

Proiectarea și Analiza Algoritmilor

S II - 2025

Curs 6

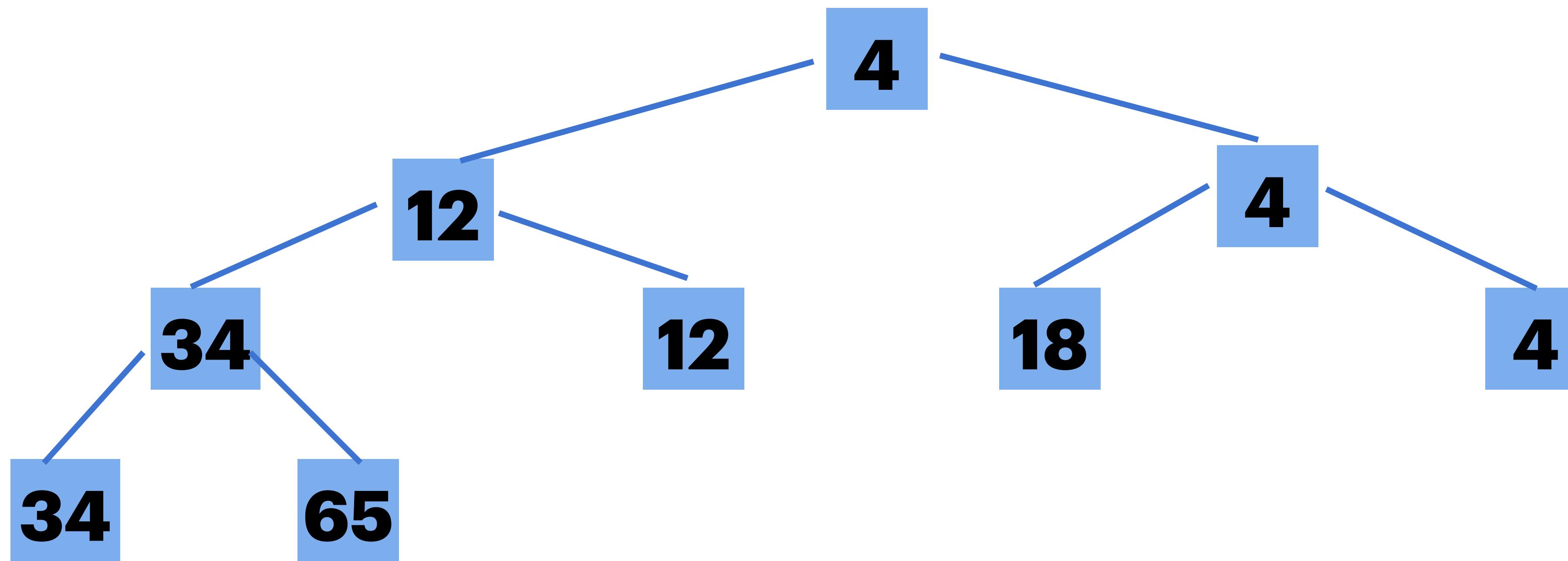
Arbore: Aplicații

Arbore Binari Parțial Ordonați

Pentru orice nod, cheia sa nu este mai mare (sau mai mică) decât cea a filor săi

Aplicație: HEAPSORT (SDA, an II, sem. 1)

He a p S o r t



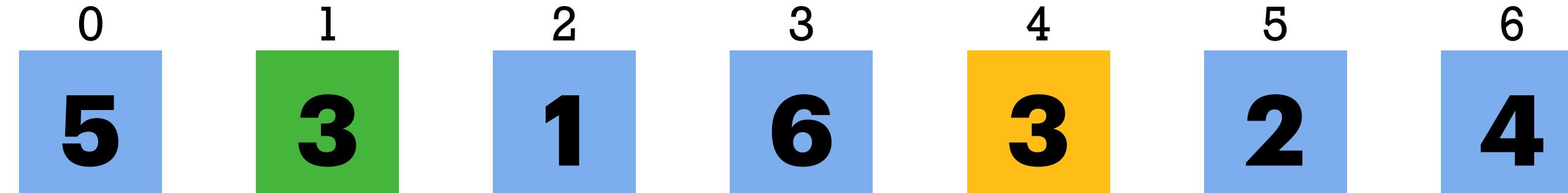
Heap Sort

- Heapsort - inventat de J. Williams
- Îmbunătățire semnificativă relativ la multe alte metode de sortare
- **Heap** = reprezentare prin tablou a unui **arbore binar parțial ordonat**
- Un **arbore binar parțial ordonat**, reprezentat cu tablou (**heap**) cu cheile h_1, h_2, \dots, h_r are următoarea proprietate:

$$\begin{cases} h_i \leq h_{2i} & \forall i = 1, 2, \dots, r/2 \\ h_i \leq h_{2i+1} \end{cases}$$

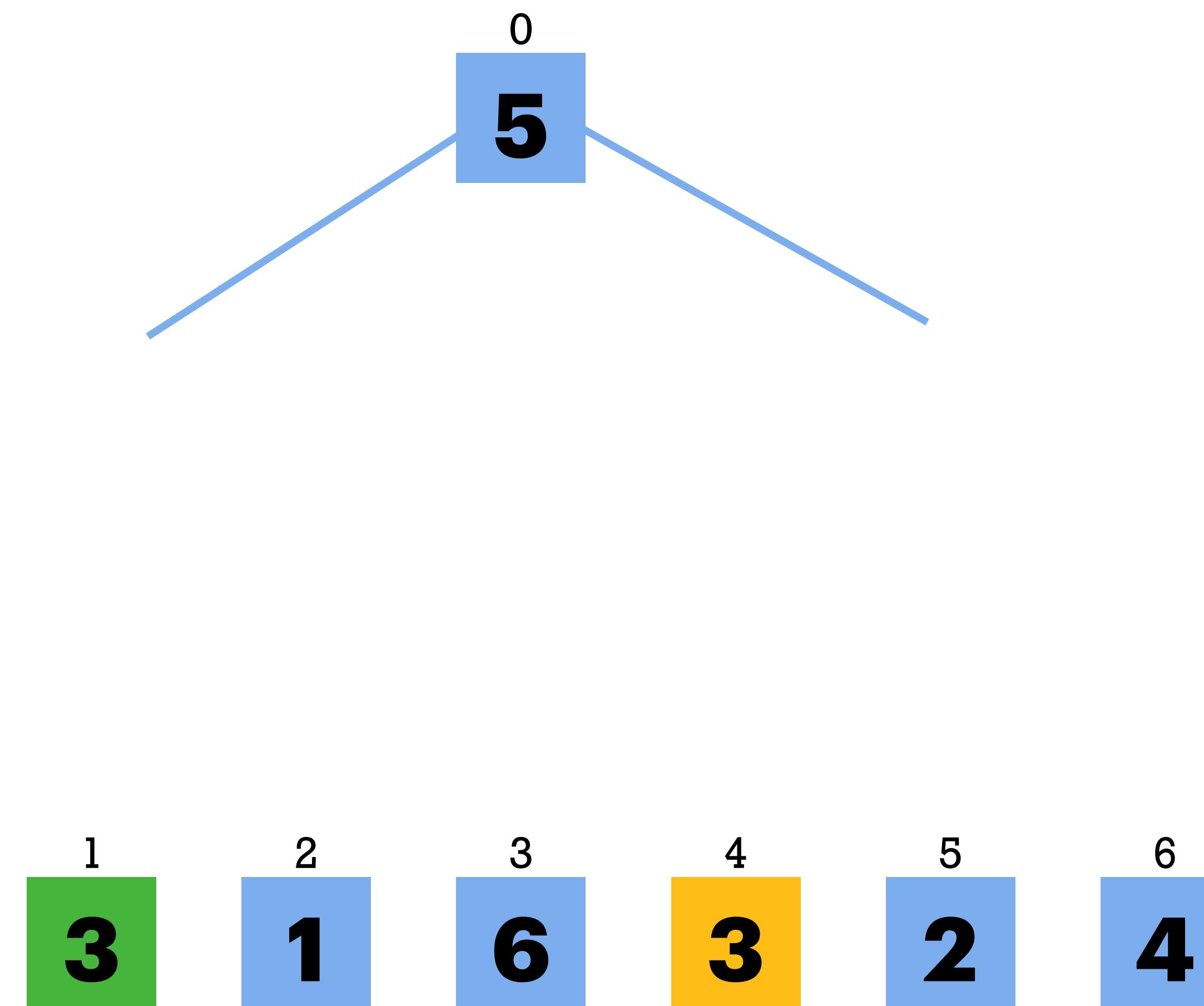
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



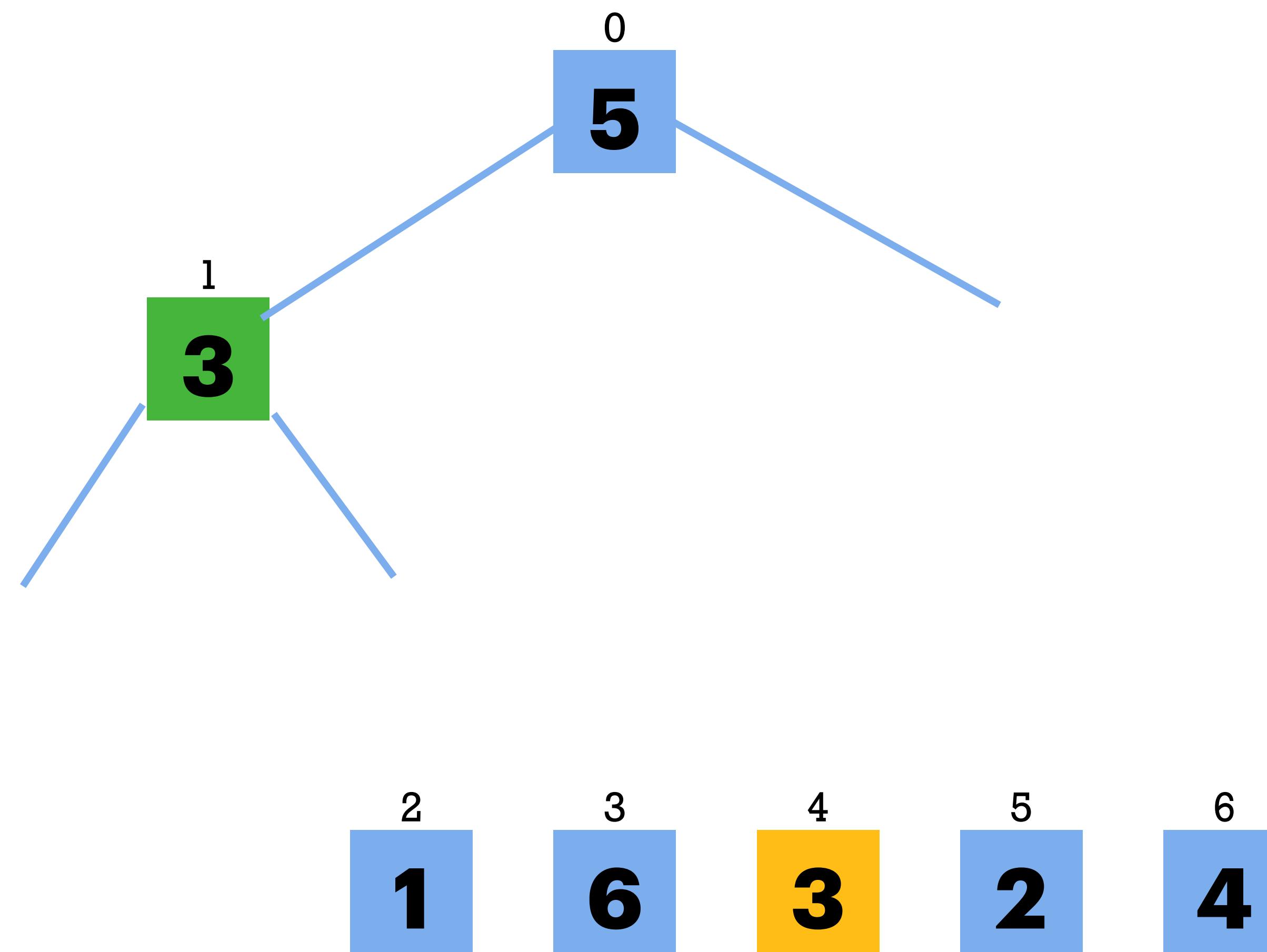
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



