

Grafos

Grafos de sucessores

Prof. Edson Alves

Faculdade UnB Gama

Grafos de sucessores

Grafos de sucessores

Um grafo $G(V, E)$ é um grafo de sucessores se, para qualquer $u \in V$, o grau de saída de u é igual a 1.

Grafos de sucessores

Um grafo $G(V, E)$ é um **grafo de sucessores** se, para qualquer $u \in V$, o grau de saída de u é igual a 1.

Um grafo de sucessores também é denominado **grafo funcional**, pois está associado a uma função $\text{succ} : V \rightarrow V$ que define o conjunto de arestas

$$E = \{ (u, \text{succ}(u)) \mid u \in V \}$$

Características dos grafos de sucessores

Características dos grafos de sucessores

- ★ Um grafo de sucessores G tem exatamente $|V|$ arestas

Características dos grafos de sucessores

- ★ Um grafo de sucessores G tem exatamente $|V|$ arestas
- ★ Há, no mínimo, um ciclo em G

Características dos grafos de sucessores

- ★ Um grafo de sucessores G tem exatamente $|V|$ arestas
- ★ Há, no mínimo, um ciclo em G
- ★ De fato, G é composto por k componentes, cada um deles com ao menos um ciclo e um ou mais caminhos que levam a estes ciclos

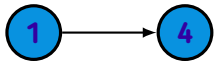
Características dos grafos de sucessores

- ★ Um grafo de sucessores G tem exatamente $|V|$ arestas
- ★ Há, no mínimo, um ciclo em G
- ★ De fato, G é composto por k componentes, cada um deles com ao menos um ciclo e um ou mais caminhos que levam a estes ciclos
- ★ Cada nó u tem um sucessor $\text{succ}(u)$ único

	1	2	3	4	5	6	7	8	9	10
succ[u] =	4	5	3	7	2	8	1	9	6	8

1

	1	2	3	4	5	6	7	8	9	10
$\text{succ}[u] =$	4	5	3	7	2	8	1	9	6	8

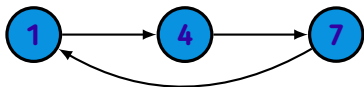


$\text{succ}[u] =$

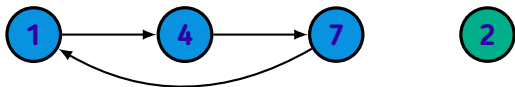
1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8



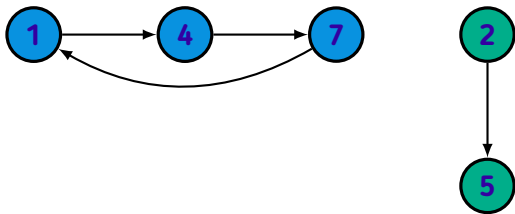
	1	2	3	4	5	6	7	8	9	10
$\text{succ}[u] =$	4	5	3	7	2	8	1	9	6	8



	1	2	3	4	5	6	7	8	9	10
$\text{succ}[u] =$	4	5	3	7	2	8	1	9	6	8

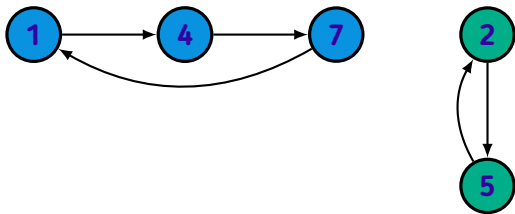


	1	2	3	4	5	6	7	8	9	10
$\text{succ}[u] =$	4	5	3	7	2	8	1	9	6	8

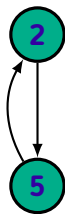
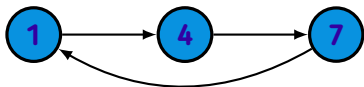


$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8

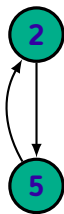
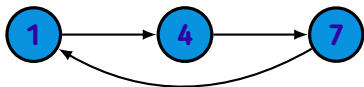


	1	2	3	4	5	6	7	8	9	10
$\text{succ}[u] =$	4	5	3	7	2	8	1	9	6	8



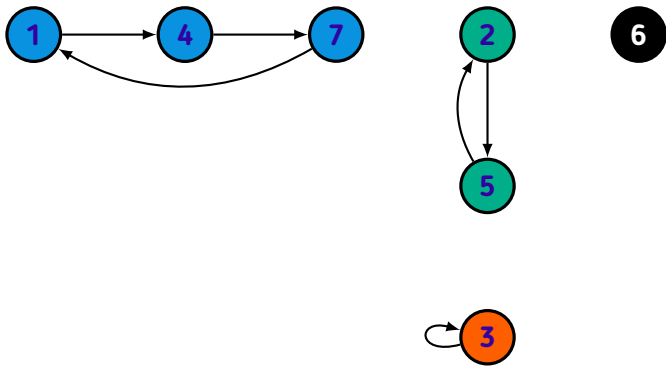
$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8



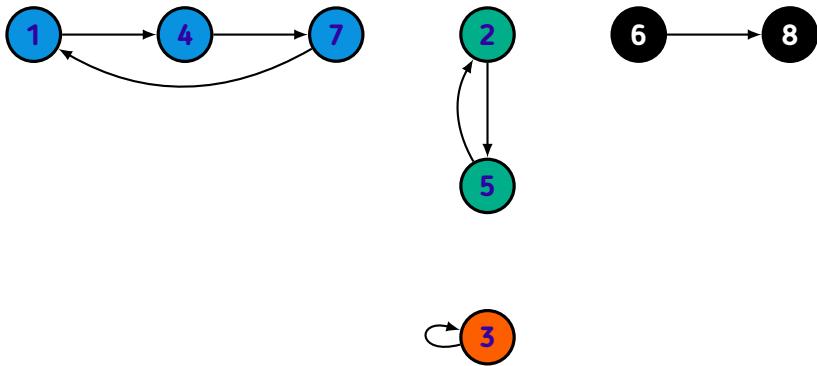
$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8



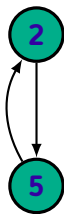
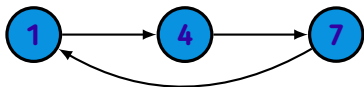
$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8

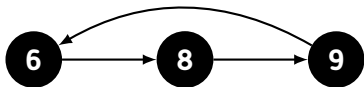
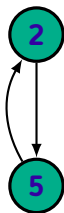
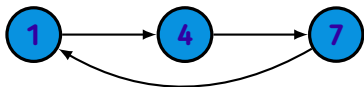


