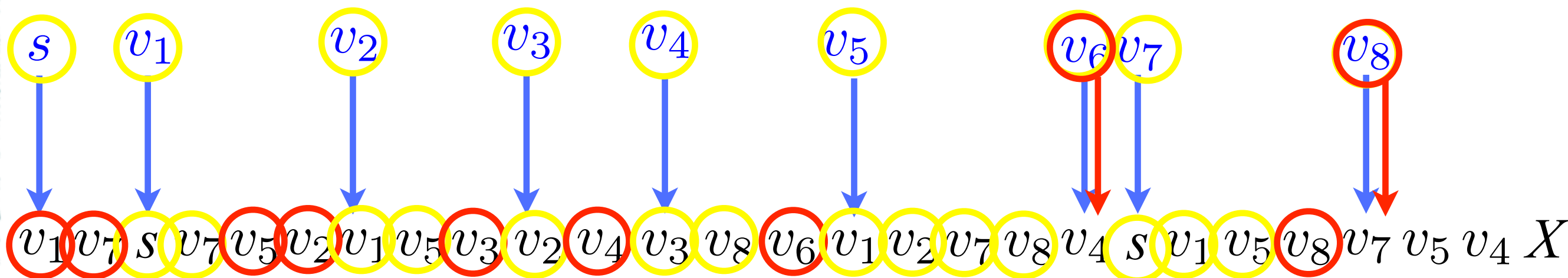
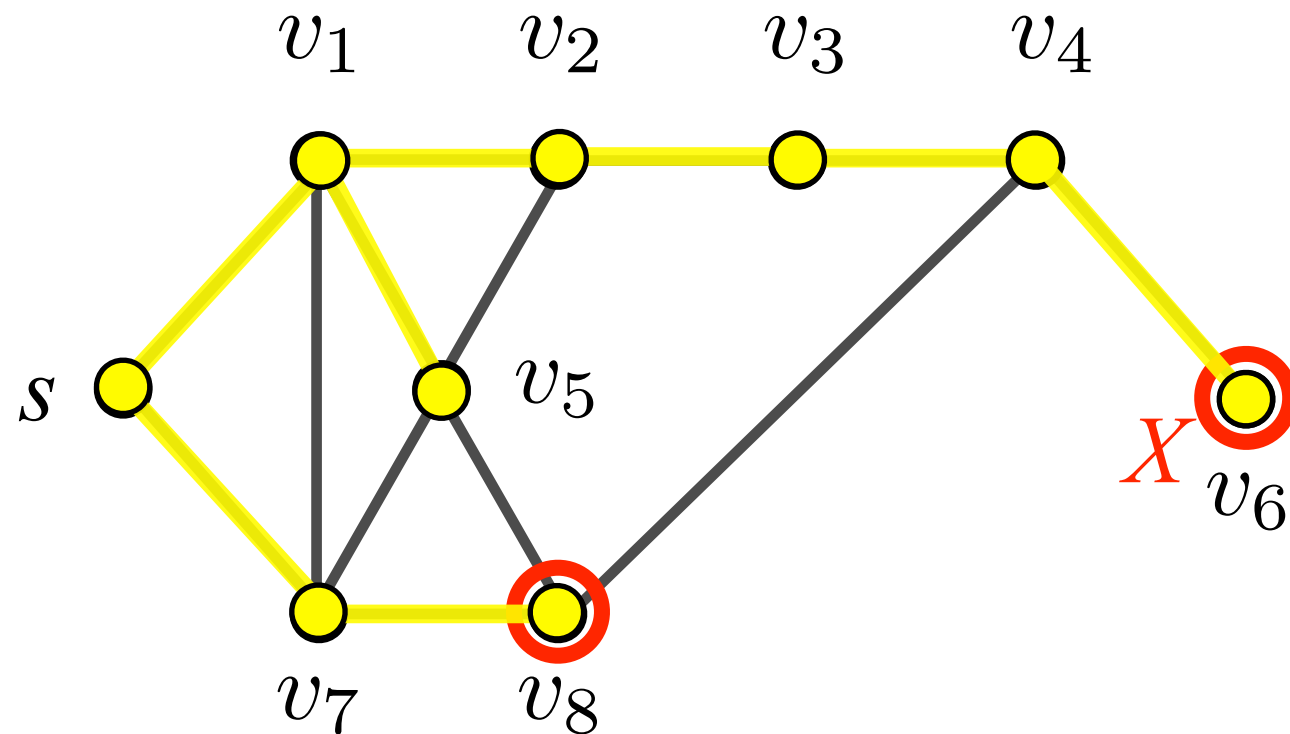
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

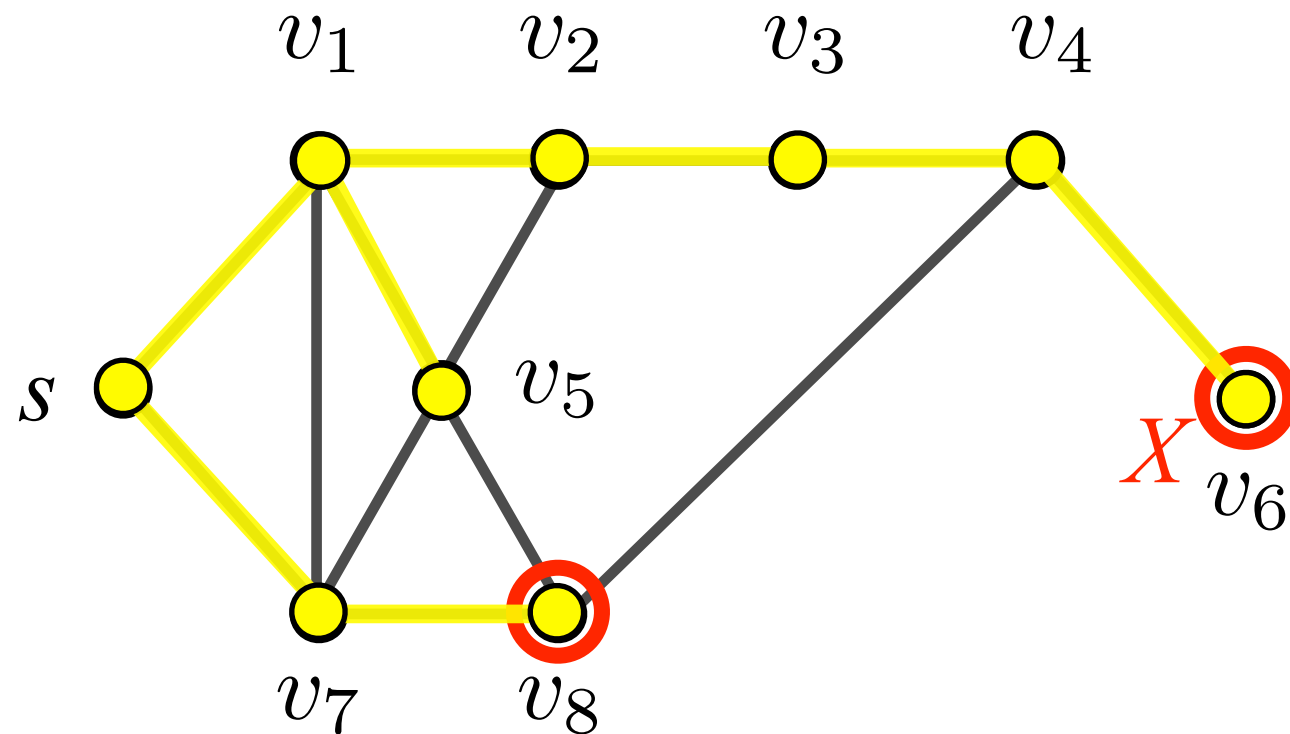
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

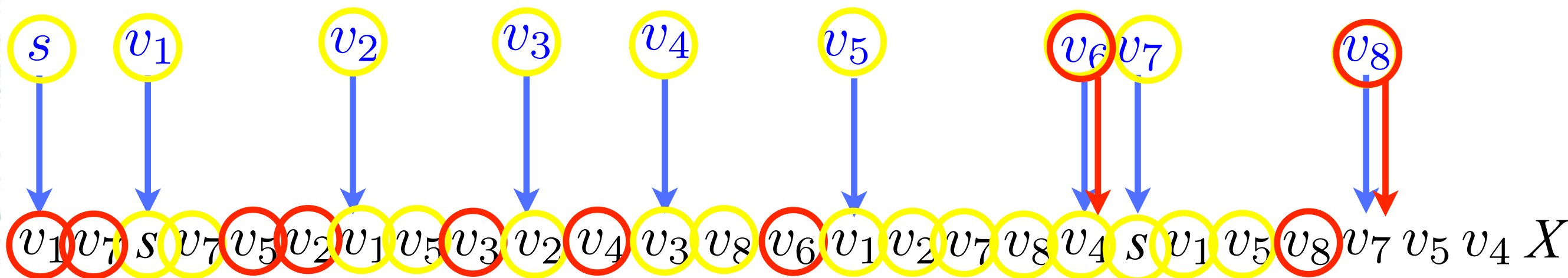
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