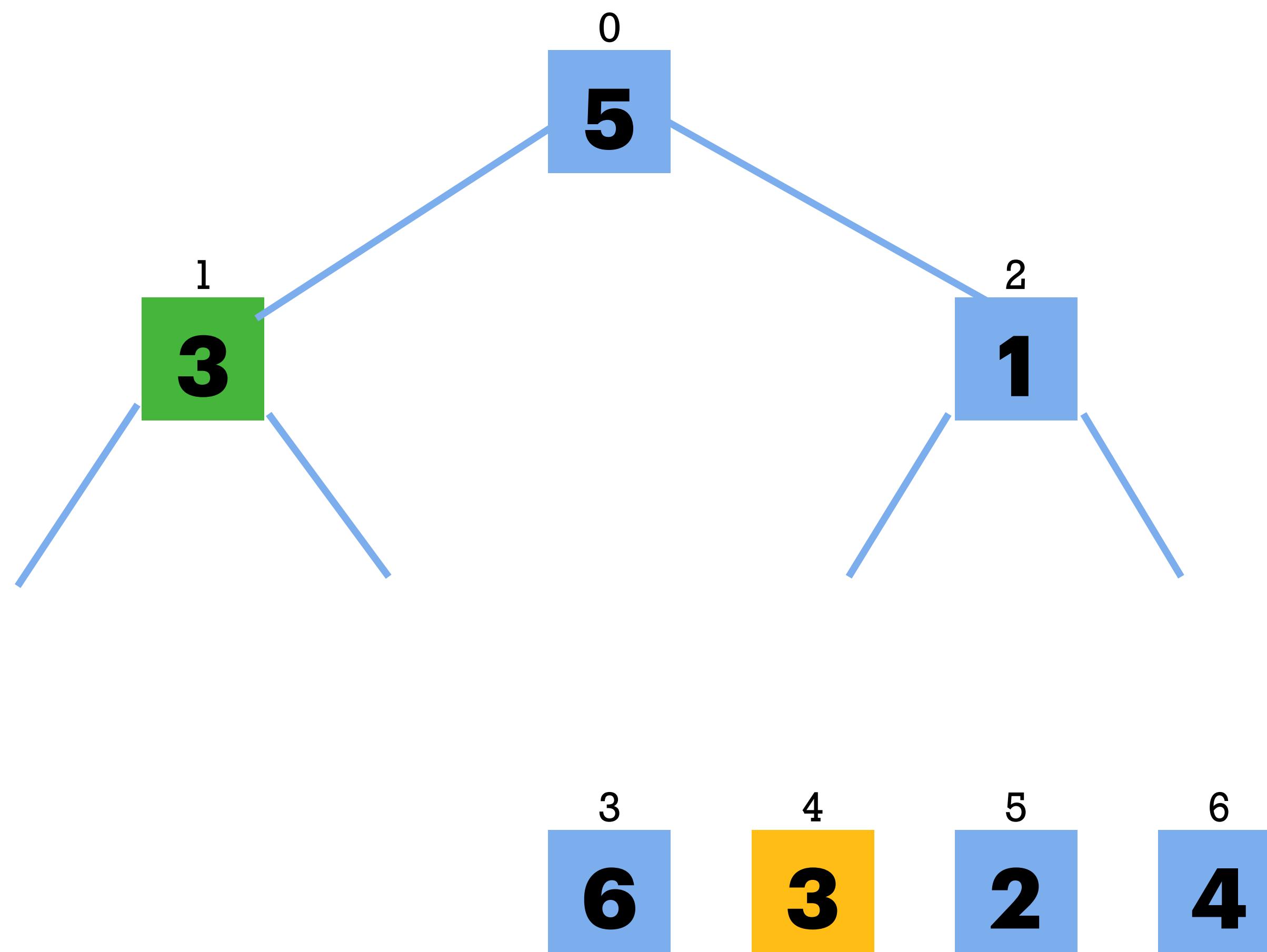
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



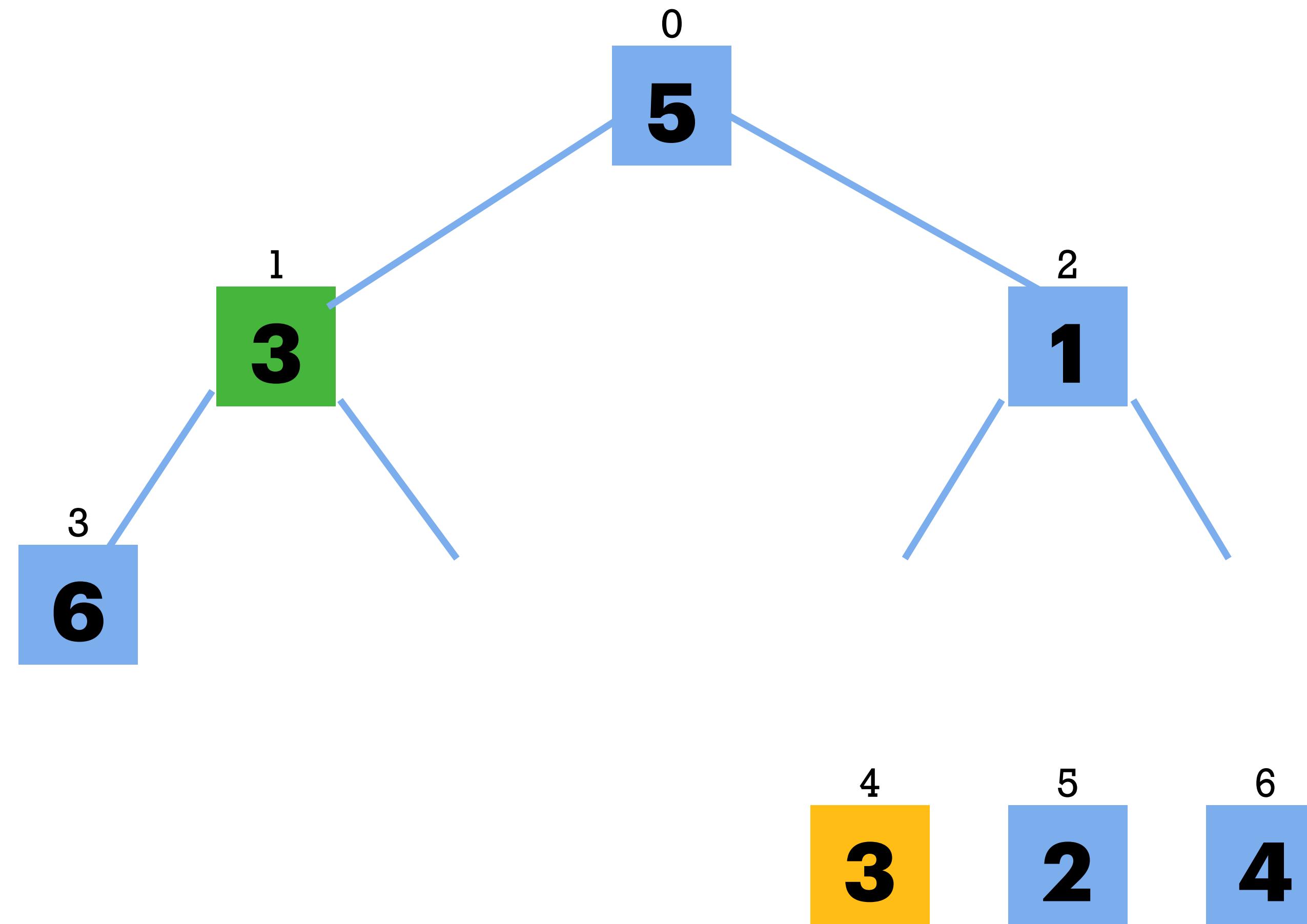
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



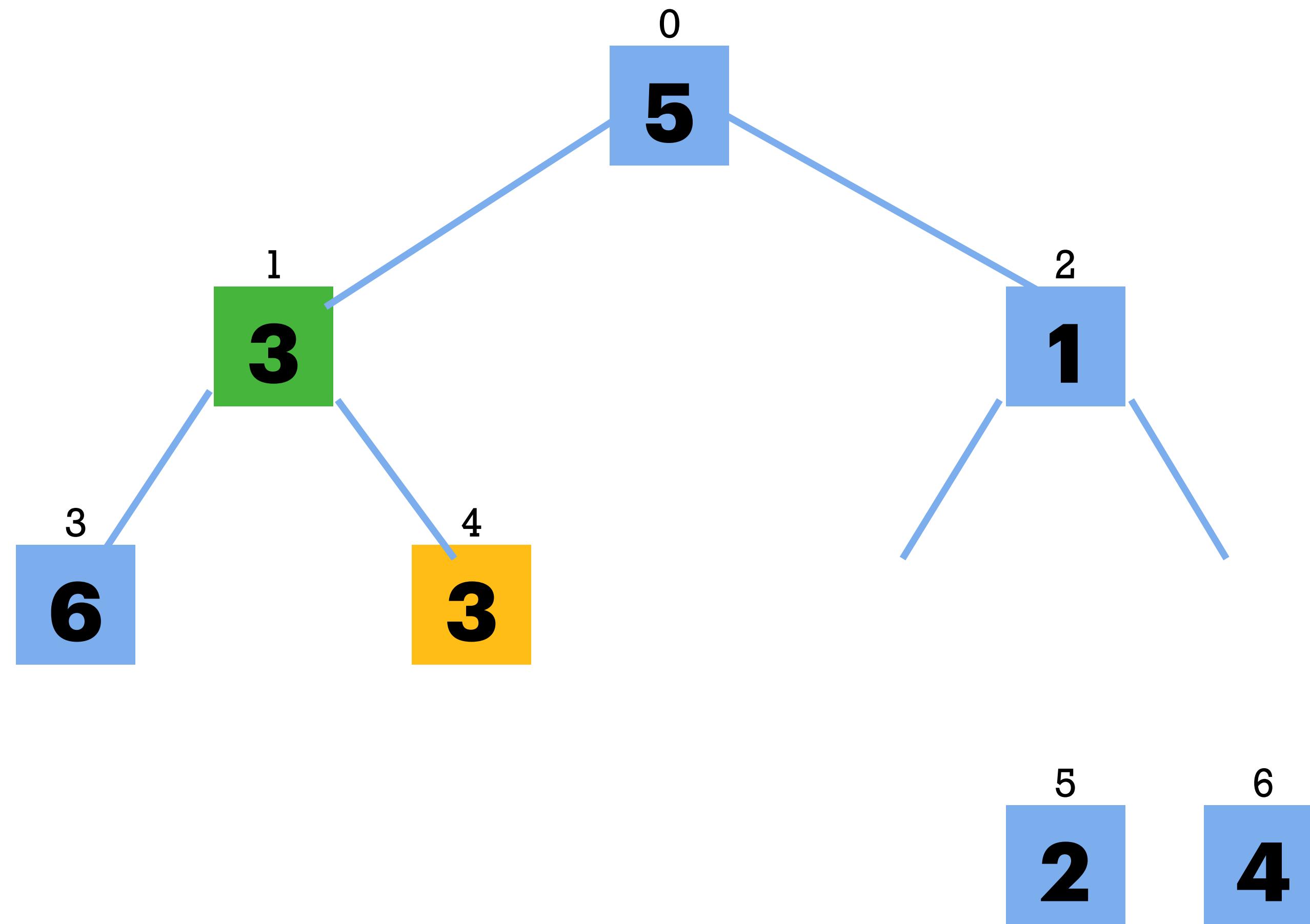
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