$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8

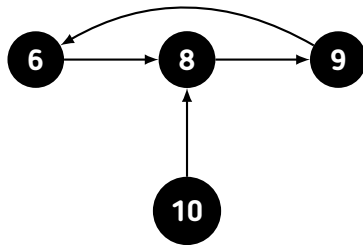
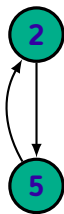
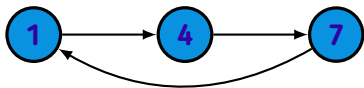


	1	2	3	4	5	6	7	8	9	10
$\text{succ}[u] =$	4	5	3	7	2	8	1	9	6	8



$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8



$\text{succ}[u] =$

1	2	3	4	5	6	7	8	9	10
4	5	3	7	2	8	1	9	6	8

k -ésimo sucessor

k -ésimo sucessor

Seja G um grafo de sucessores. O k -ésimo sucessor de um vértice u é definido como

$$\text{succ}(u, k) = \text{succ}^k(u) = \text{succ}(\text{succ}(\dots \text{succ}(u)))$$

k -ésimo sucessor

Seja G um grafo de sucessores. O k -ésimo sucessor de um vértice u é definido como

$$\text{succ}(u, k) = \text{succ}^k(u) = \underbrace{\text{succ}(\text{succ}(\dots \text{succ}(u)))}_{k \text{ vezes}}$$

Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

- ★ A função $\text{succ}(u, k)$ pode ser computada, trivialmente, em $O(k)$

Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

- ★ A função $\text{succ}(u, k)$ pode ser computada, trivialmente, em $O(k)$
- ★ Contudo, é possível computar $\text{succ}(u, v)$ em $O(\log k)$

Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

- ★ A função $\text{succ}(u, k)$ pode ser computada, trivialmente, em $O(k)$
- ★ Contudo, é possível computar $\text{succ}(u, v)$ em $O(\log k)$
- ★ Basta pré-computar, para cada $u \in V$, os valores de $\text{succ}(u, 2^i)$, para cada $i = 0, 1, \dots, M$ tal que $2^M \leq k$, por meio da recursão:

Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

- ★ A função $\text{succ}(u, k)$ pode ser computada, trivialmente, em $O(k)$
- ★ Contudo, é possível computar $\text{succ}(u, v)$ em $O(\log k)$
- ★ Basta pré-computar, para cada $u \in V$, os valores de $\text{succ}(u, 2^i)$, para cada $i = 0, 1, \dots, M$ tal que $2^M \leq k$, por meio da recursão:

$$\text{succ}(u, 2^i) = \begin{cases} \text{succ}(u), & \text{se } i = 0 \\ \text{succ}(\text{succ}(u, 2^{i-1}), 2^{i-1}), & \text{caso contrário} \end{cases}$$

Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

★ Estes valores podem ser pré-computados em $O(|V| \log k)$

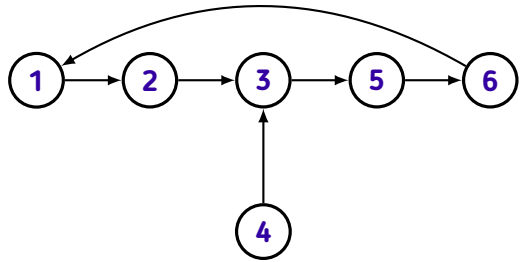
Cálculo de $\text{succ}(u, k)$ em $O(\log k)$

★ Estes valores podem ser pré-computados em $O(|V| \log k)$

★ De posse destes valores, $\text{succ}(u, k)$ é dado por

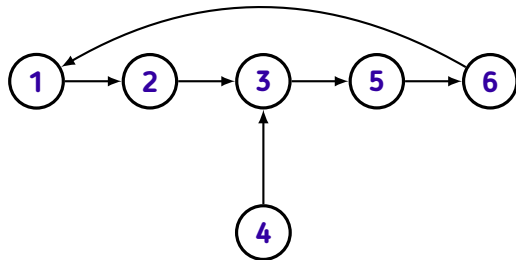
$$\text{succ}(u, k) = \text{succ}(\text{succ}(\text{succ}(\text{succ}(u, 2^\alpha), 2^\beta), \dots), 2^\omega),$$

onde $k = 2^\alpha + 2^\beta + \dots + 2^\omega$



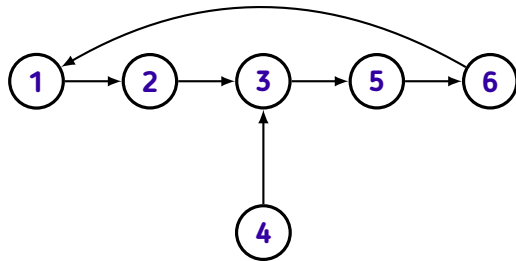
	1	2	4	8
1				
2				
3				
4				
5				
6				

$\text{succ}(u, k)$



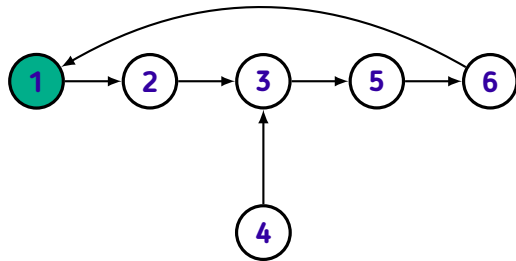
	1	2	4	8
1	2			
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



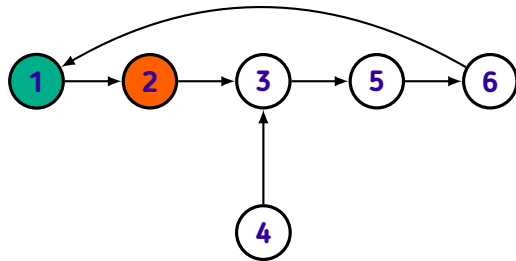
	1	2	4	8
1	2			
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



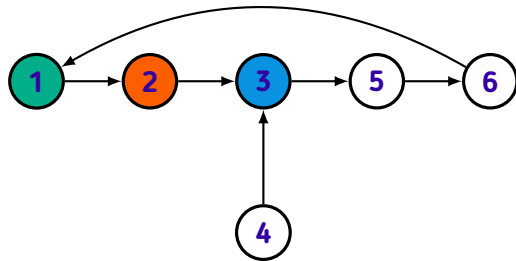
	1	2	4	8
1	2			
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