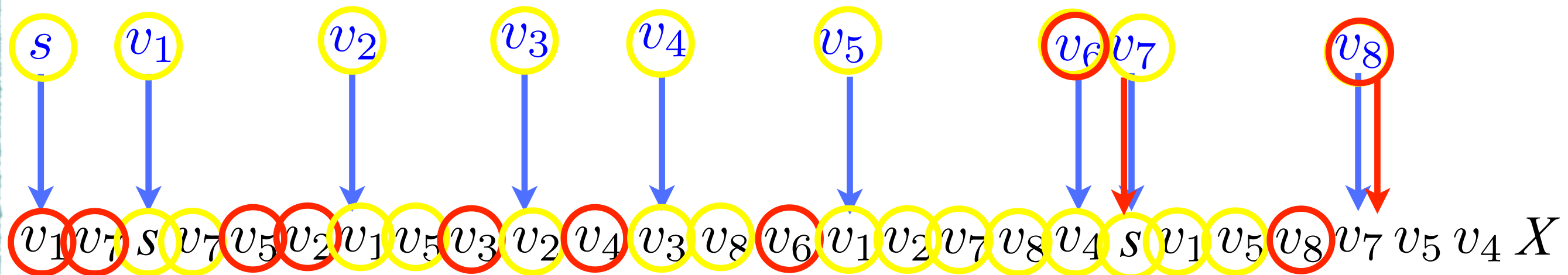
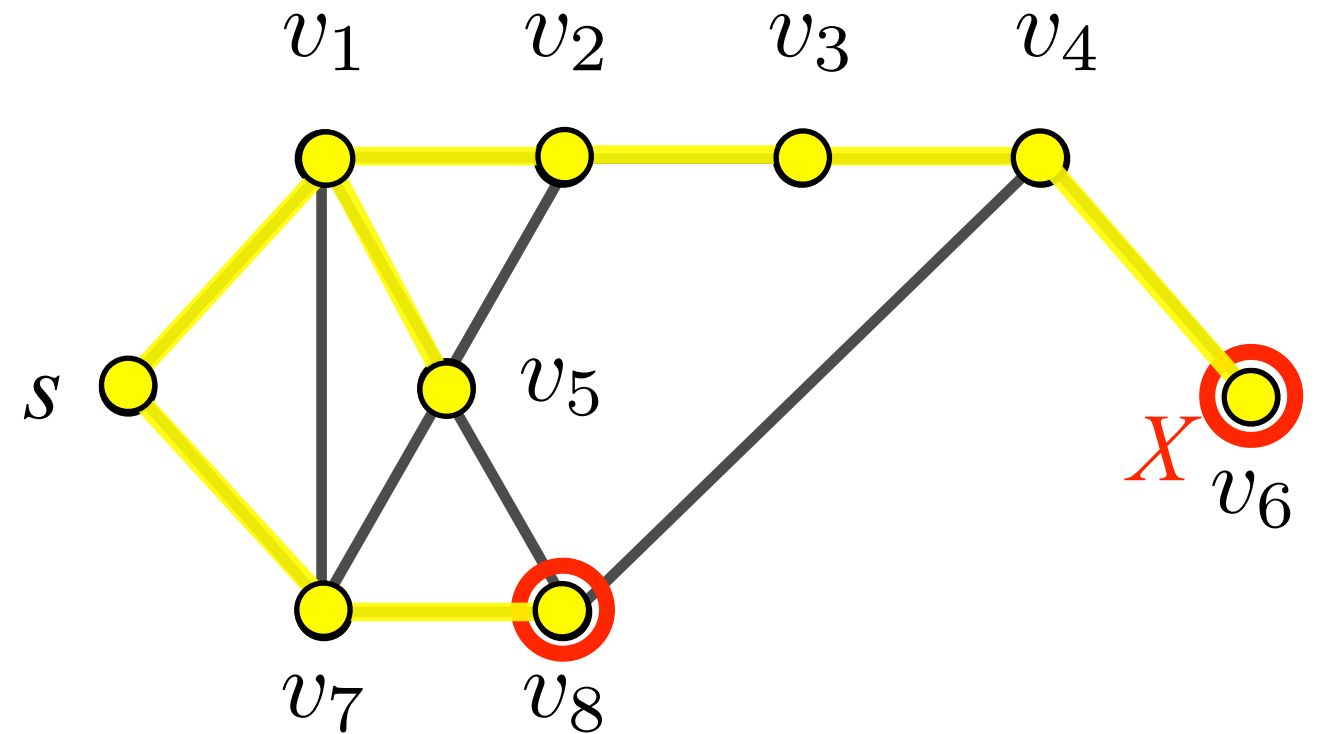
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

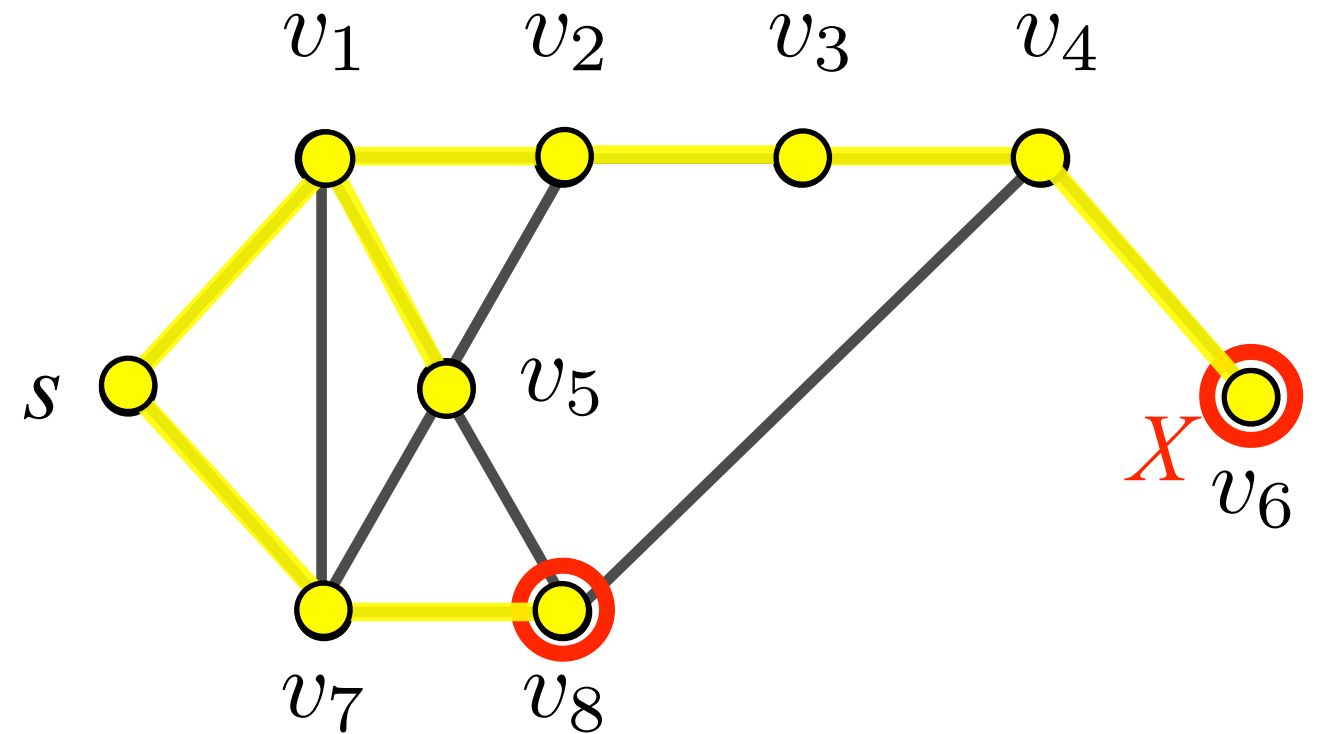
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

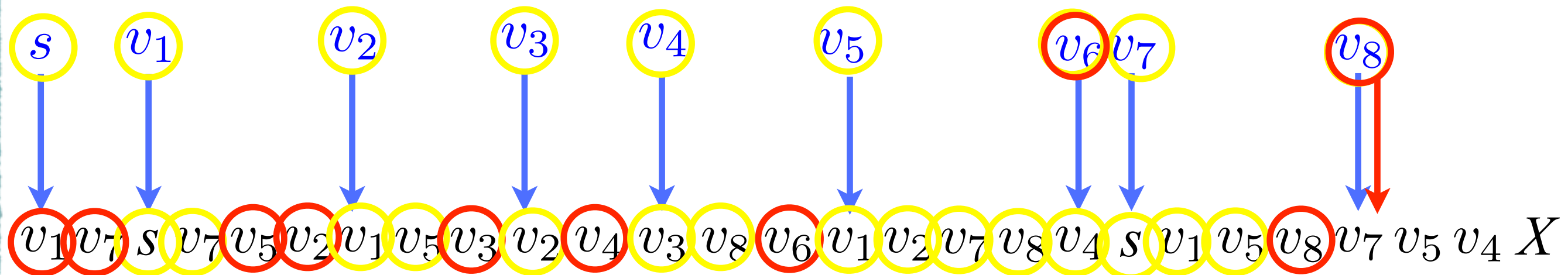
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

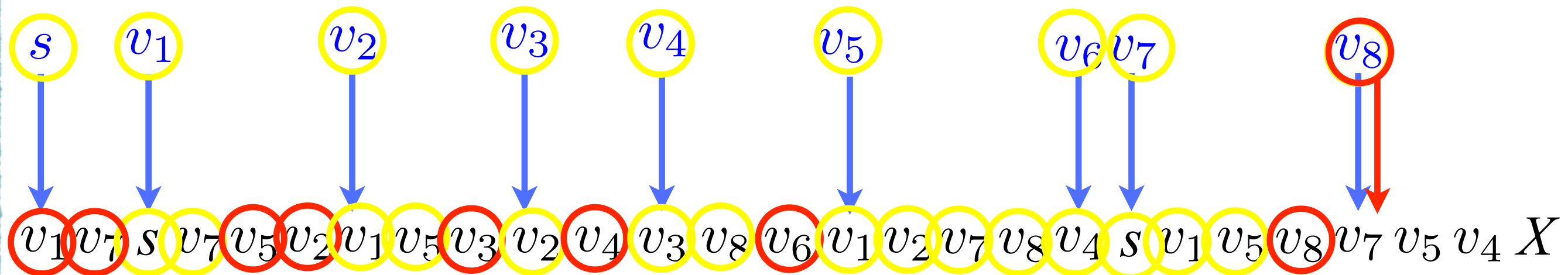
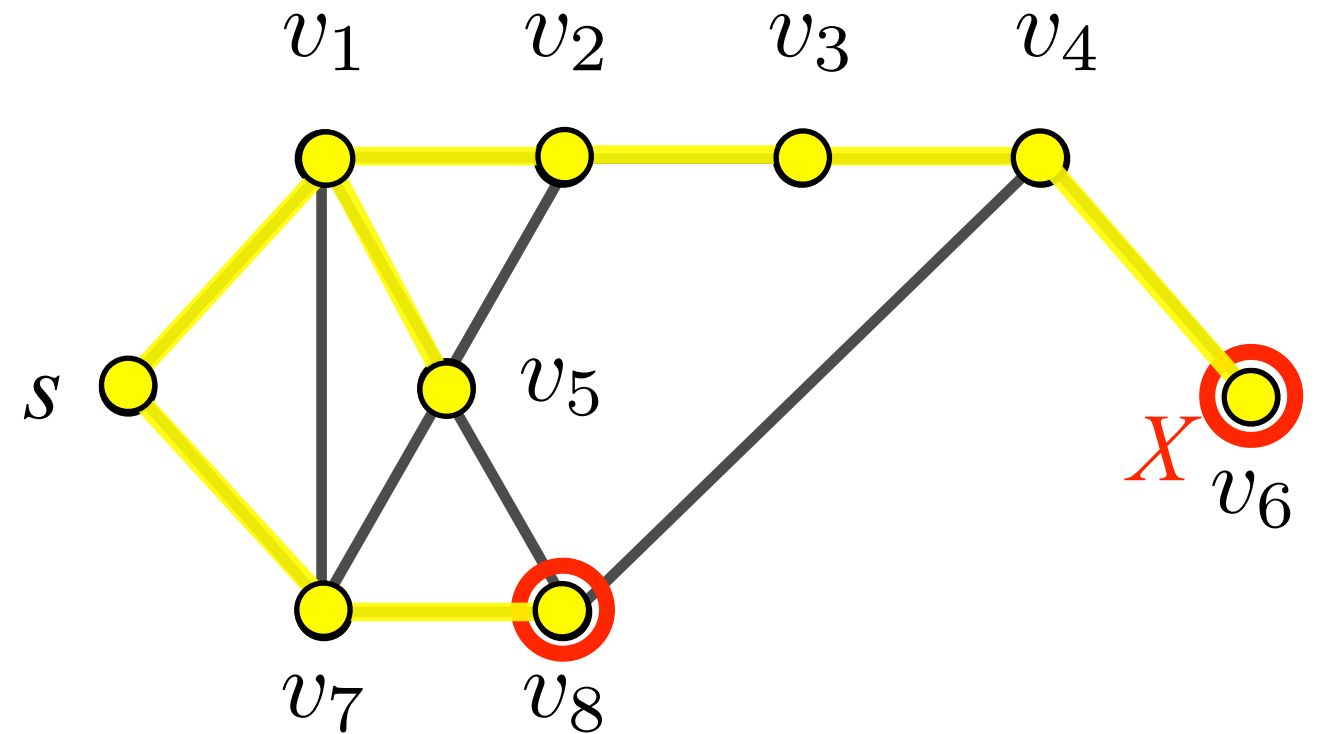
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