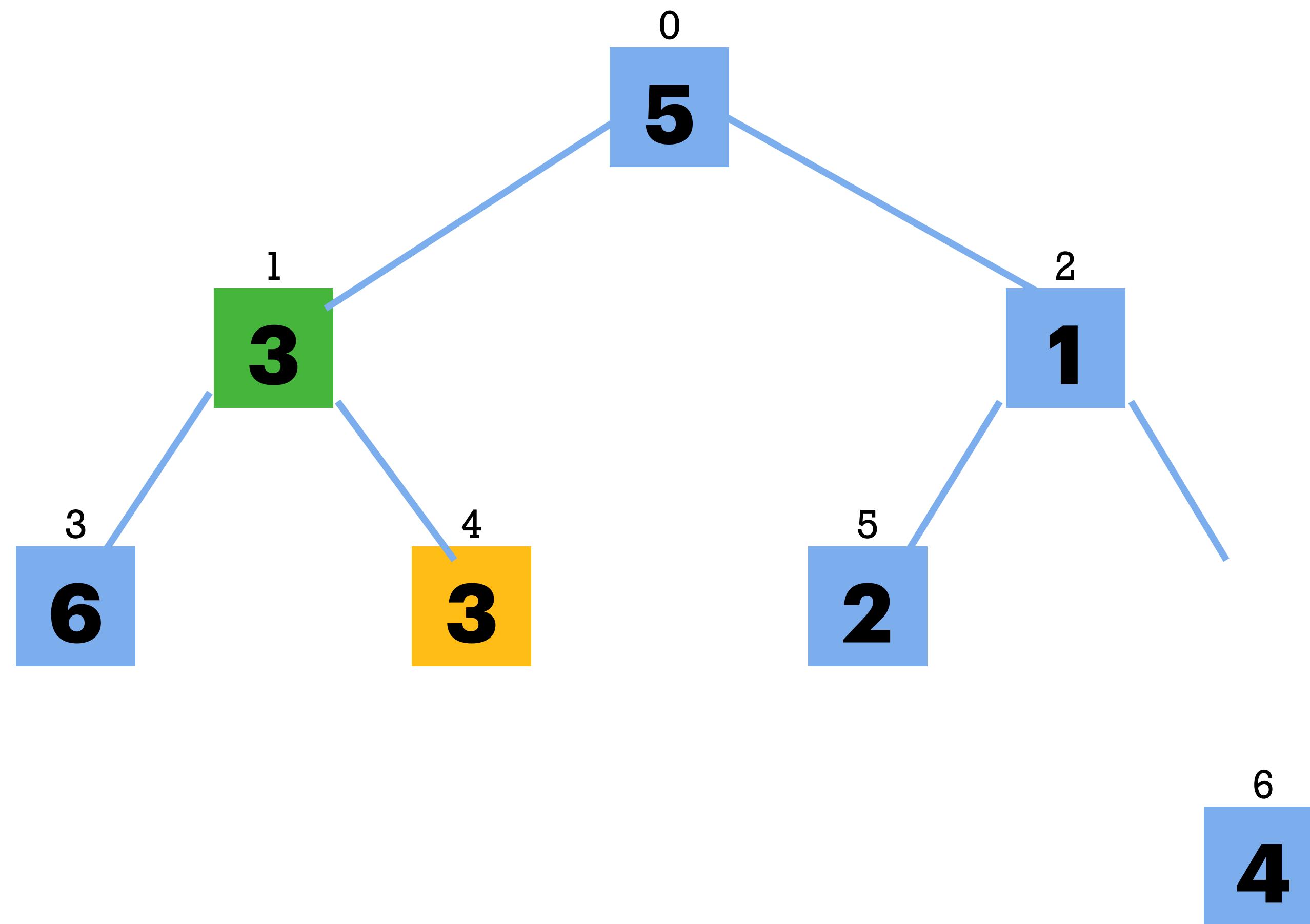
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