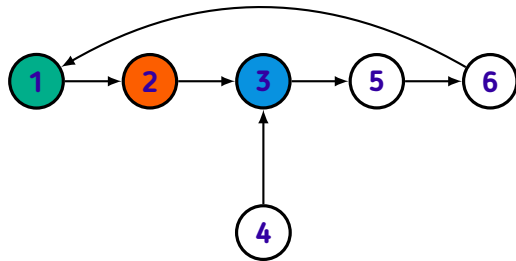
	1	2	4	8
1	2			
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



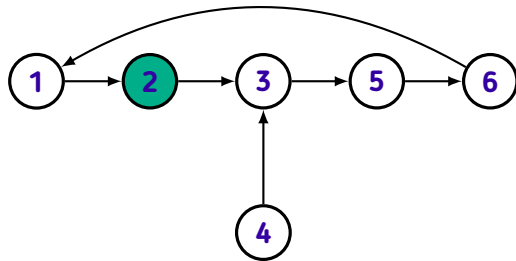
	1	2	4	8
1	2	3		
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



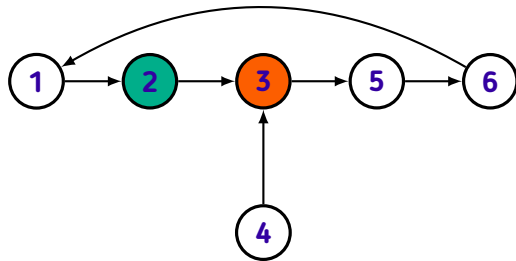
	1	2	4	8
1	2	3		
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



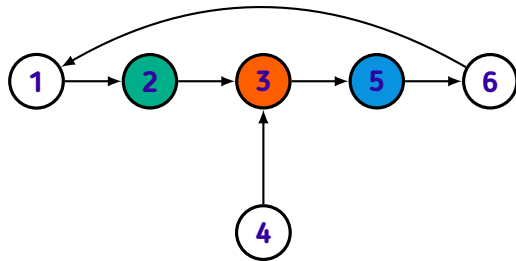
	1	2	4	8
1	2	3		
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



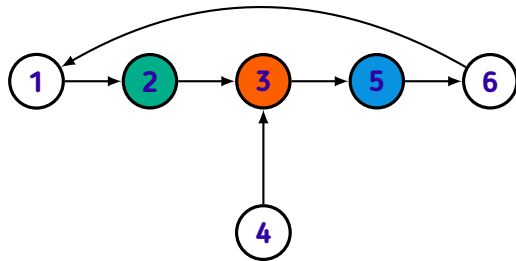
	1	2	4	8
1	2	3		
2	3			
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



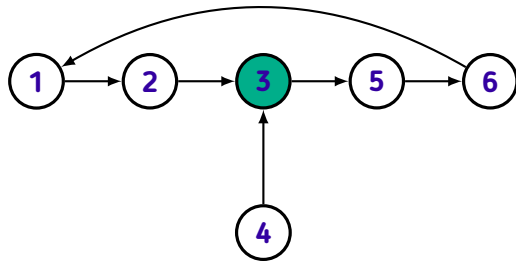
	1	2	4	8
1	2	3		
2	3	5		
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



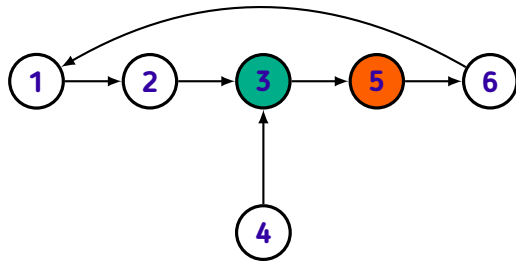
	1	2	4	8
1	2	3		
2	3	5		
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



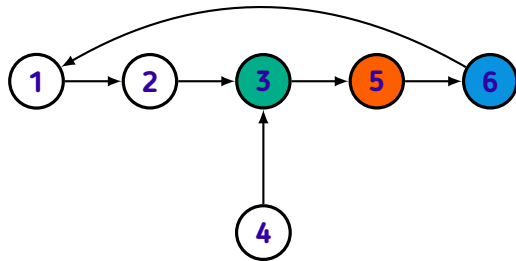
	1	2	4	8
1	2	3		
2	3	5		
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



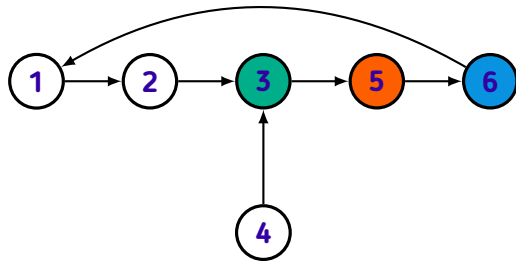
	1	2	4	8
1	2	3		
2	3	5		
3	5			
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



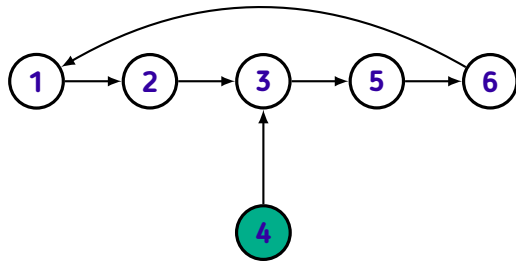
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



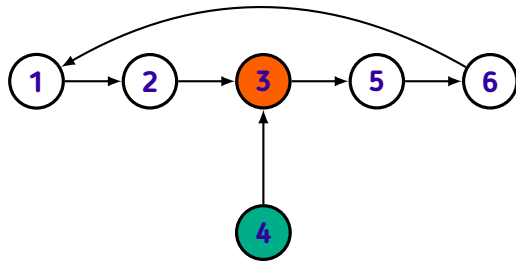
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



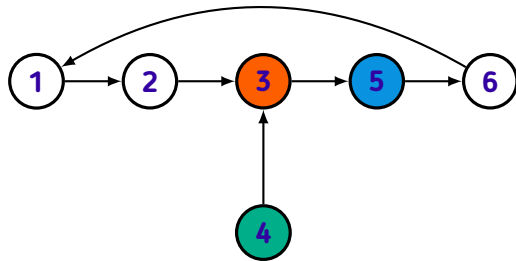
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



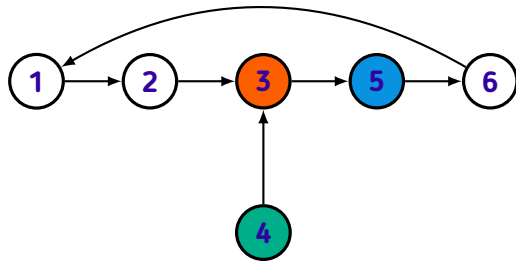
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3			
5	6			
6	1			