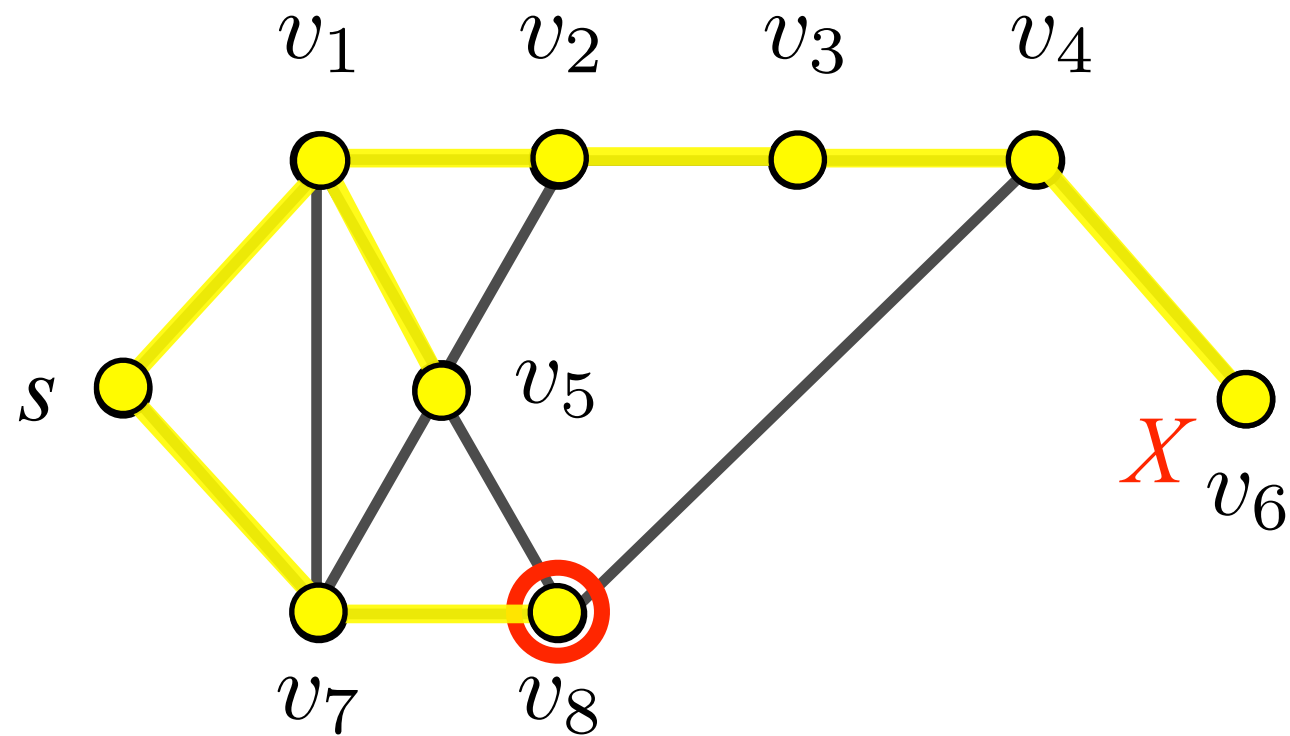
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

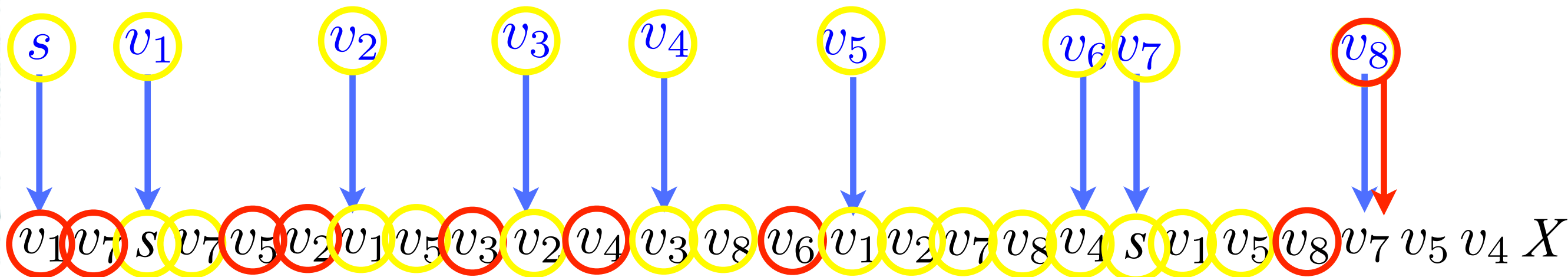
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

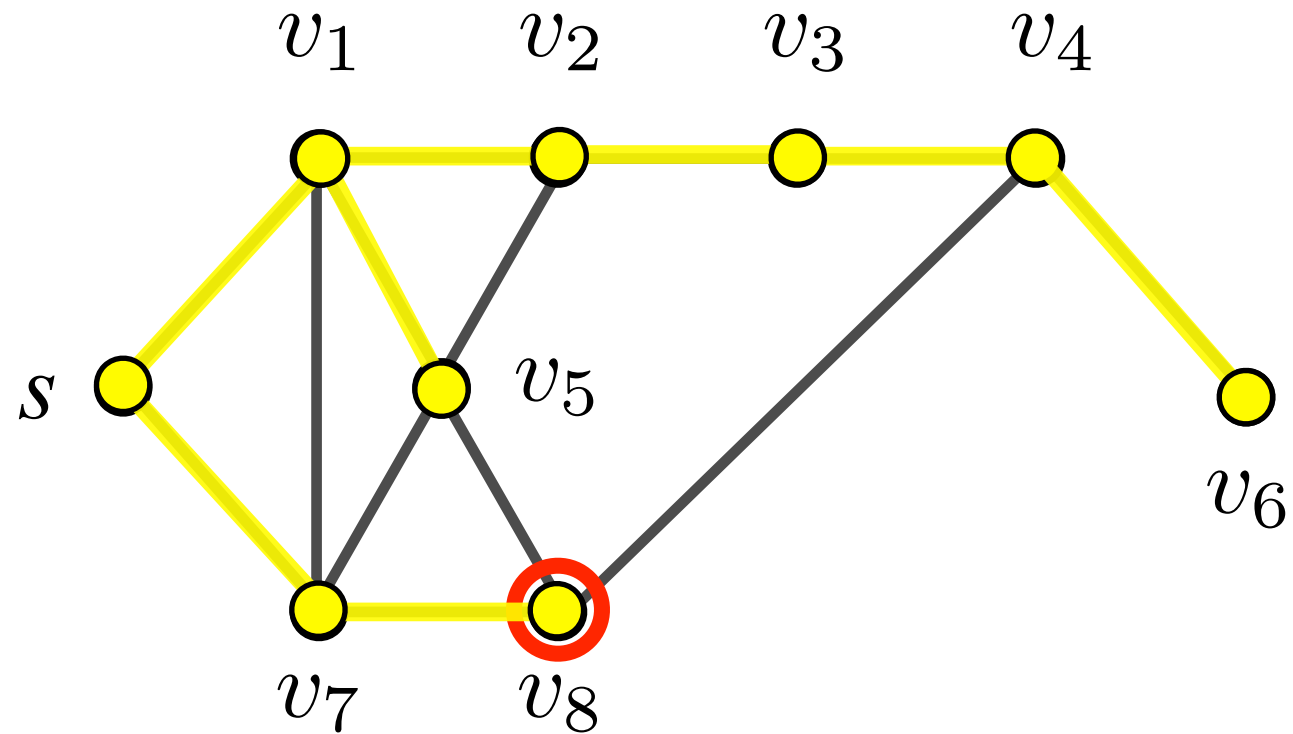
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

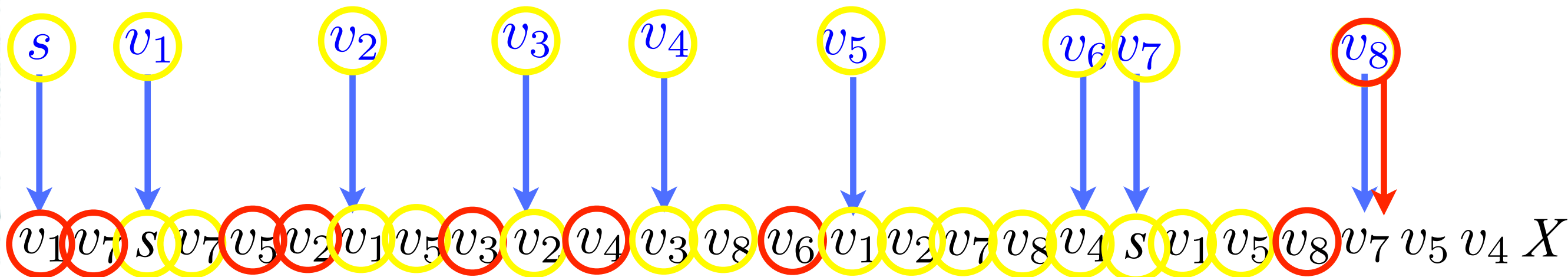
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

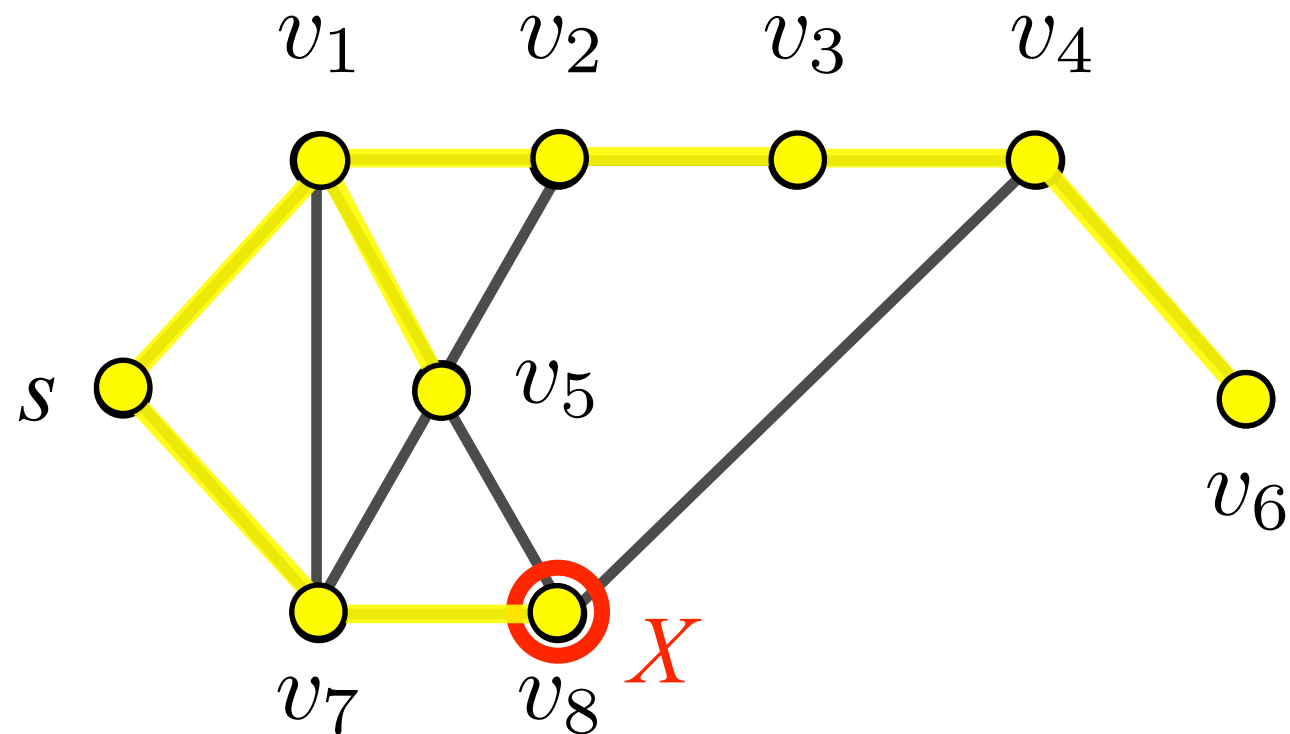
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