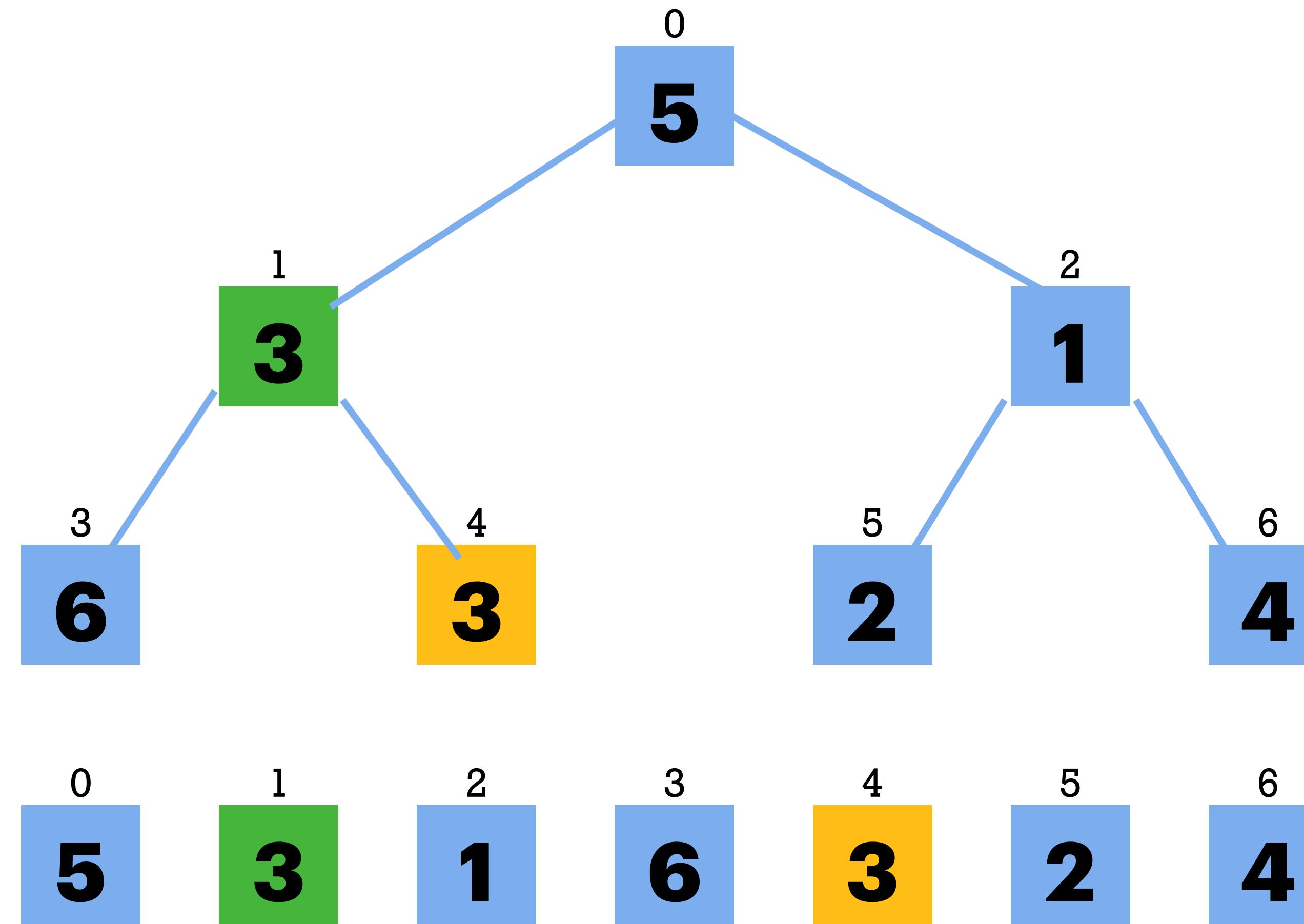
He a p S o r t

- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta

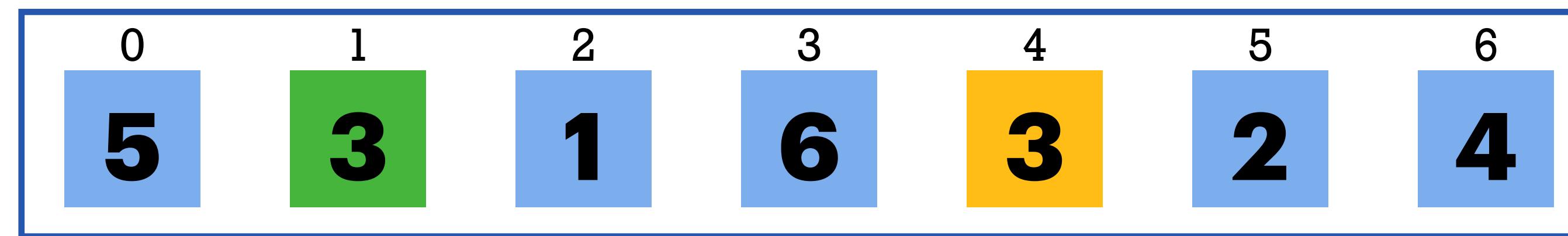
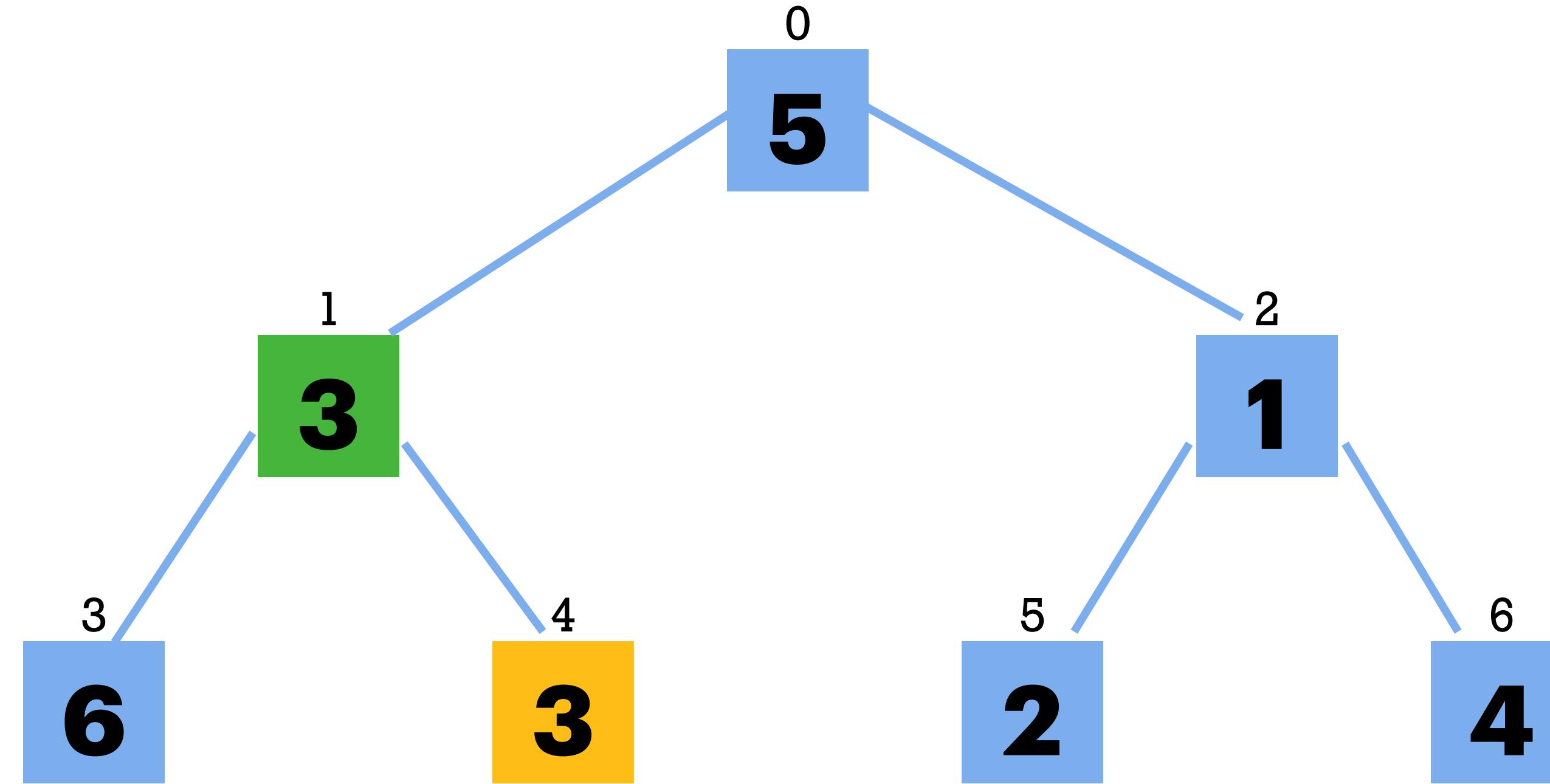


He a p S o r t

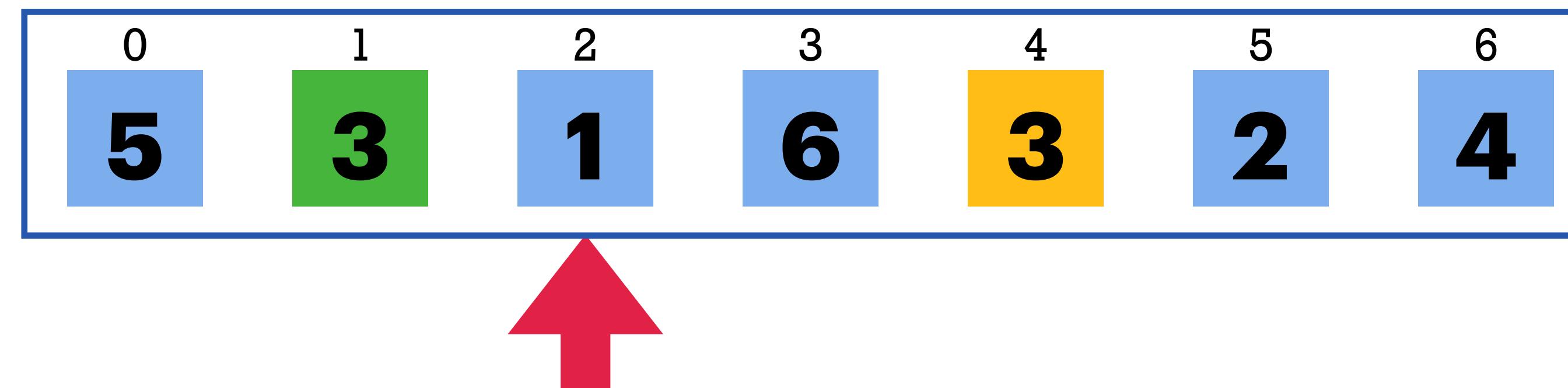
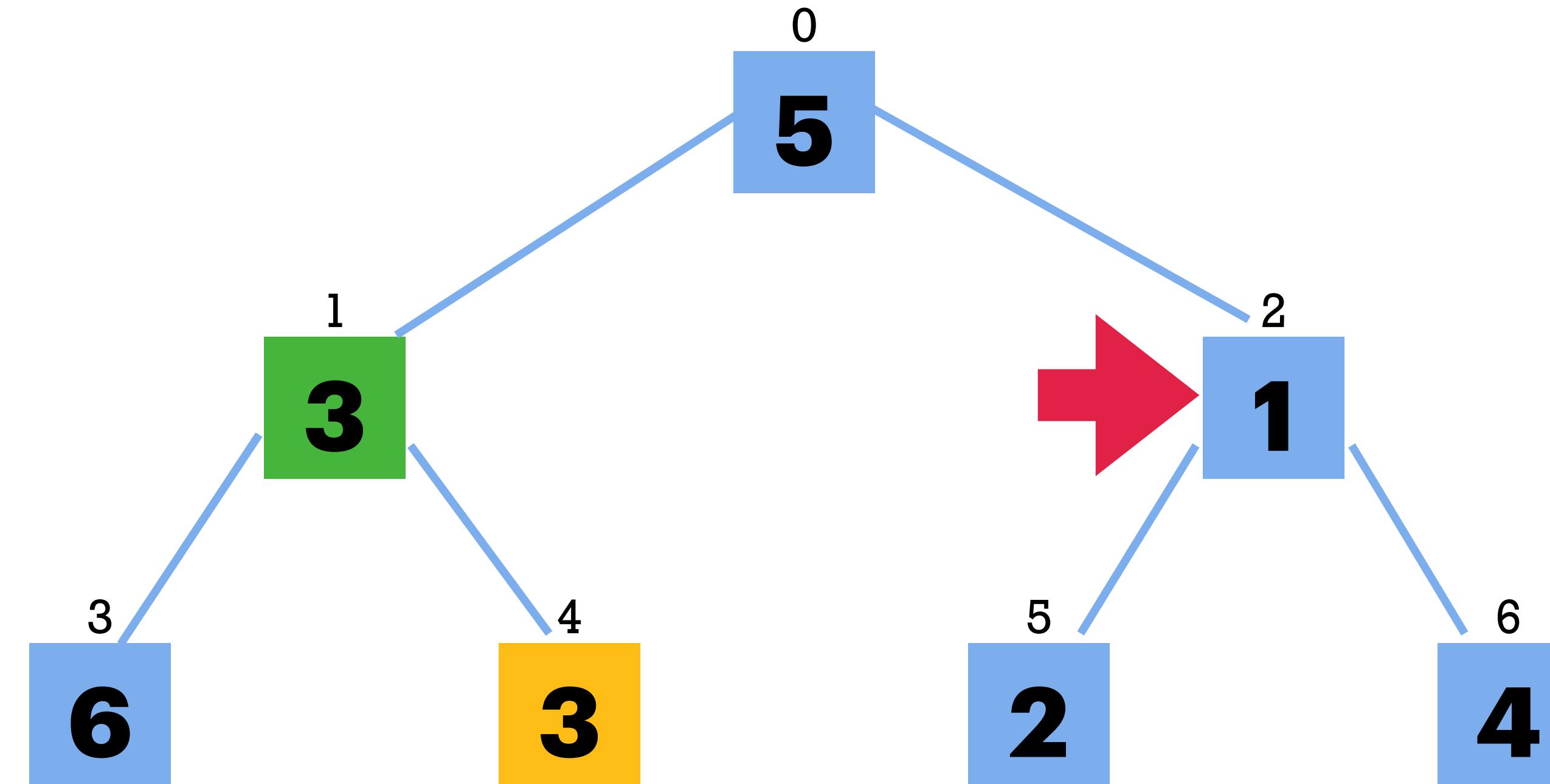
- Elementele din tablou sunt considerate sub formă de arbore binar parțial ordonat de sus în jos și de la stânga la dreapta