$\text{succ}(u, k)$



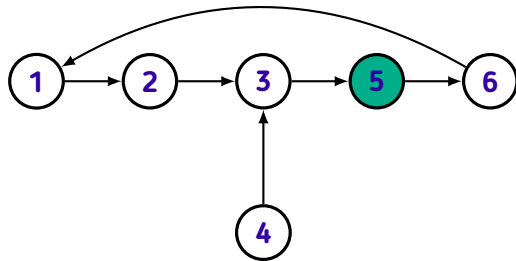
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6			
6	1			

$\text{succ}(u, k)$



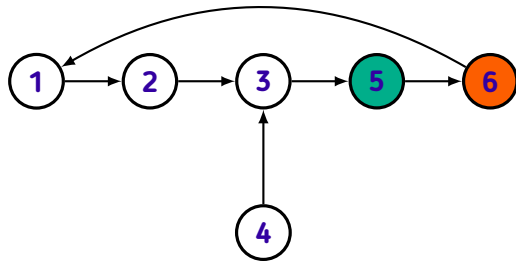
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6			
6	1			

$\text{succ}(u, k)$



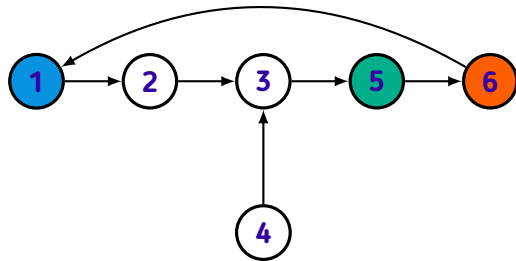
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6			
6	1			

$\text{succ}(u, k)$



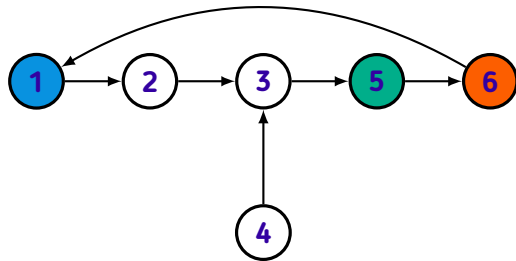
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6			
6	1			

$\text{succ}(u, k)$



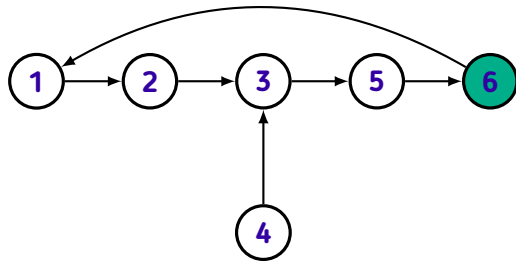
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1			

$\text{succ}(u, k)$



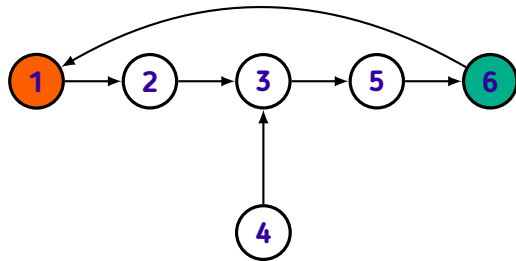
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1			

$\text{succ}(u, k)$



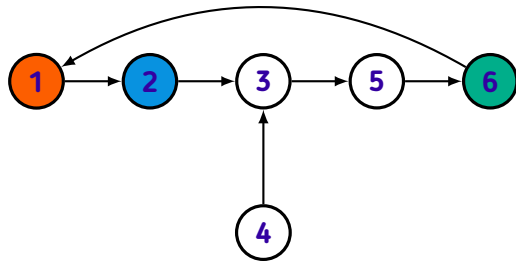
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1			

$\text{succ}(u, k)$



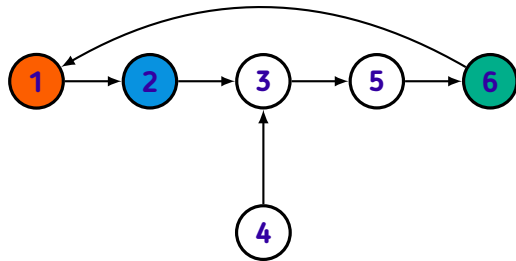
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1			

$\text{succ}(u, k)$



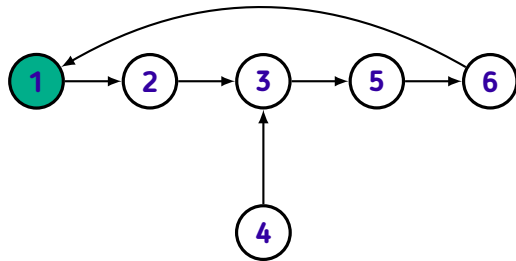
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



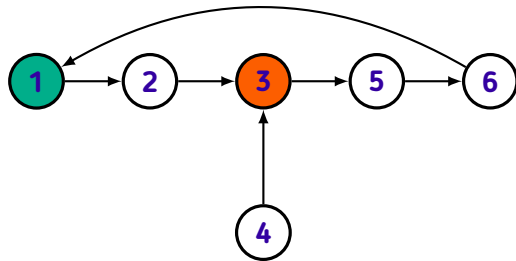
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



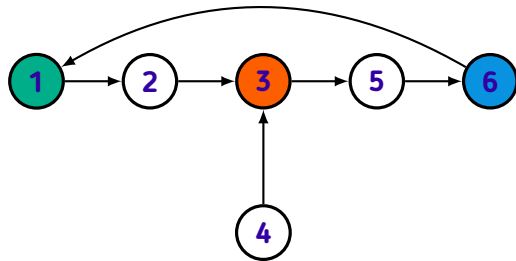
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



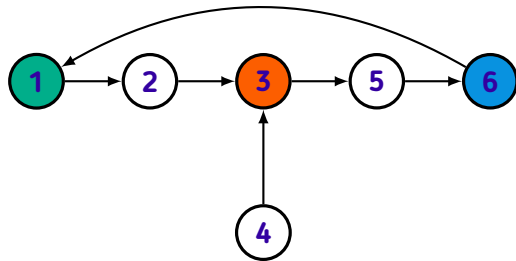
	1	2	4	8
1	2	3		
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