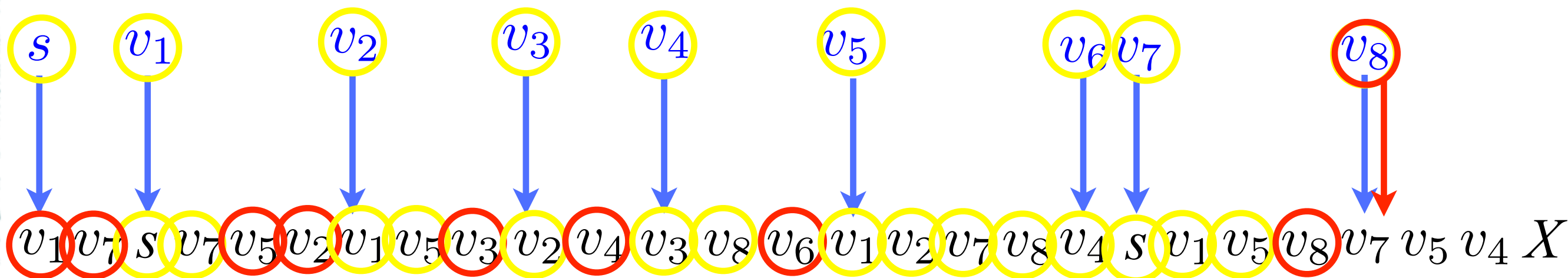
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

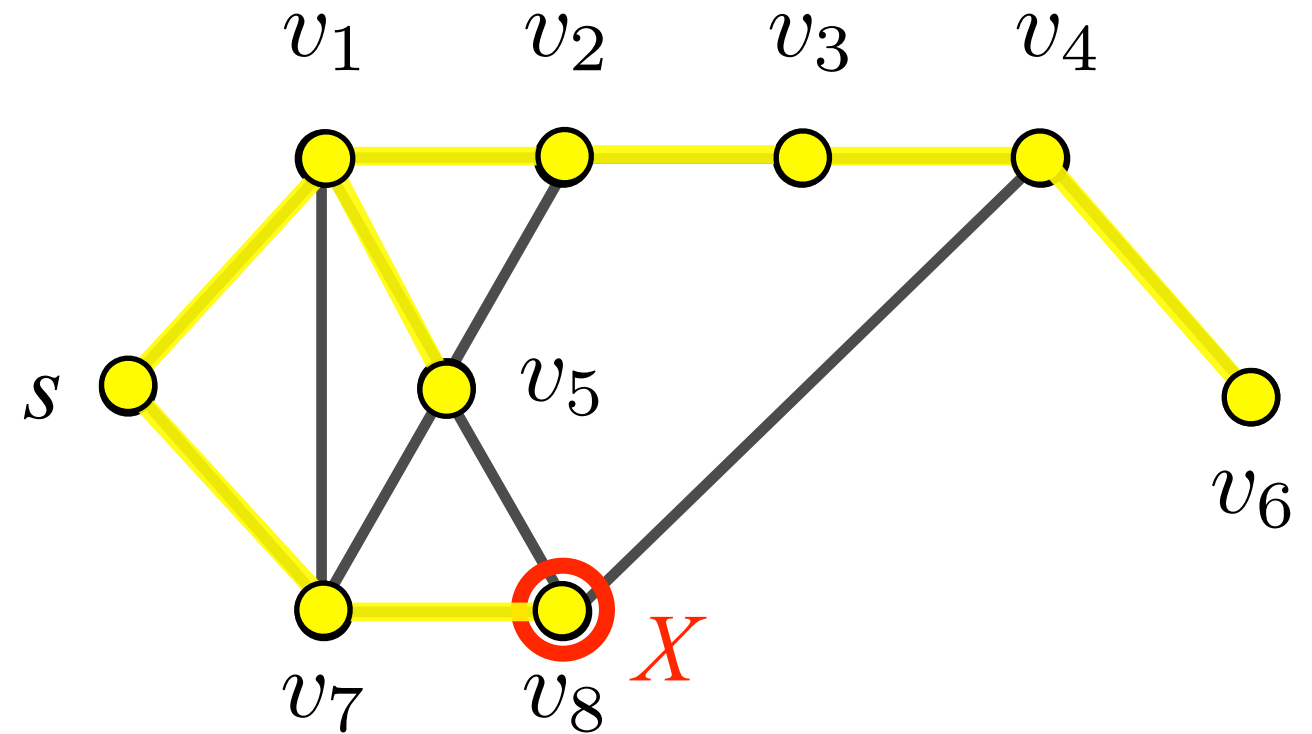
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

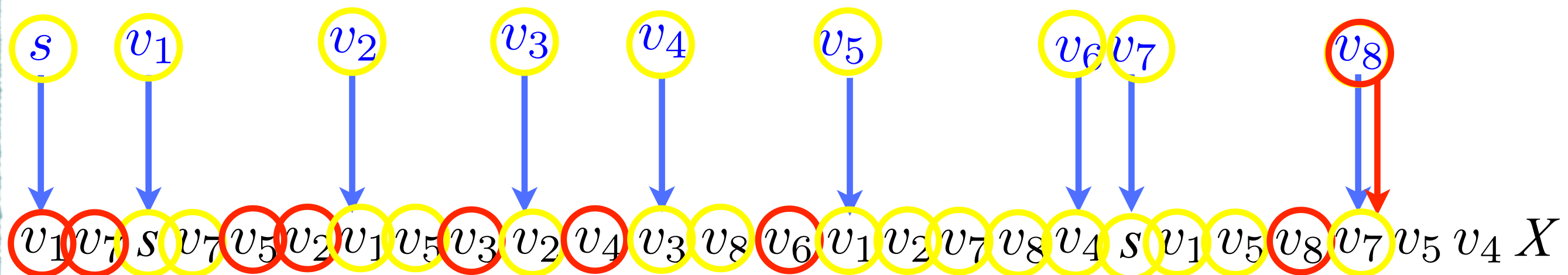
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

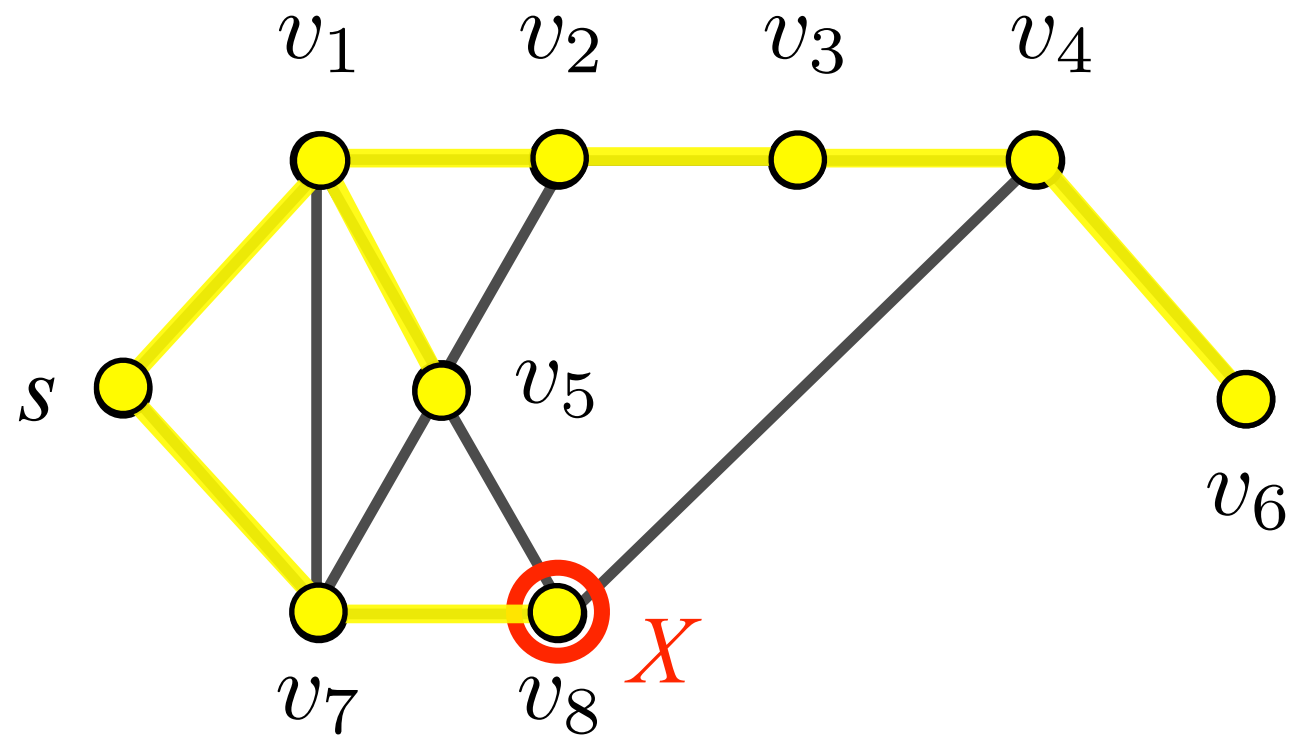
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

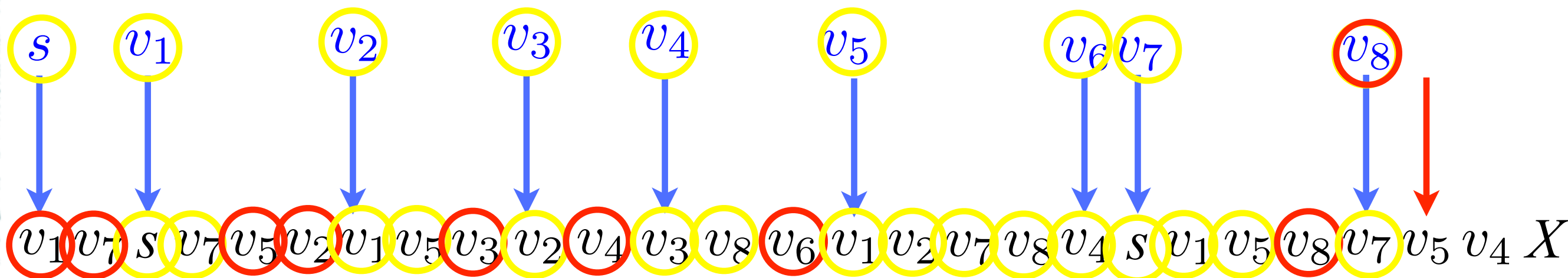
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