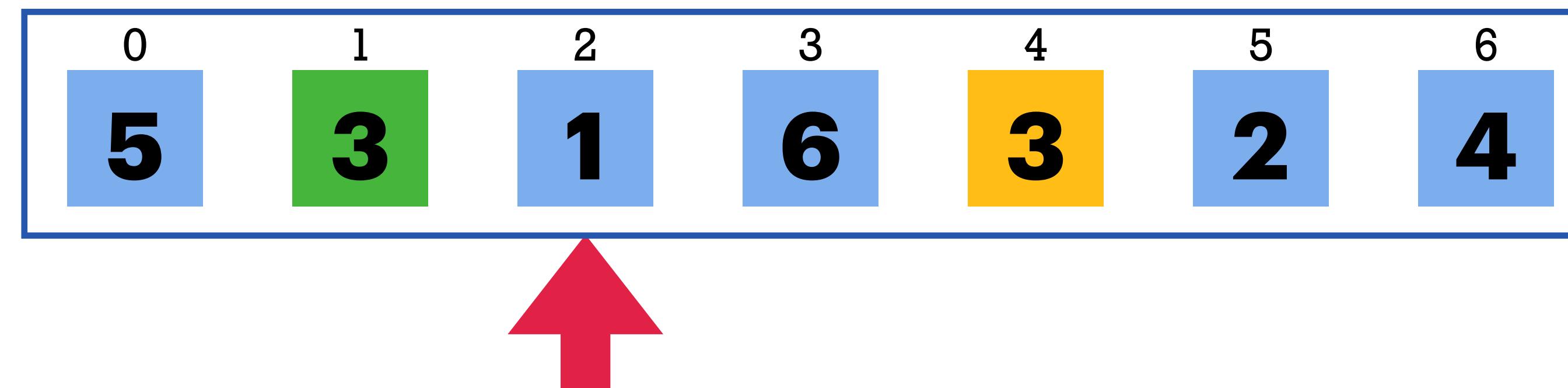
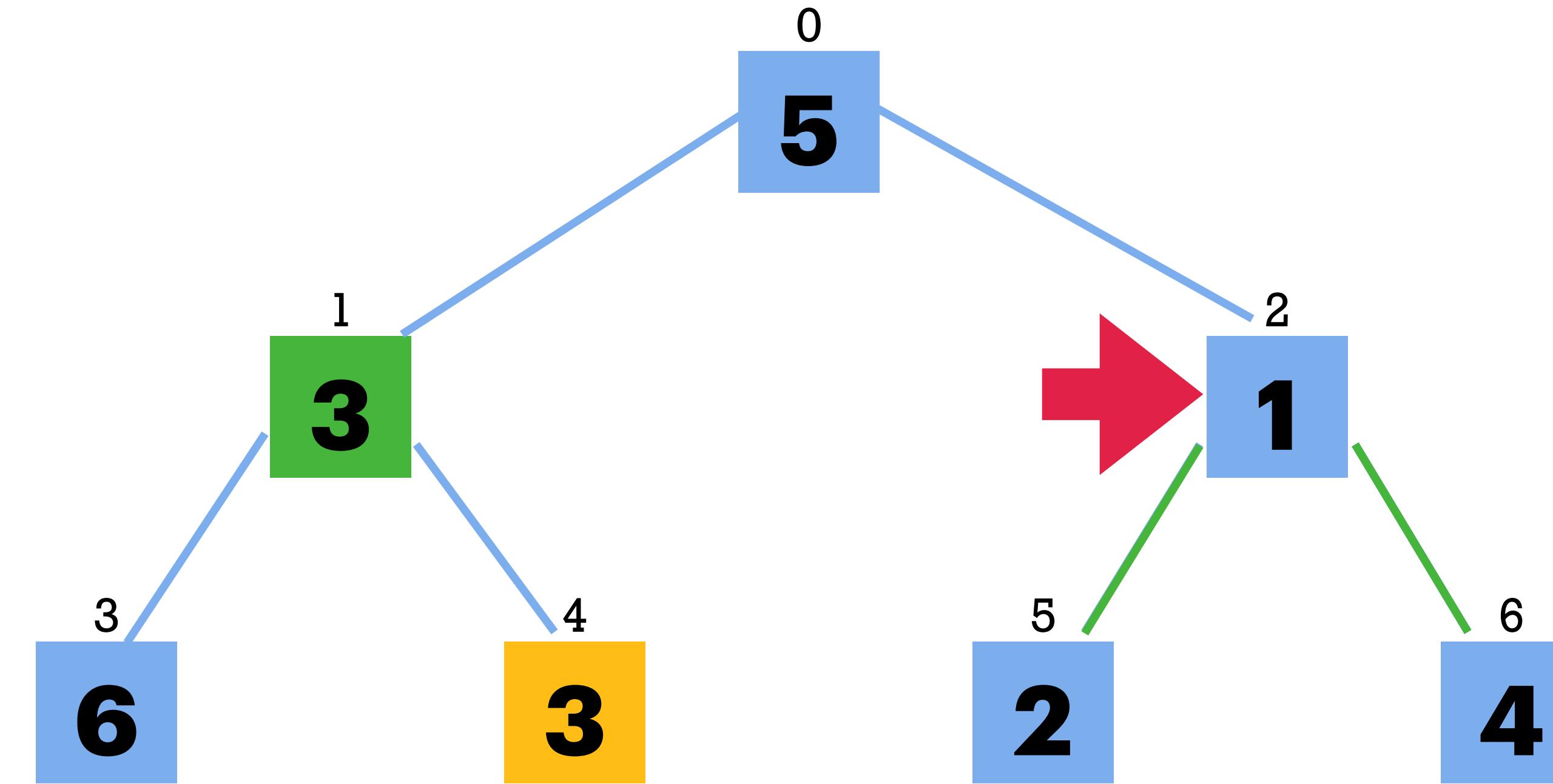
He a p S o r t



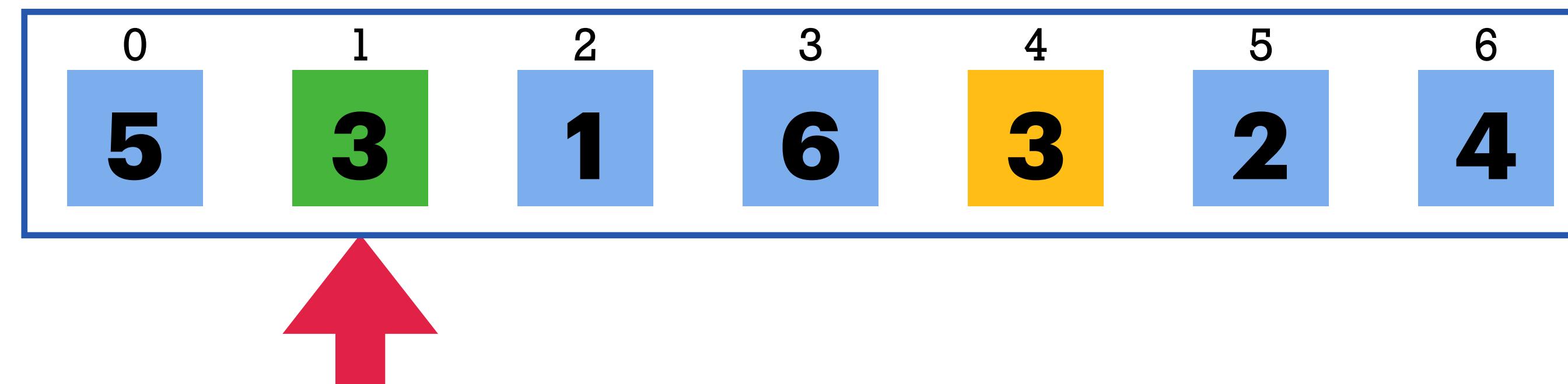
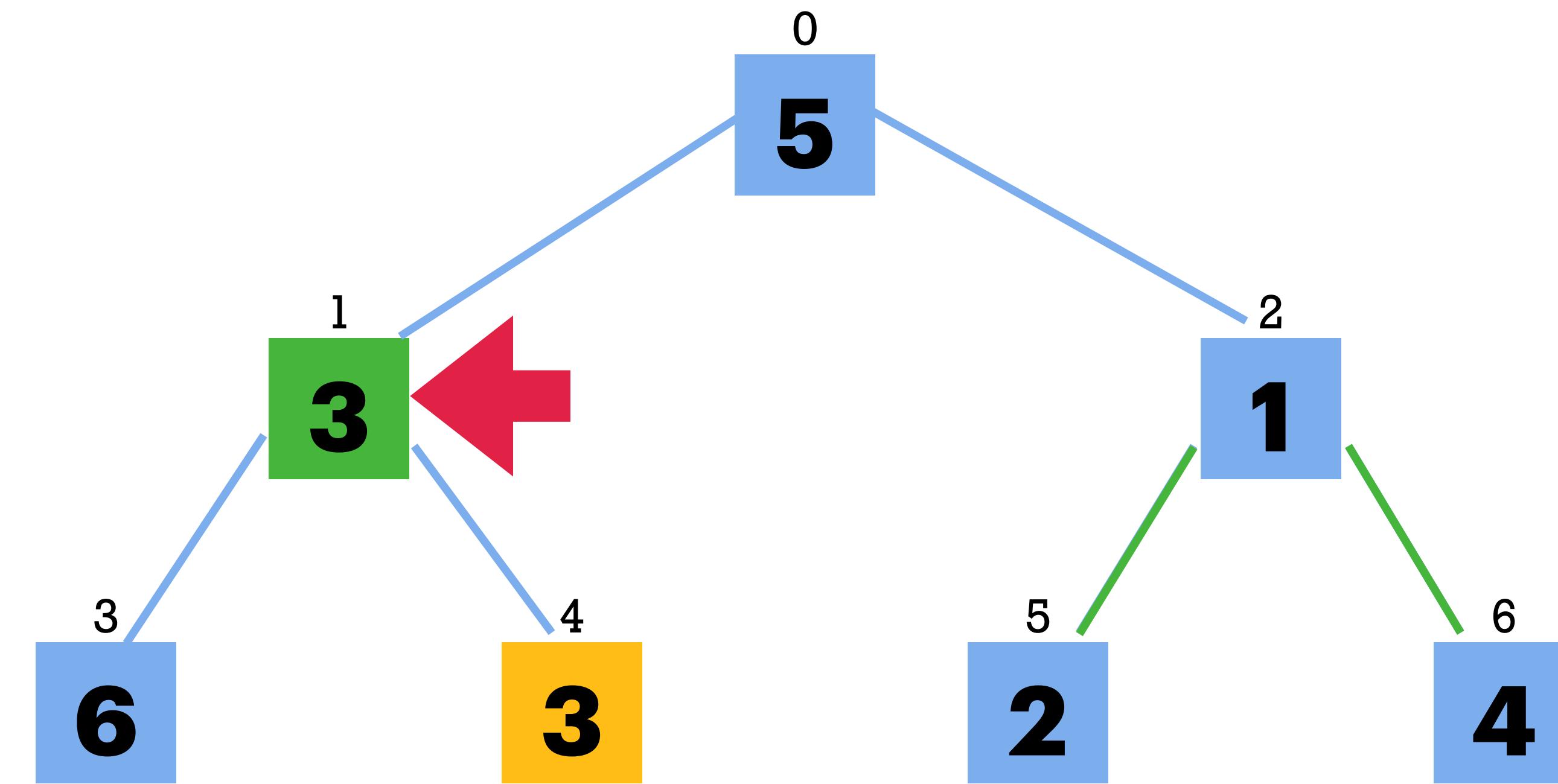
He a p S o r t