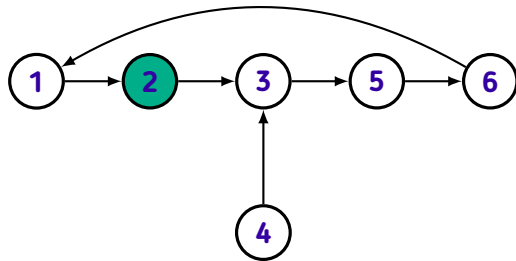
	1	2	4	8
1	2	3	6	
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



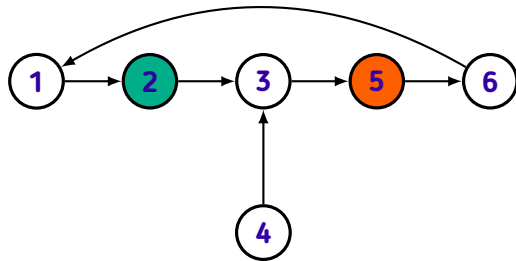
	1	2	4	8
1	2	3	6	
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



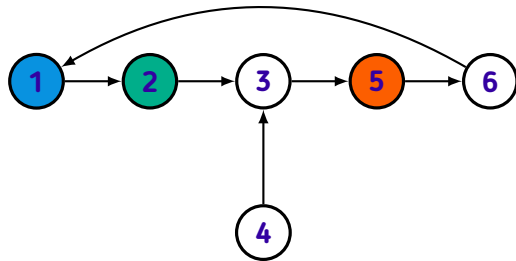
	1	2	4	8
1	2	3	6	
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



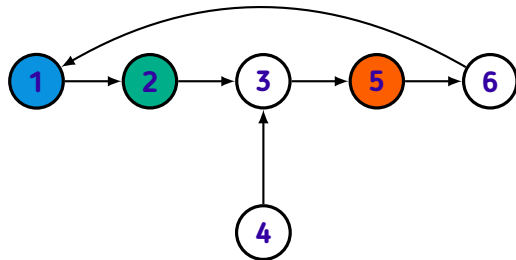
	1	2	4	8
1	2	3	6	
2	3	5		
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



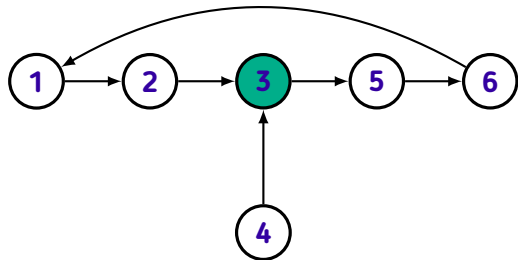
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



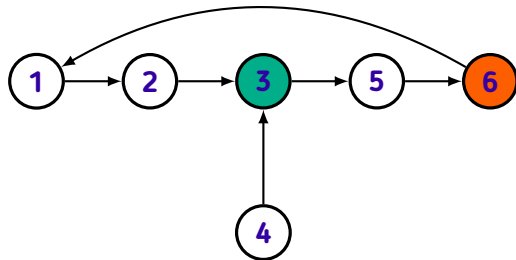
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



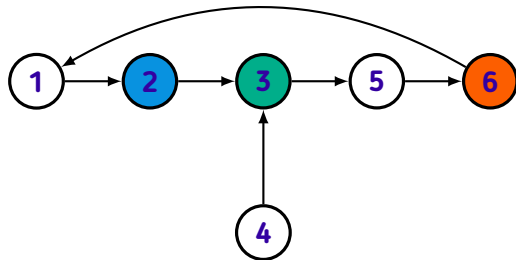
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



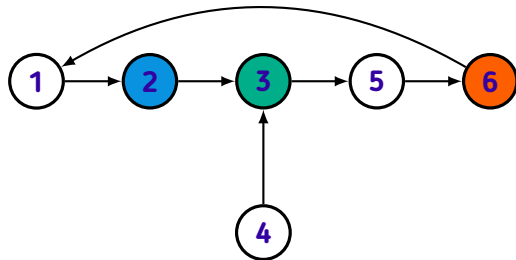
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6		
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



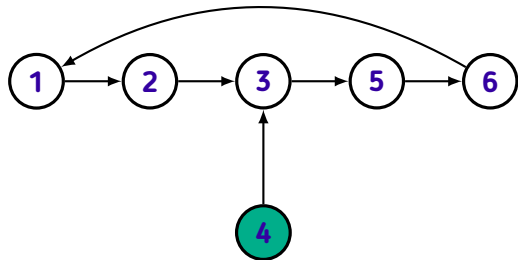
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



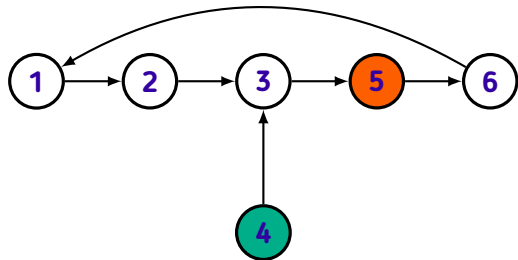
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



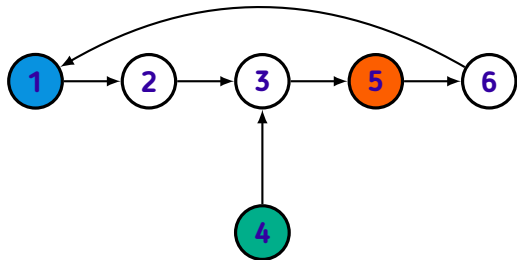
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



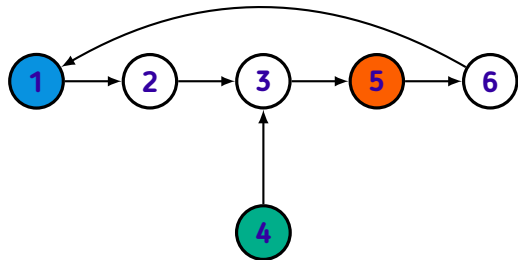
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5		
5	6	1		
6	1	2		

$\text{succ}(u, k)$



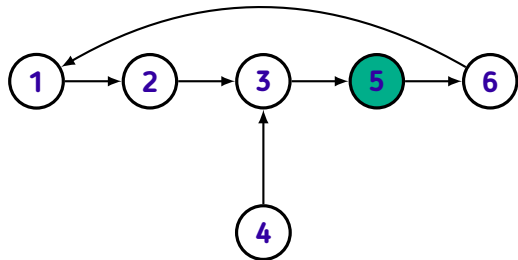
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1		
6	1	2		