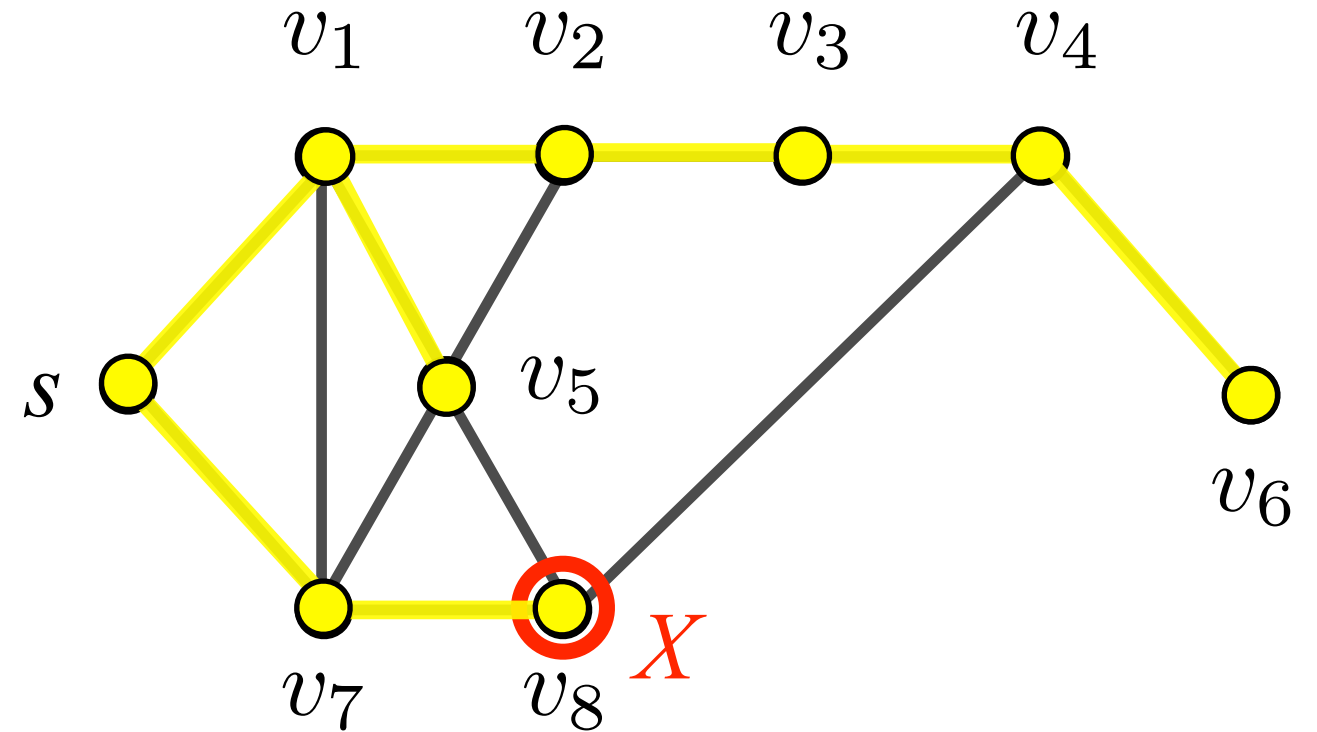
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

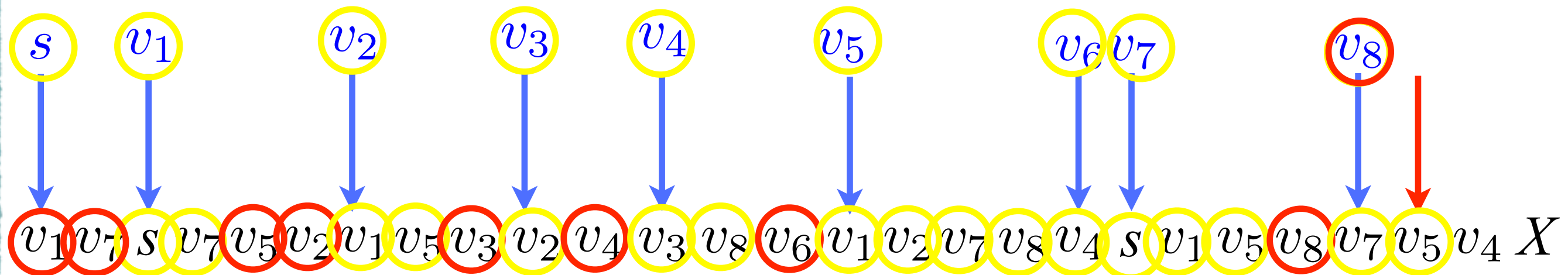
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

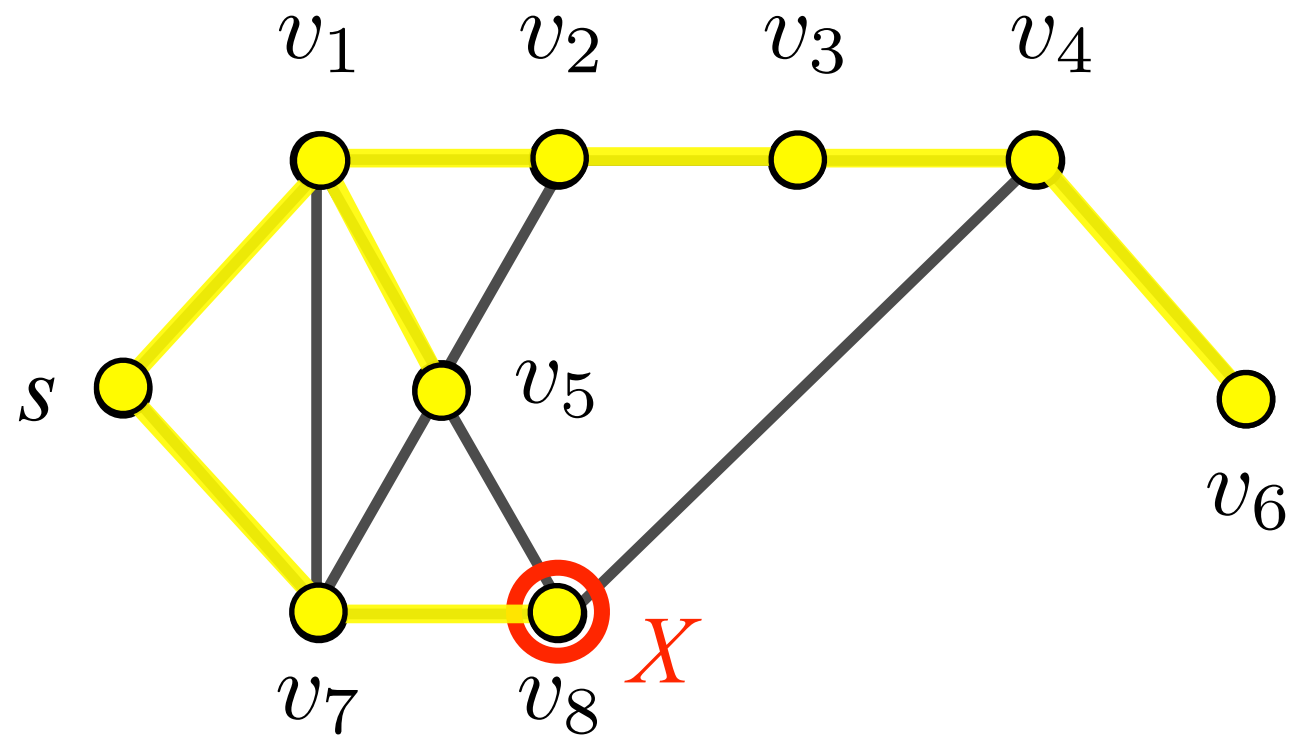
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

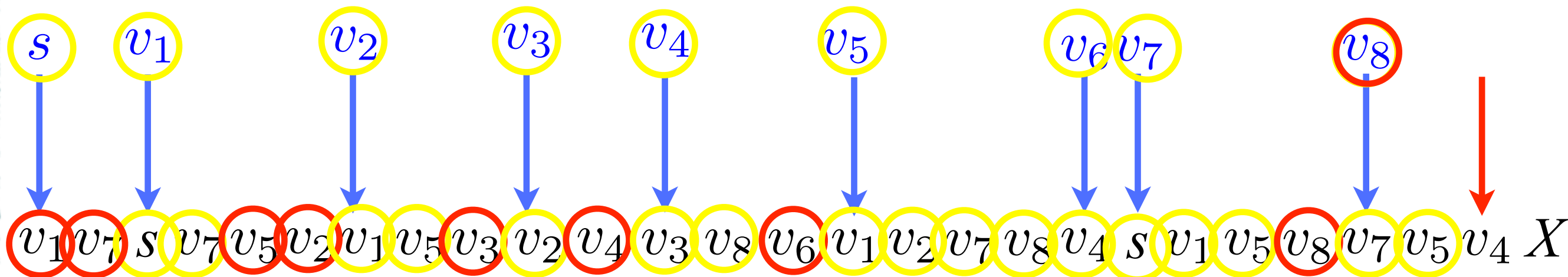
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

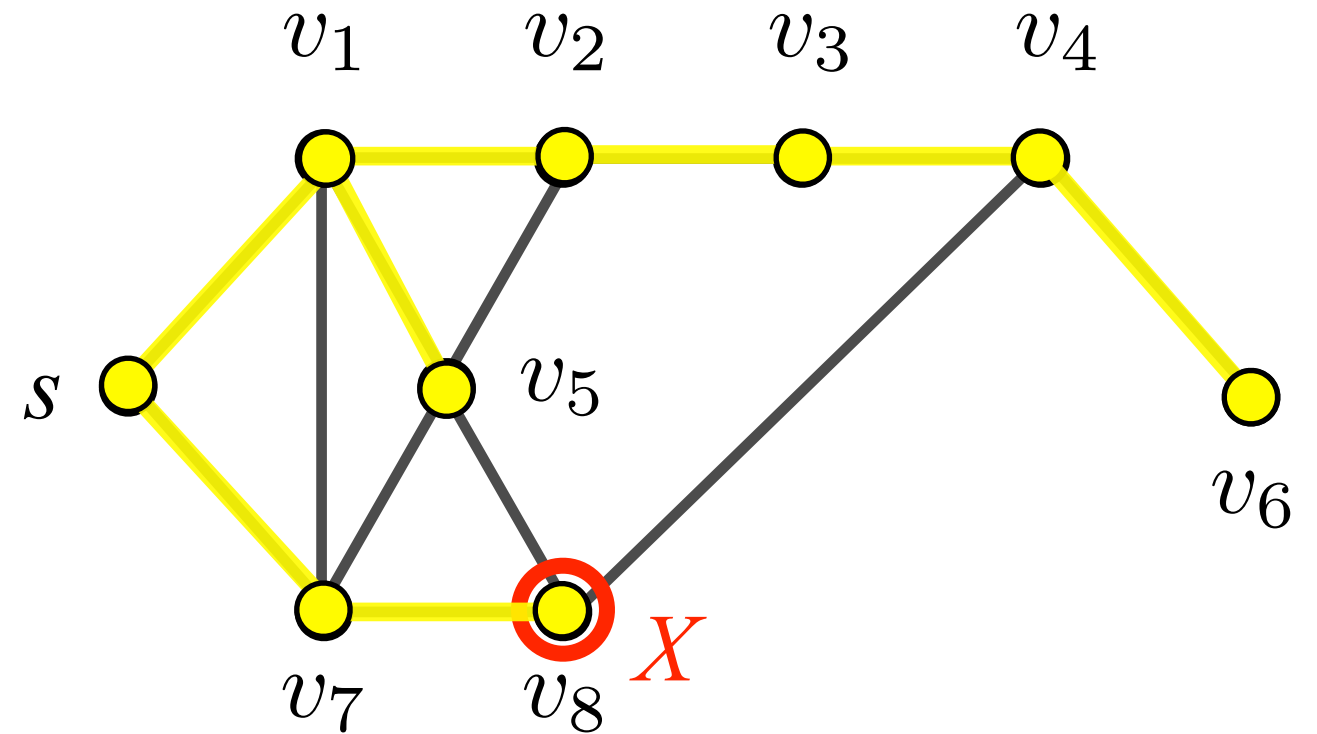
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