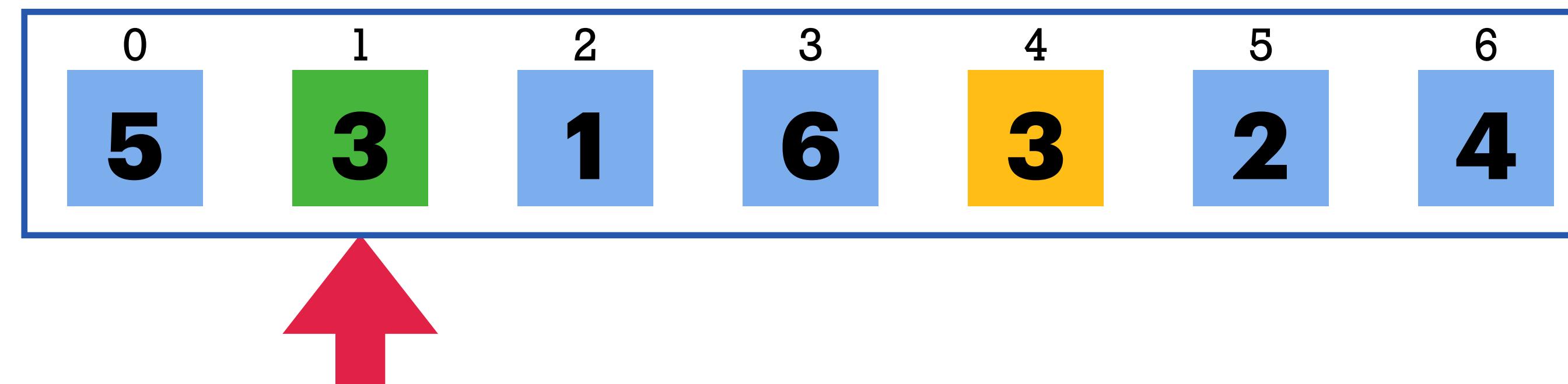
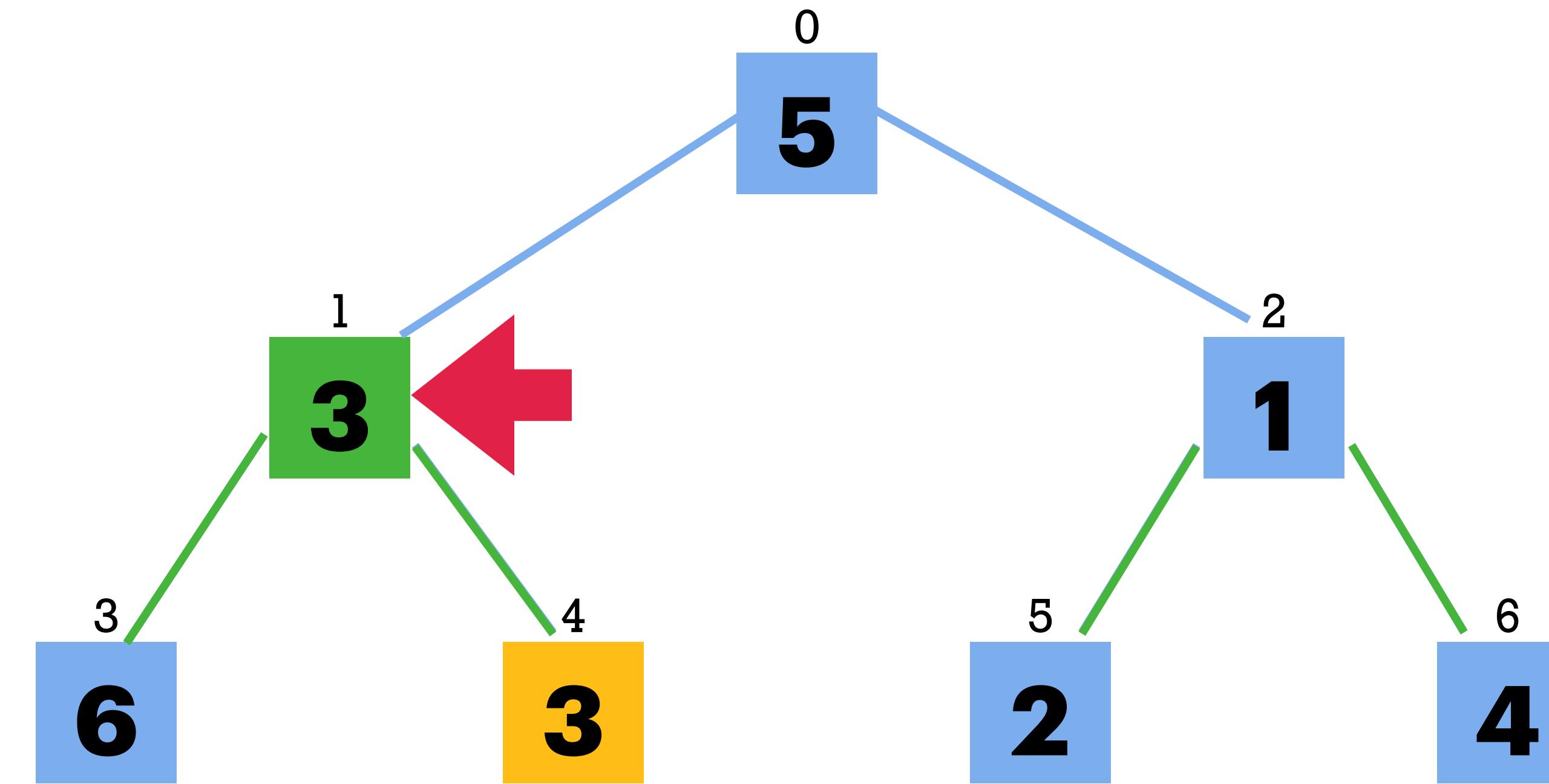
He a p S o r t



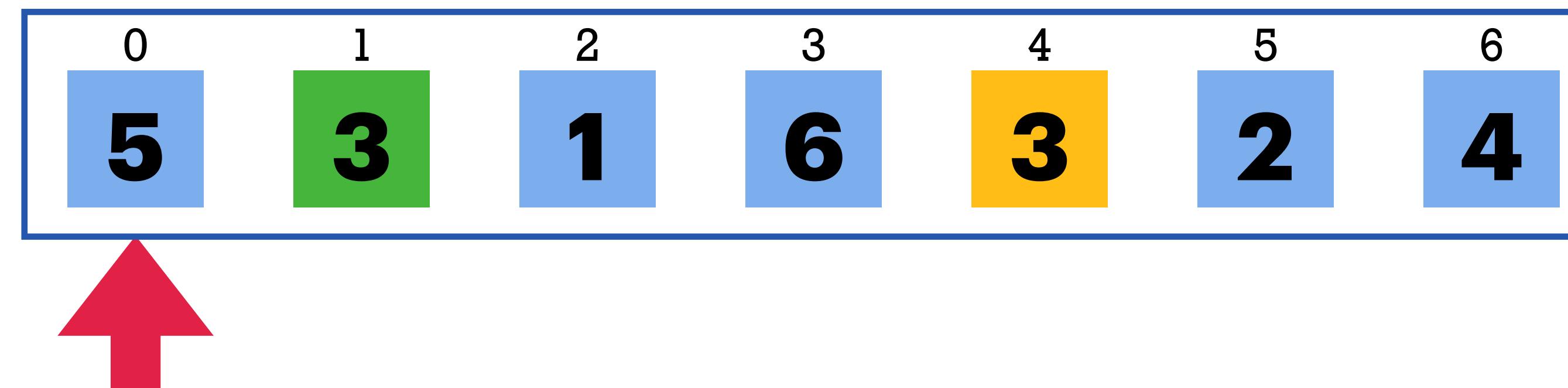
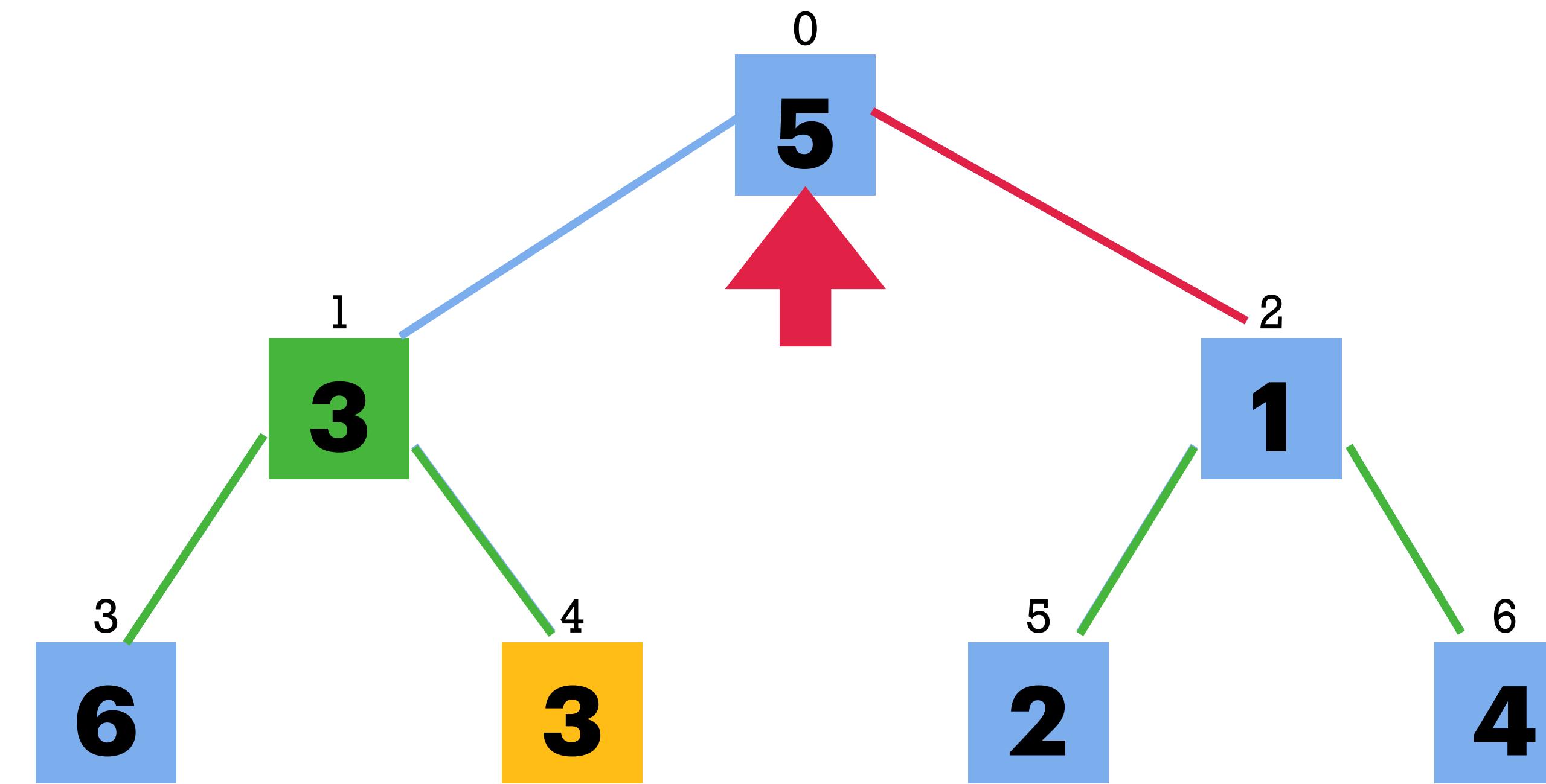
He a p S o r t