$\text{succ}(u, k)$



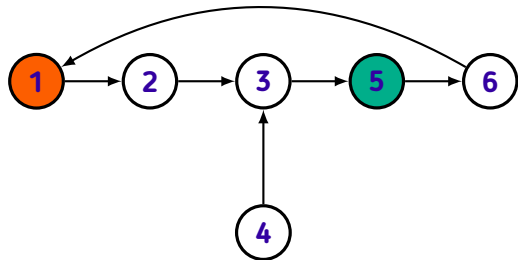
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1		
6	1	2		

$\text{succ}(u, k)$



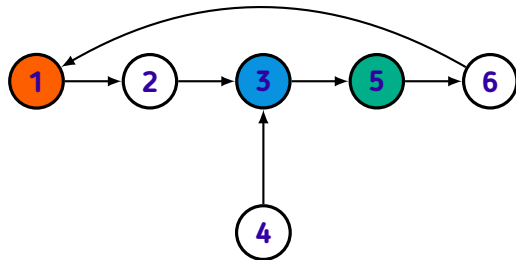
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1		
6	1	2		

$\text{succ}(u, k)$



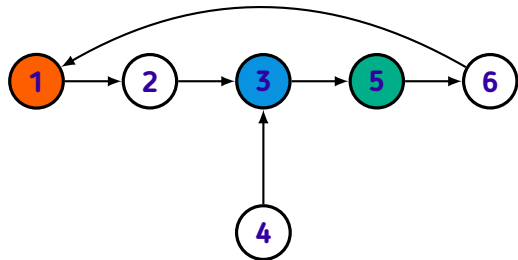
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1		
6	1	2		

$\text{succ}(u, k)$



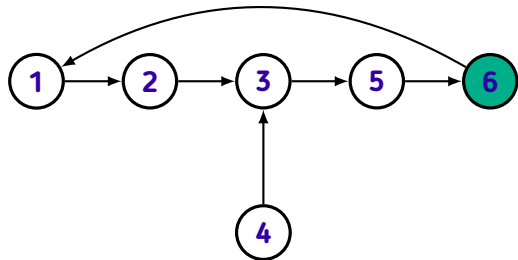
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2		

$\text{succ}(u, k)$



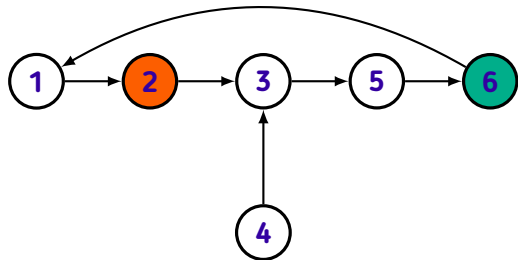
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2		

$\text{succ}(u, k)$



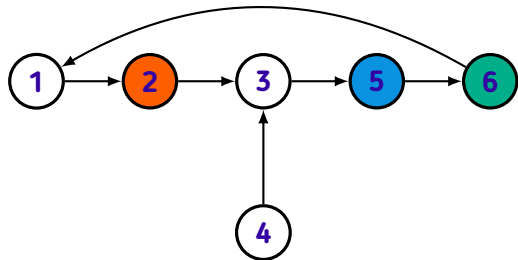
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2		

$\text{succ}(u, k)$



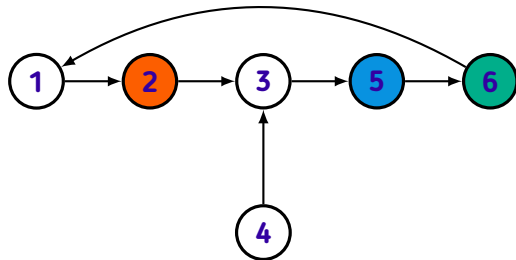
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2		

$\text{succ}(u, k)$



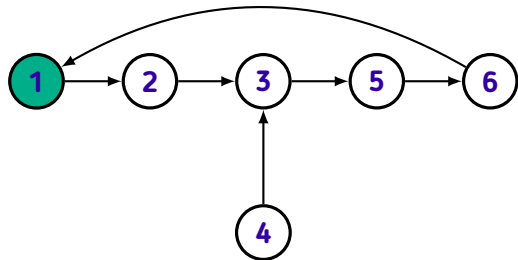
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



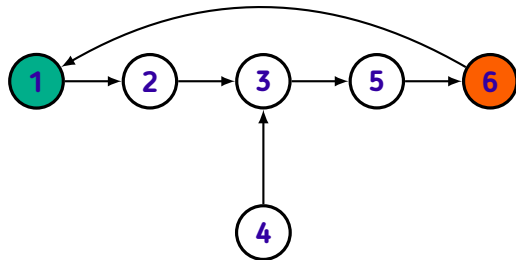
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



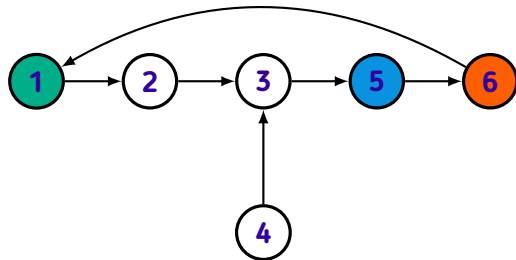
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



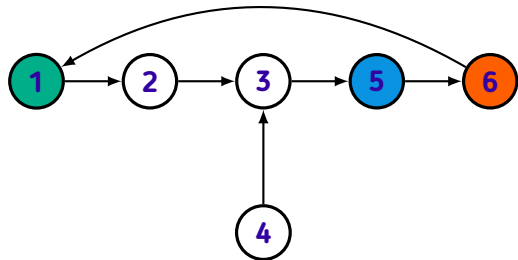
	1	2	4	8
1	2	3	6	
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



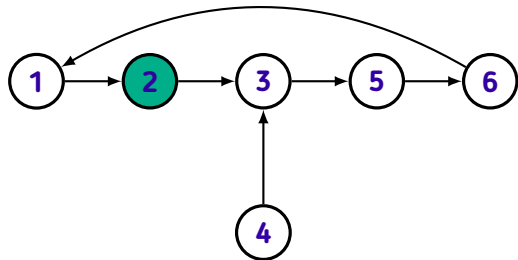
	1	2	4	8
1	2	3	6	5
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