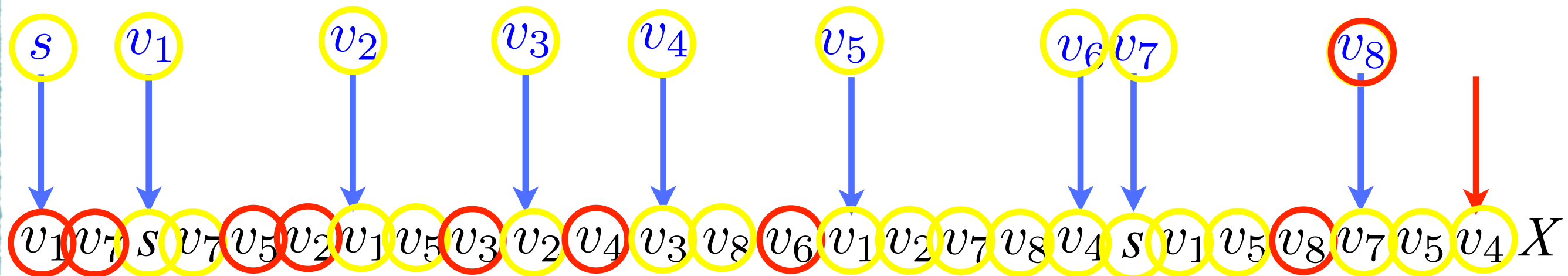
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

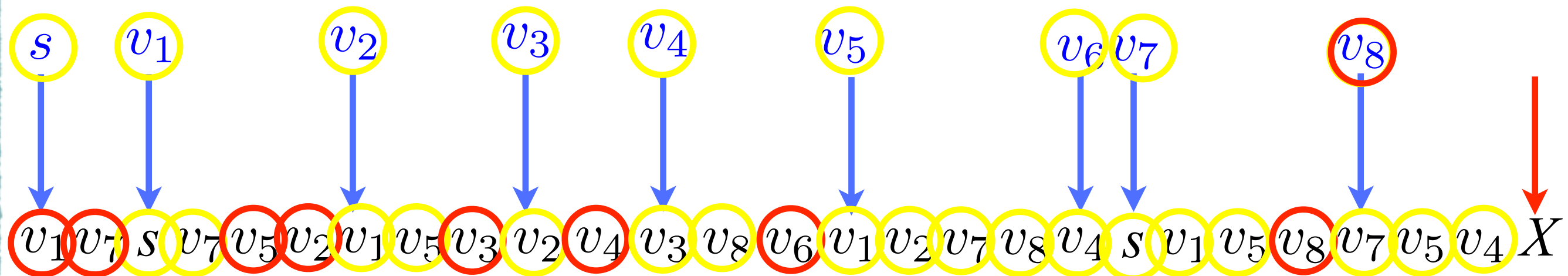
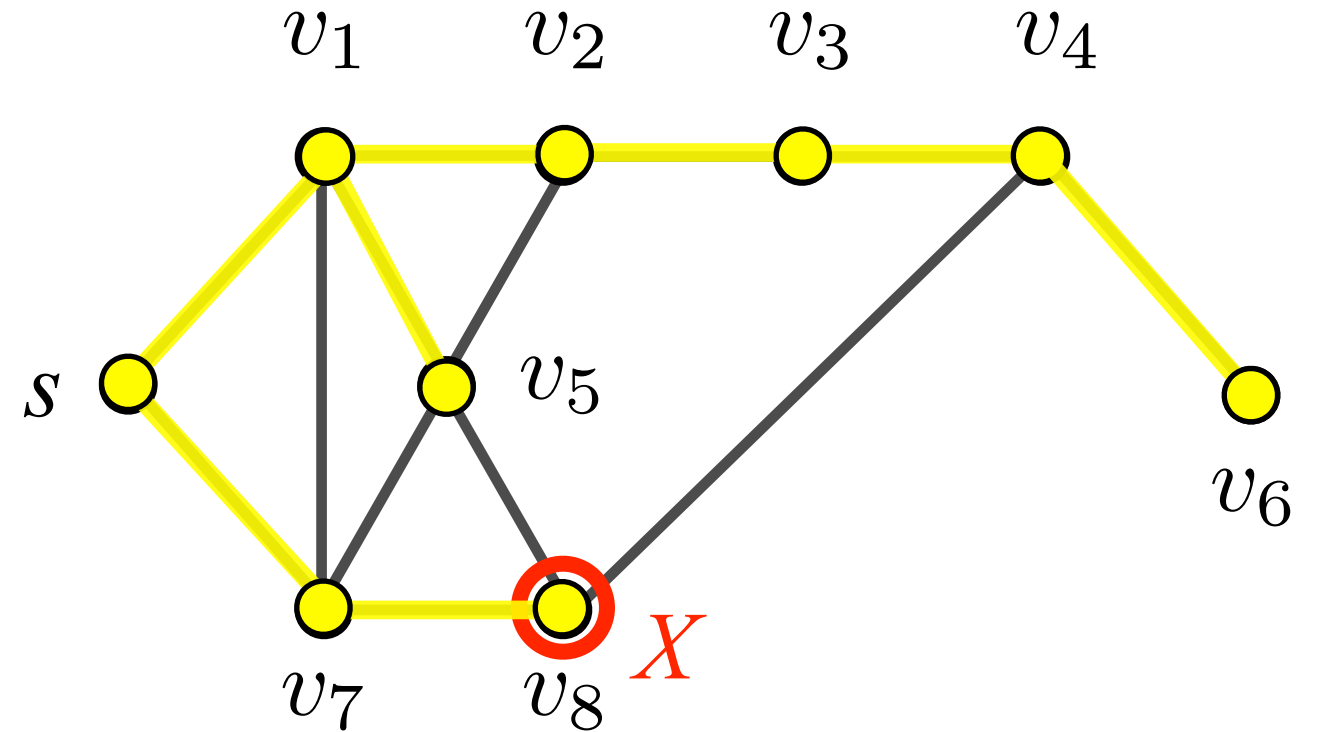
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

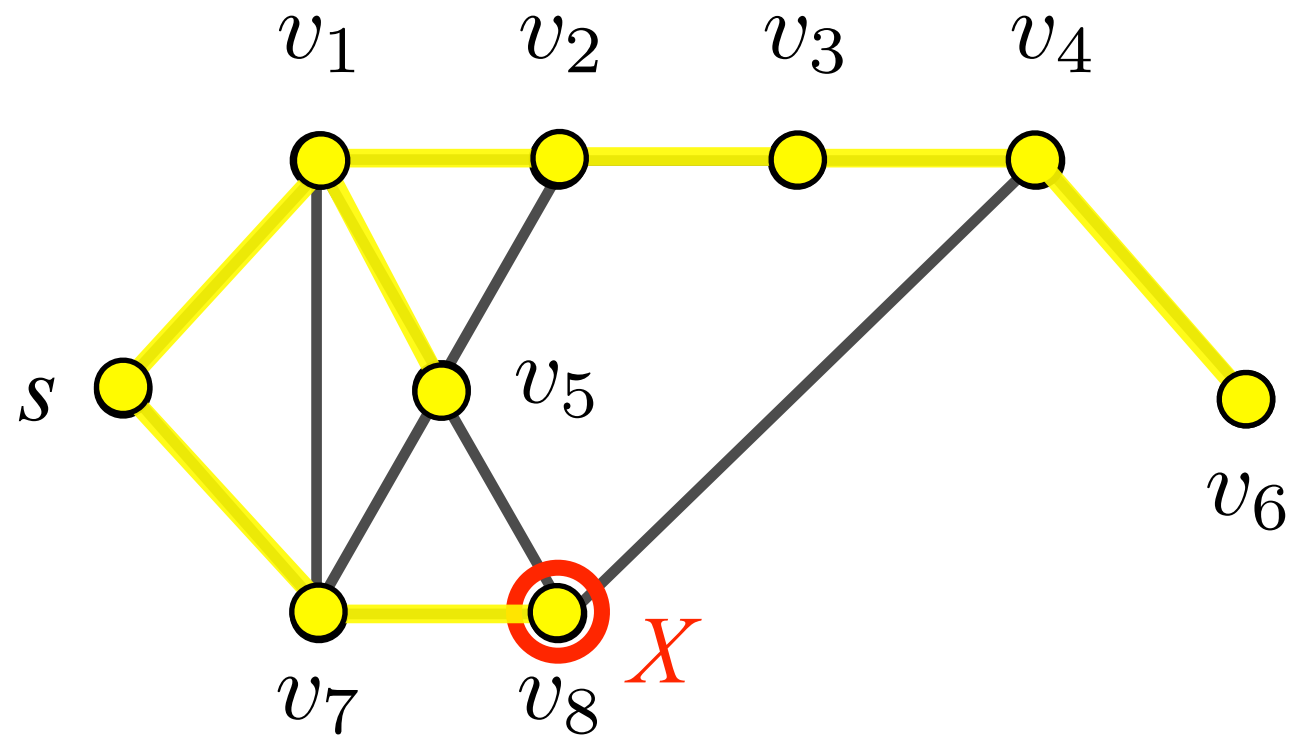
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

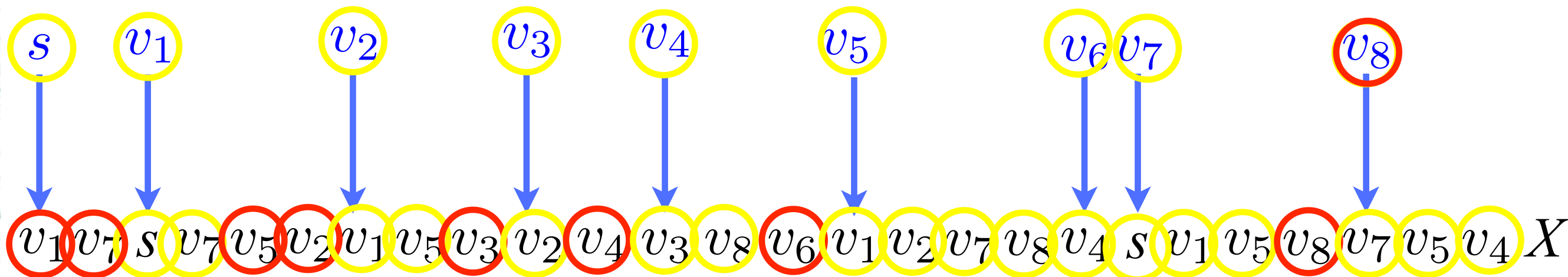
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