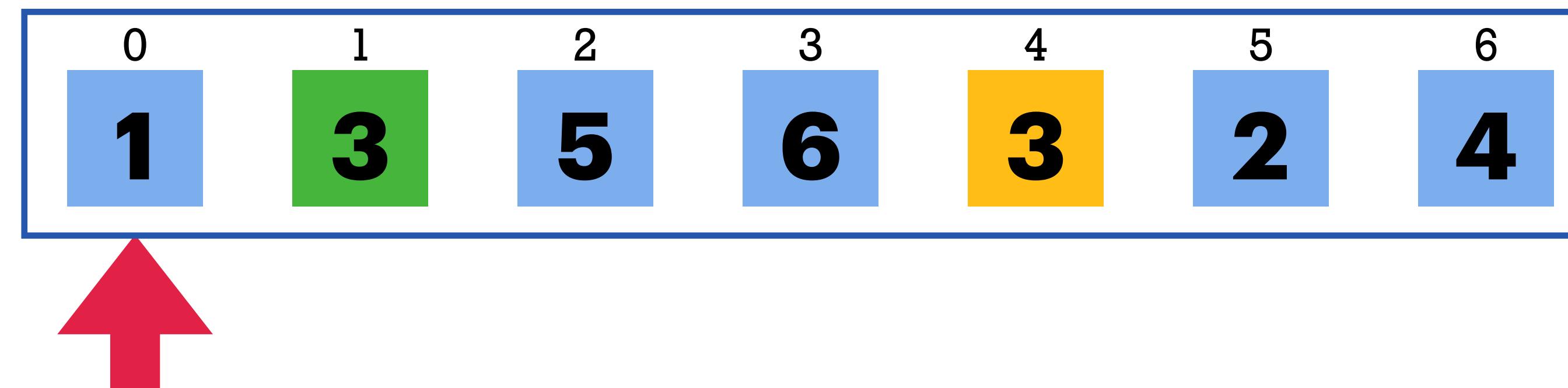
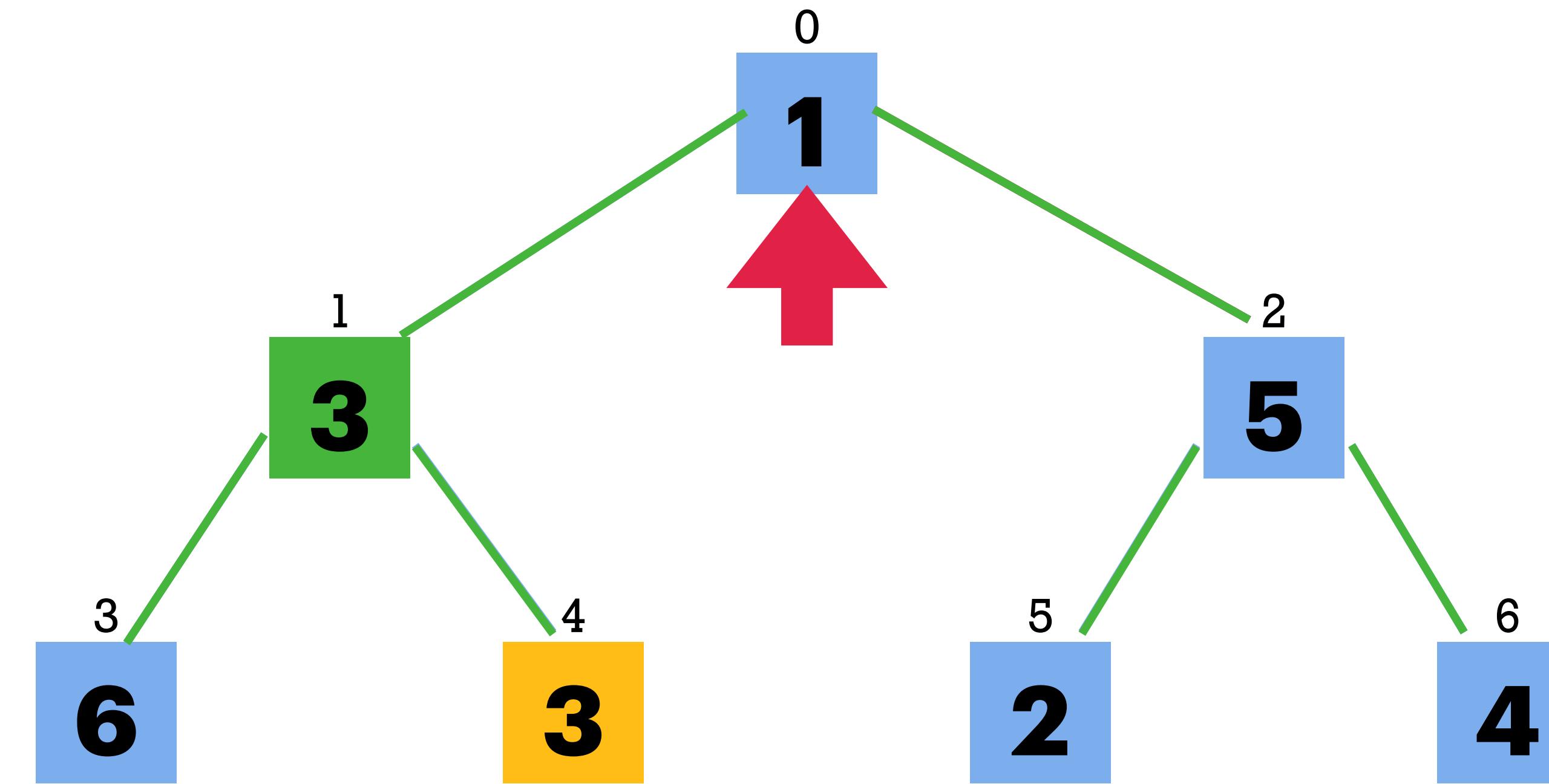
He a p S o r t