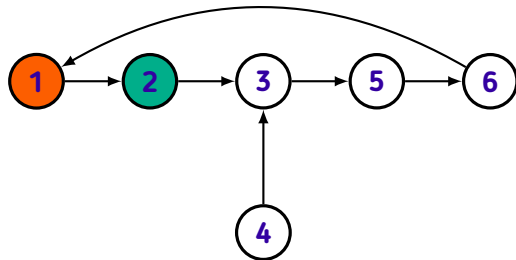
	1	2	4	8
1	2	3	6	5
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



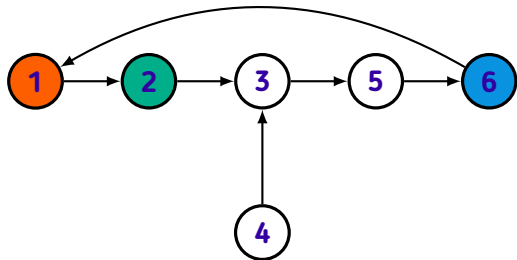
	1	2	4	8
1	2	3	6	5
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



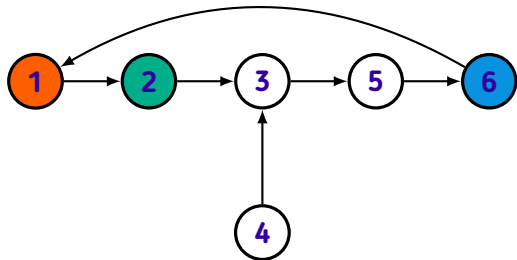
	1	2	4	8
1	2	3	6	5
2	3	5	1	
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



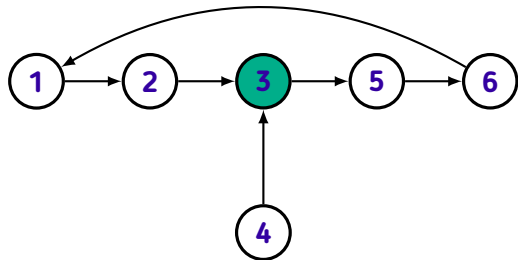
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



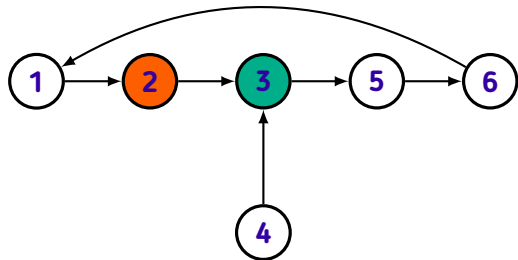
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



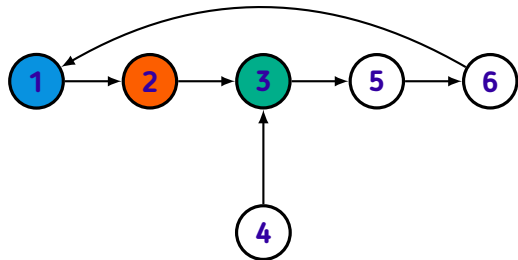
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



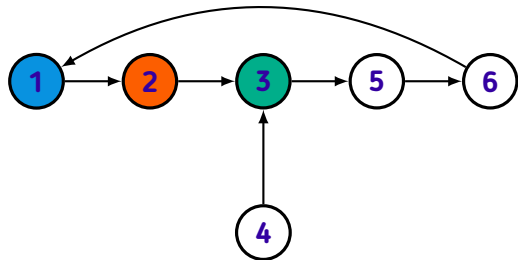
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



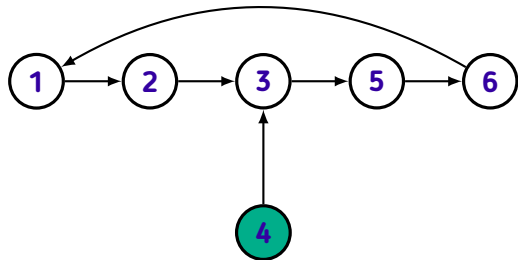
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



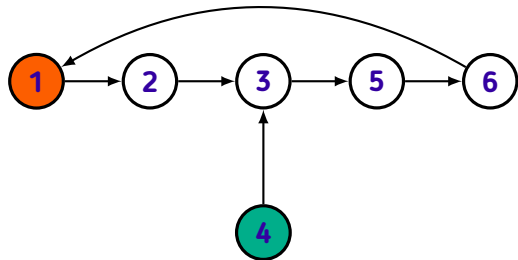
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



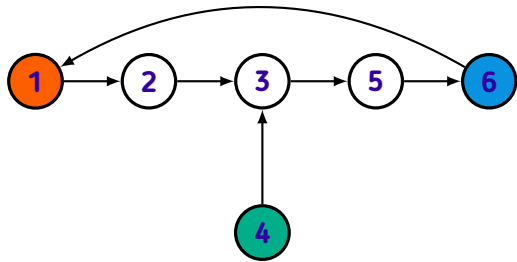
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



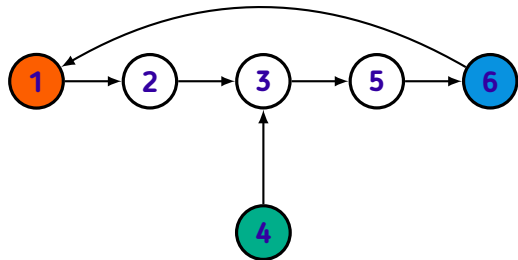
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



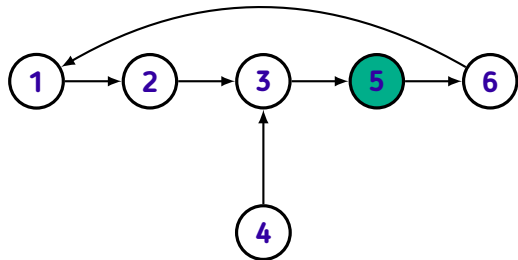
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



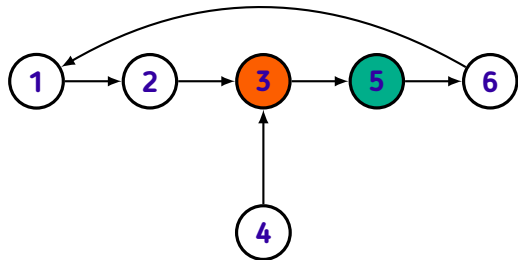
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