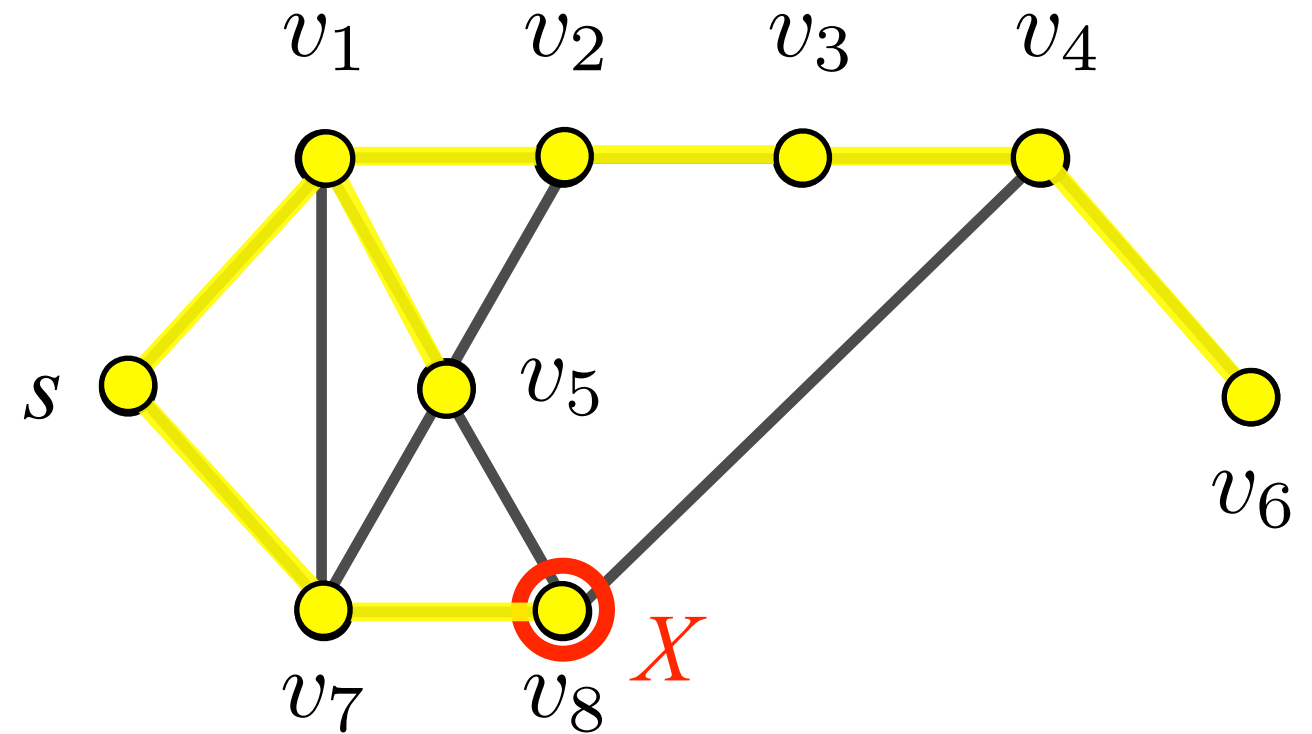
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

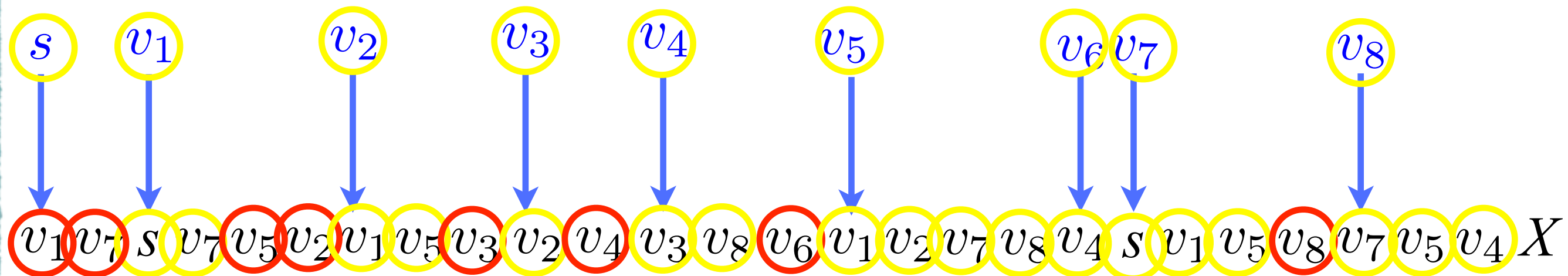
}

}

3. STOP



2



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

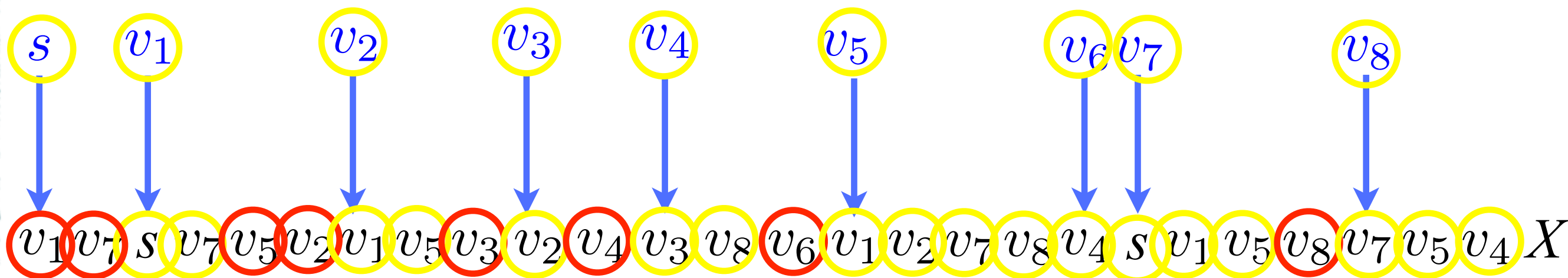
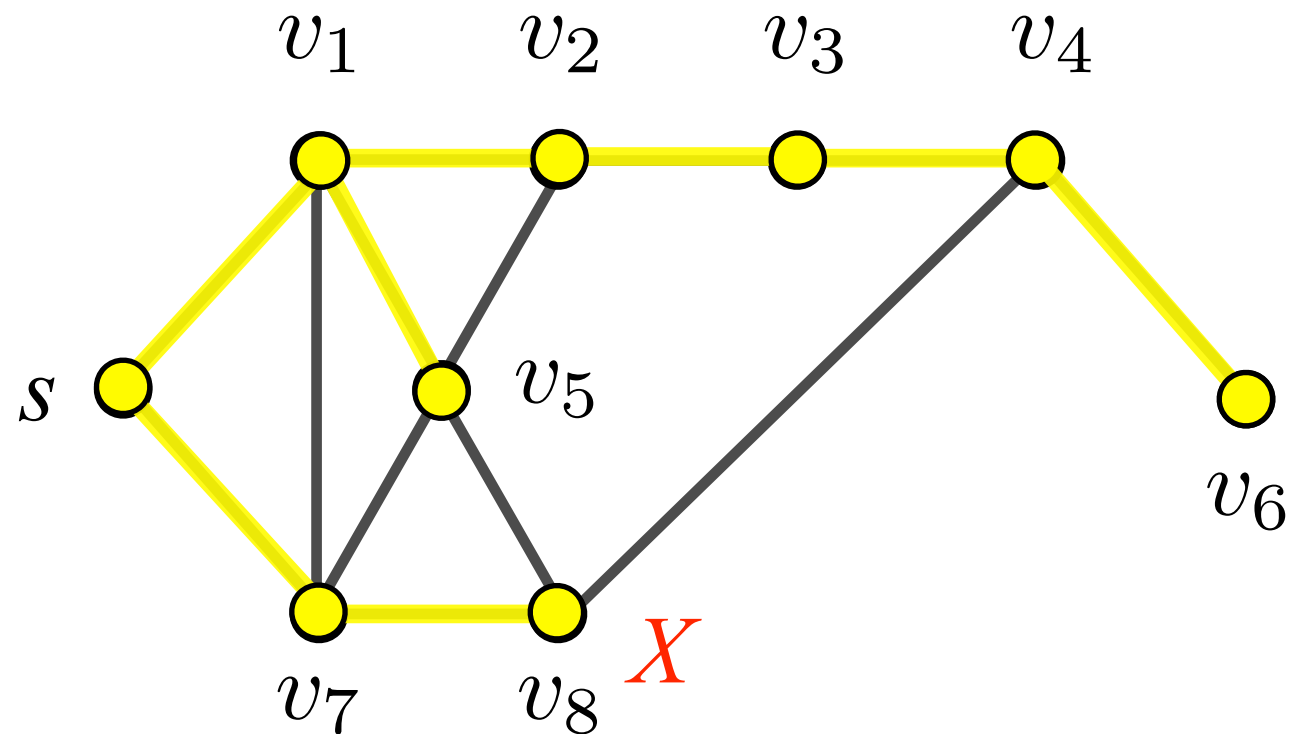
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

}

}

3. STOP



Algorithmus 3.7

INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$

WHILE ($R \neq \emptyset$) DO {

2.1. Wähle $v \in R$

2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN

2.2.1. $R := R \setminus \{v\}$

2.3. ELSE {

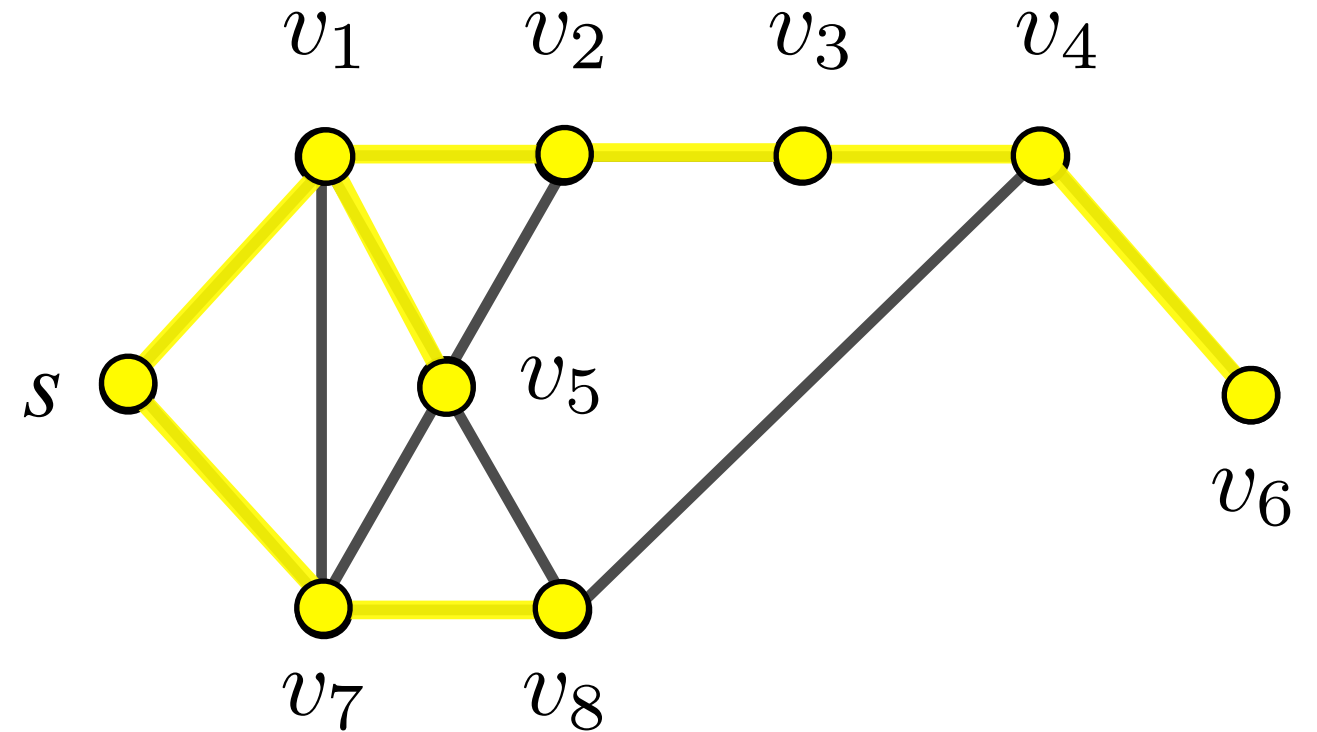
2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$