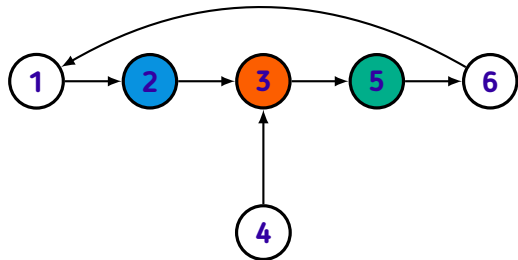
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



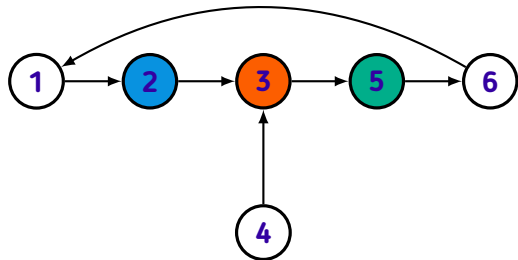
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	
6	1	2	5	

$\text{succ}(u, k)$



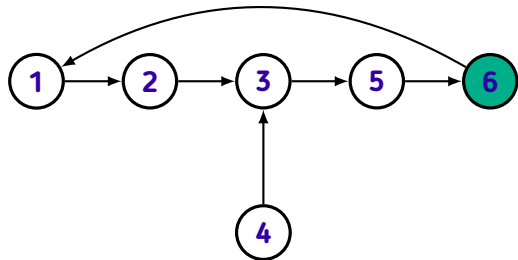
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	

$\text{succ}(u, k)$



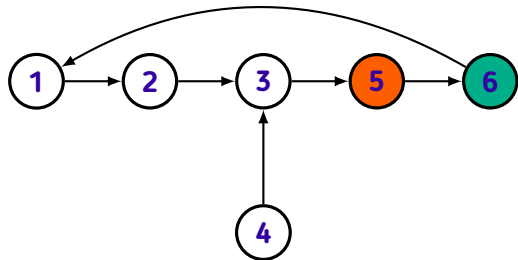
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	

$\text{succ}(u, k)$



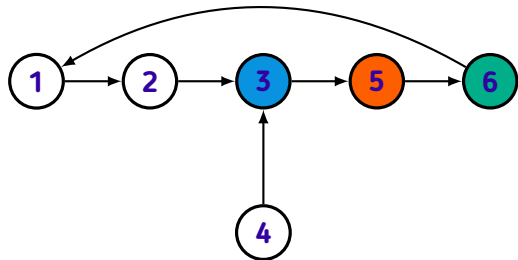
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	

$\text{succ}(u, k)$



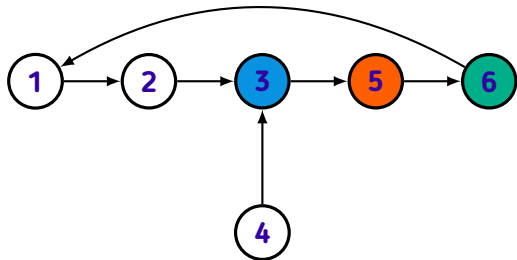
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	

$\text{succ}(u, k)$



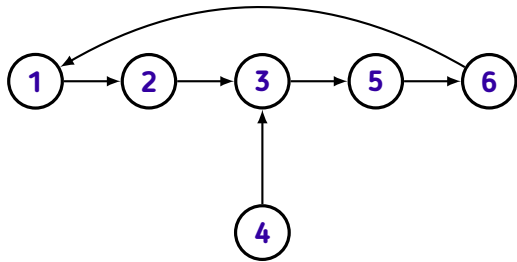
	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	3

$\text{succ}(u, k)$



	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	3

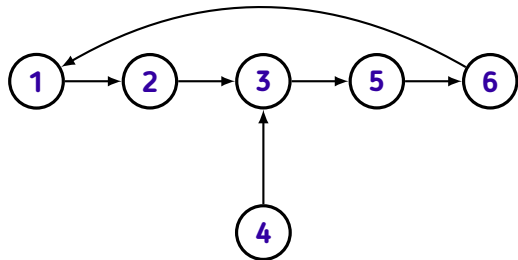
$\text{succ}(u, k)$



$$\text{succ}(4, 14) = \text{succ}(\text{succ}(\text{succ}(4, 2), 4), 8)$$

	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	3

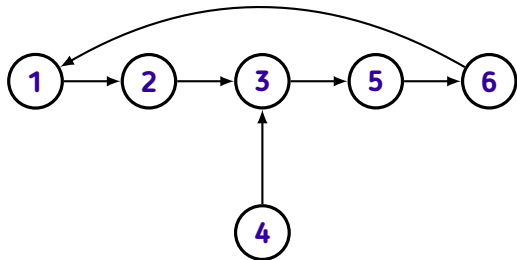
$\text{succ}(u, k)$



$$\text{succ}(4, 14) = \text{succ}(\text{succ}(5, 4), 8)$$

	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	3

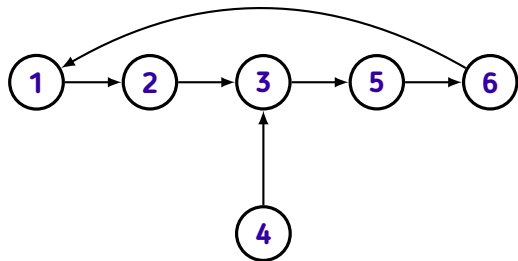
$\text{succ}(u, k)$



$$\text{succ}(4, 14) = \text{succ}(3, 8)$$

	1	2	4	8
1	2	3	6	5
2	3	5	1	6
3	5	6	2	1
4	3	5	1	6
5	6	1	3	2
6	1	2	5	3

$\text{succ}(u, k)$



$$\text{succ}(4, 14) = 1$$

```
void precomp(int N, int M, const vector<int>& s)
{
    for (int u = 1; u <= N; ++u)
        S[u][0] = s[u];

    for (int i = 1; i <= M; ++i)
        for (int u = 1; u <= N; ++u)
            S[u][i] = S[S[u][i - 1]][i - 1];
}

int succ(int u, int k)
{
    for (int i = 0; (1 << i) <= k; ++i)
        if (k & (1 << i))
            u = S[u][i];

    return u;
}
```

Problemas sugeridos

1. CSES 1750 – Planets Queries I
2. USACO 2017 December Contest, Silver – Problem 3: The Bovine Shuffle

Referências

1. HALIM, Felix; HALIM, Steve. *Competitive Programming 3*, 2010.
2. LAAKSONEN, Antti. *Competitive Programmer's Handbook*, 2018.
2. USACO, Guide. *Introduction to Functional Graphs*, acesso em 21/09/2021.