2.3.2. Setze $R := R \cup \{w\}$, $Y := Y \cup \{w\}$, $T := T \cup \{e\}$

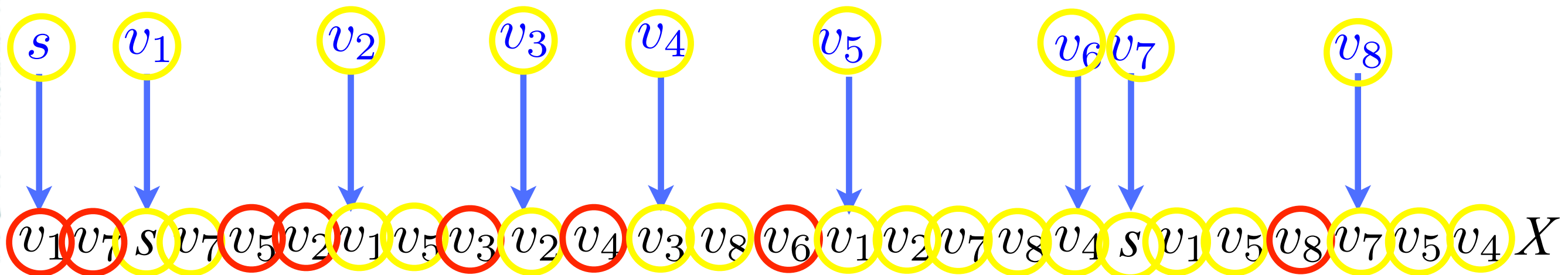
}

}

3. STOP



2



Algorithmus 3.7

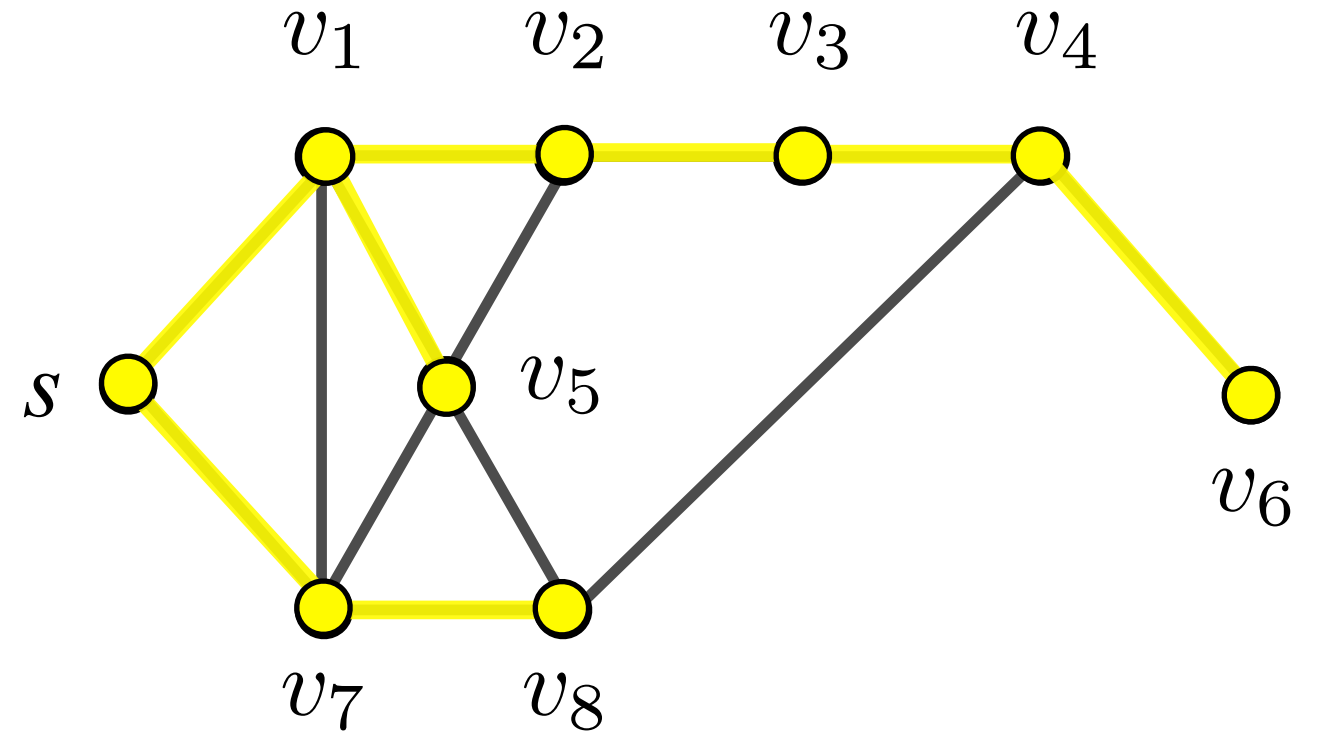
INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,
Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

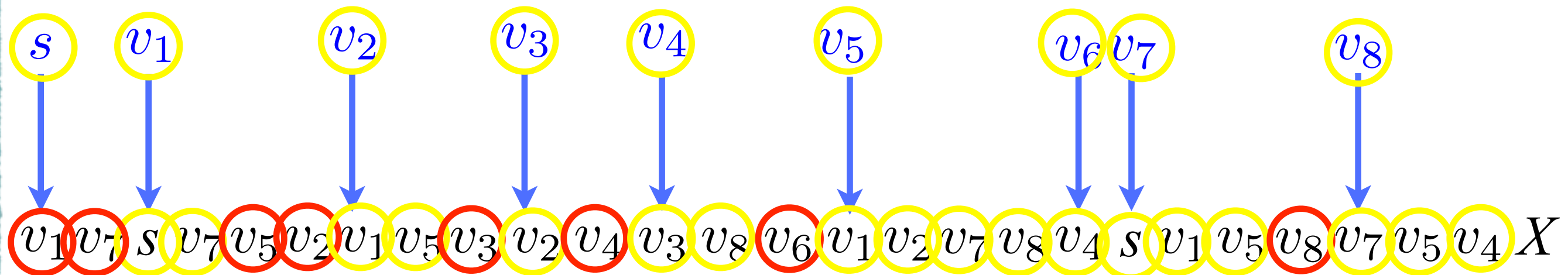
```

1. Sei  $R := \{s\}$ ,  $Y := \{s\}$ ,  $T := \emptyset$ 
2. WHILE ( $R \neq \emptyset$ ) DO {
    2.1. Wähle  $v \in R$ 
    2.2. IF (es gibt kein  $w \in V \setminus Y$  mit  $e = \{v, w\} \in E$ ) THEN
        2.2.1.  $R := R \setminus \{v\}$ 
    2.3. ELSE {
        2.3.1. Wähle ein  $w \in V \setminus Y$  mit  $e = \{v, w\} \in E$ 
        2.3.2. Setze  $R := R \cup \{w\}$ ,  $Y := Y \cup \{w\}$ ,  $T := T \cup \{e\}$ 
    }
}
3. STOP

```



2



Satz 3.13

Der Graphen-Scan-Algorithmus 3.7 lässt sich so implementieren, dass die Laufzeit $O(n+m)$ ist.

Algorithmus 3.7

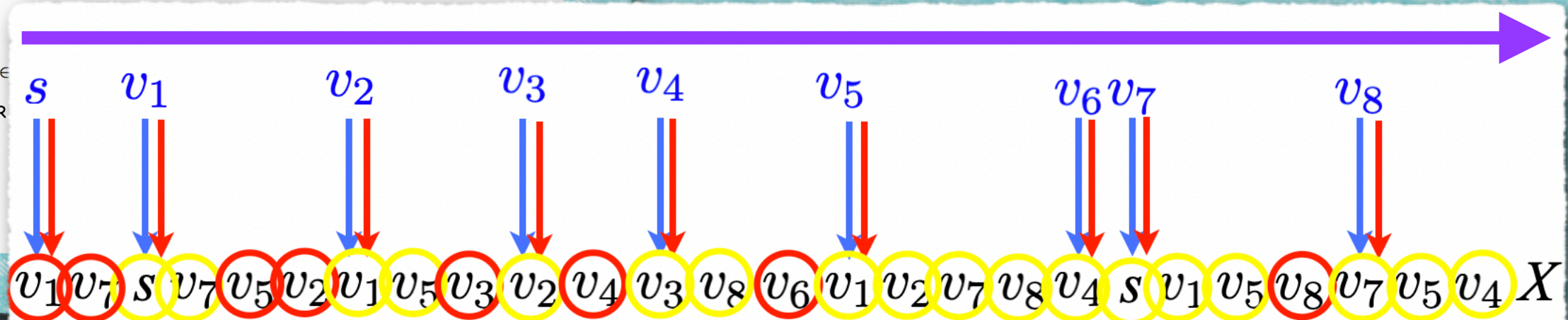
INPUT: Graph $G = (V, E)$, Knoten s

OUTPUT: Knotenmenge $Y \subseteq V$, die von s aus erreichbar ist,

Kantenmenge $T \subseteq E$, die die Erreichbarkeit sicherstellt

1. Sei $R := \{s\}$, $Y := \{s\}$, $T := \emptyset$
2. WHILE ($R \neq \emptyset$) DO {
 - 2.1. Wähle $v \in R$
 - 2.2. IF (es gibt kein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$) THEN
 - 2.2.1. $R := R \setminus \{v\}$
 - 2.3. ELSE {
 - 2.3.1. Wähle ein $w \in V \setminus Y$ mit $e = \{v, w\} \in E$
 - 2.3.2. Setze $R := R \cup \{w\}$}}
3. STOP

Adjazenzliste!



Demnächst mehr!

s.fekete@tu-bs.de