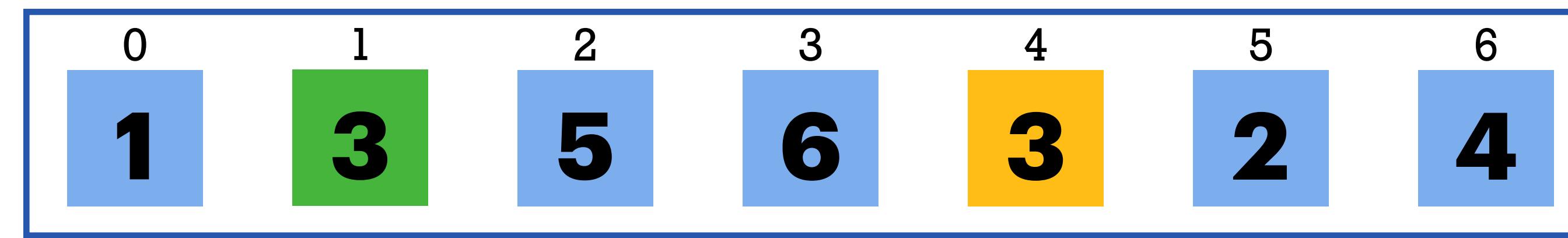
He a p S o r t



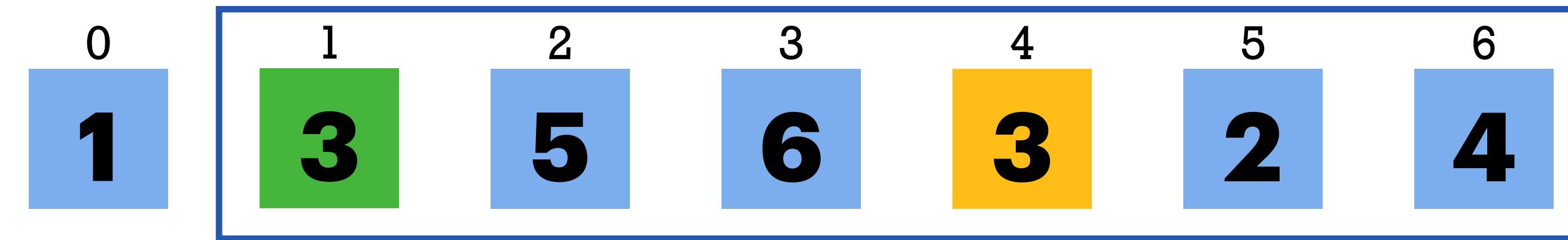
He a p S o r t



He a p S o r t

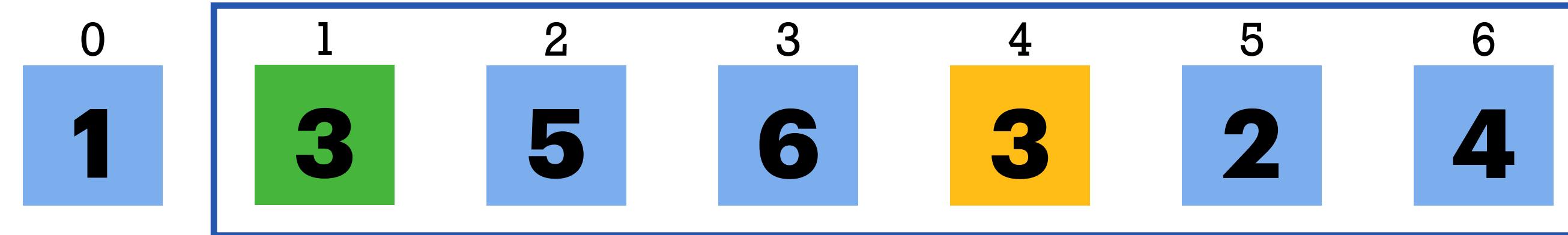


He a p S o r t

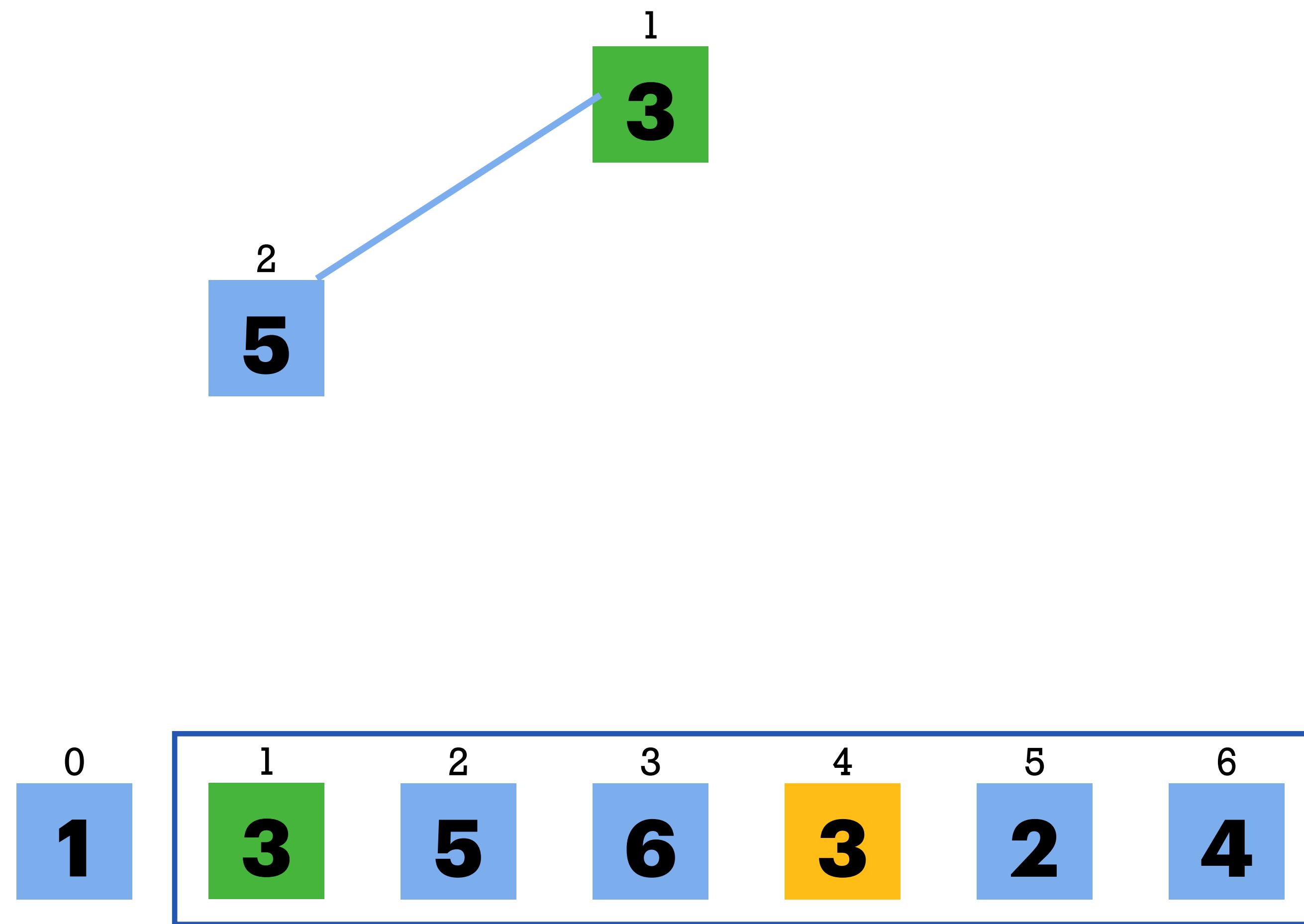


He a p S o r t

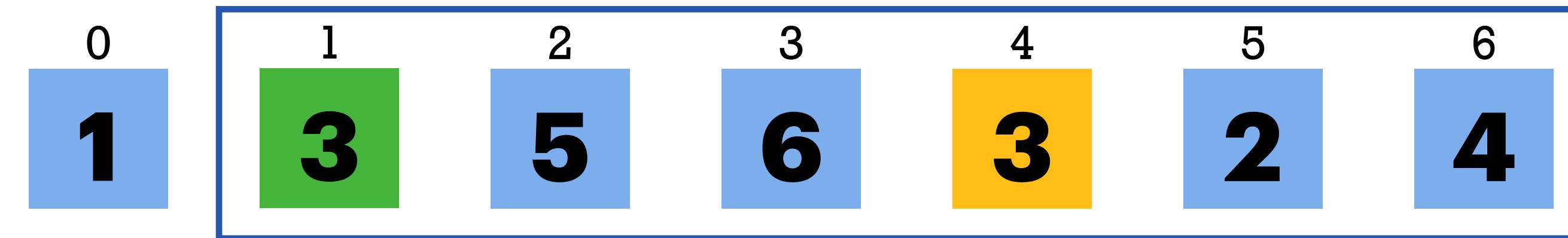
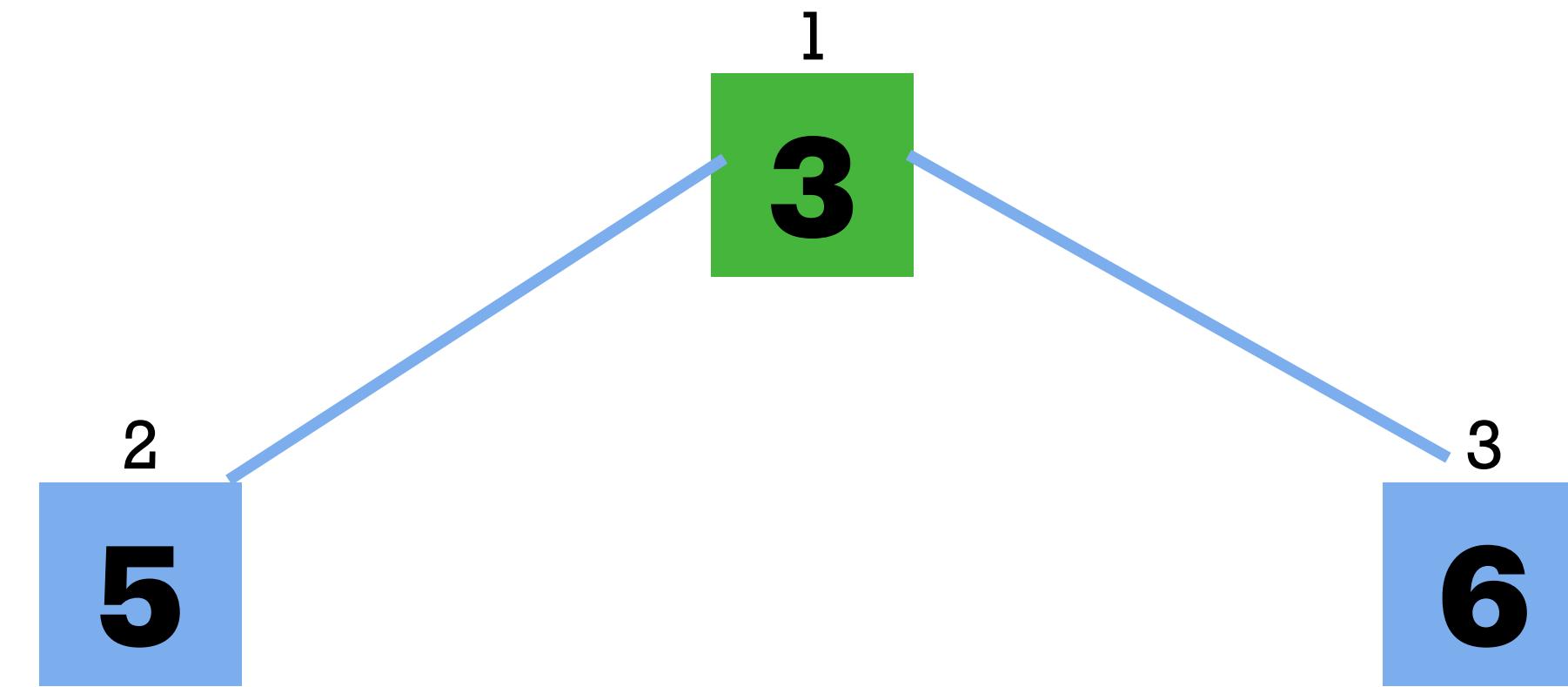
1
3



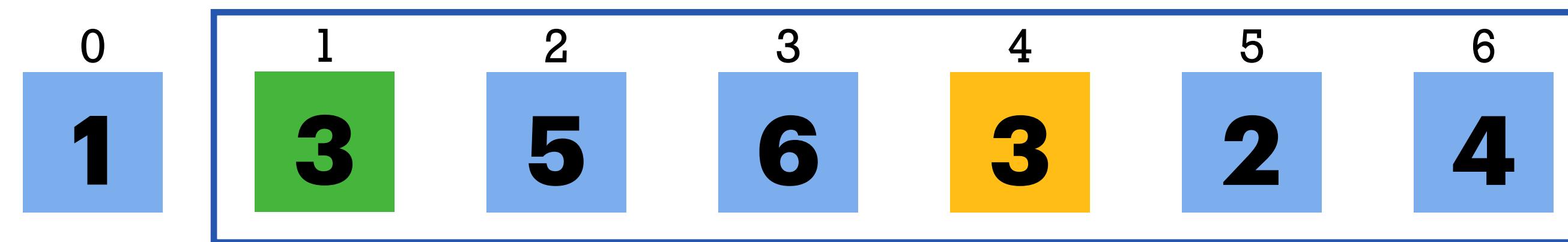
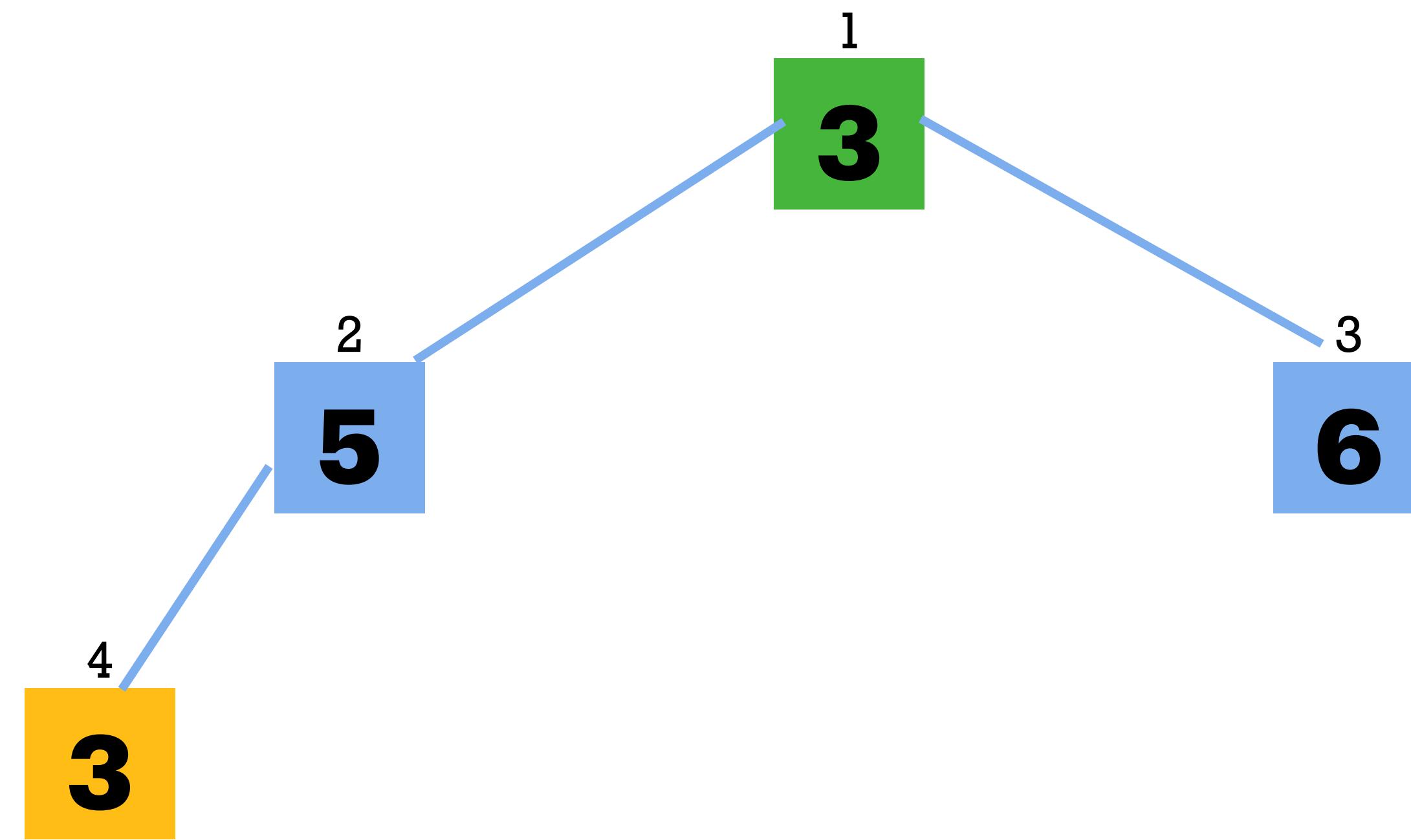
He a p S o r t



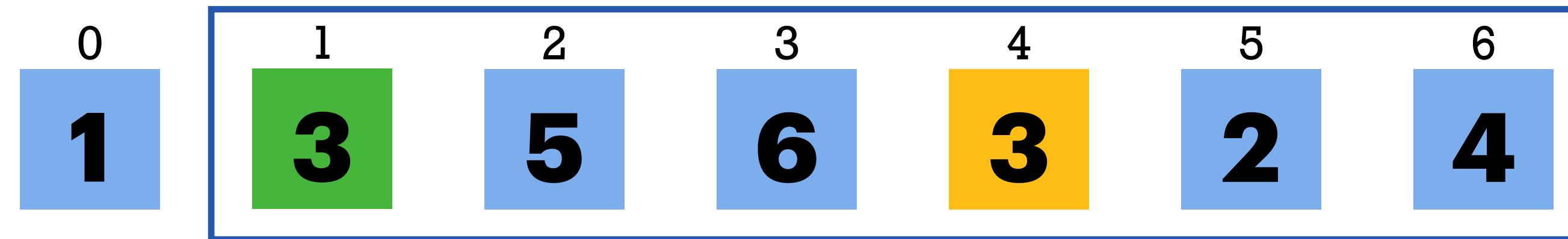
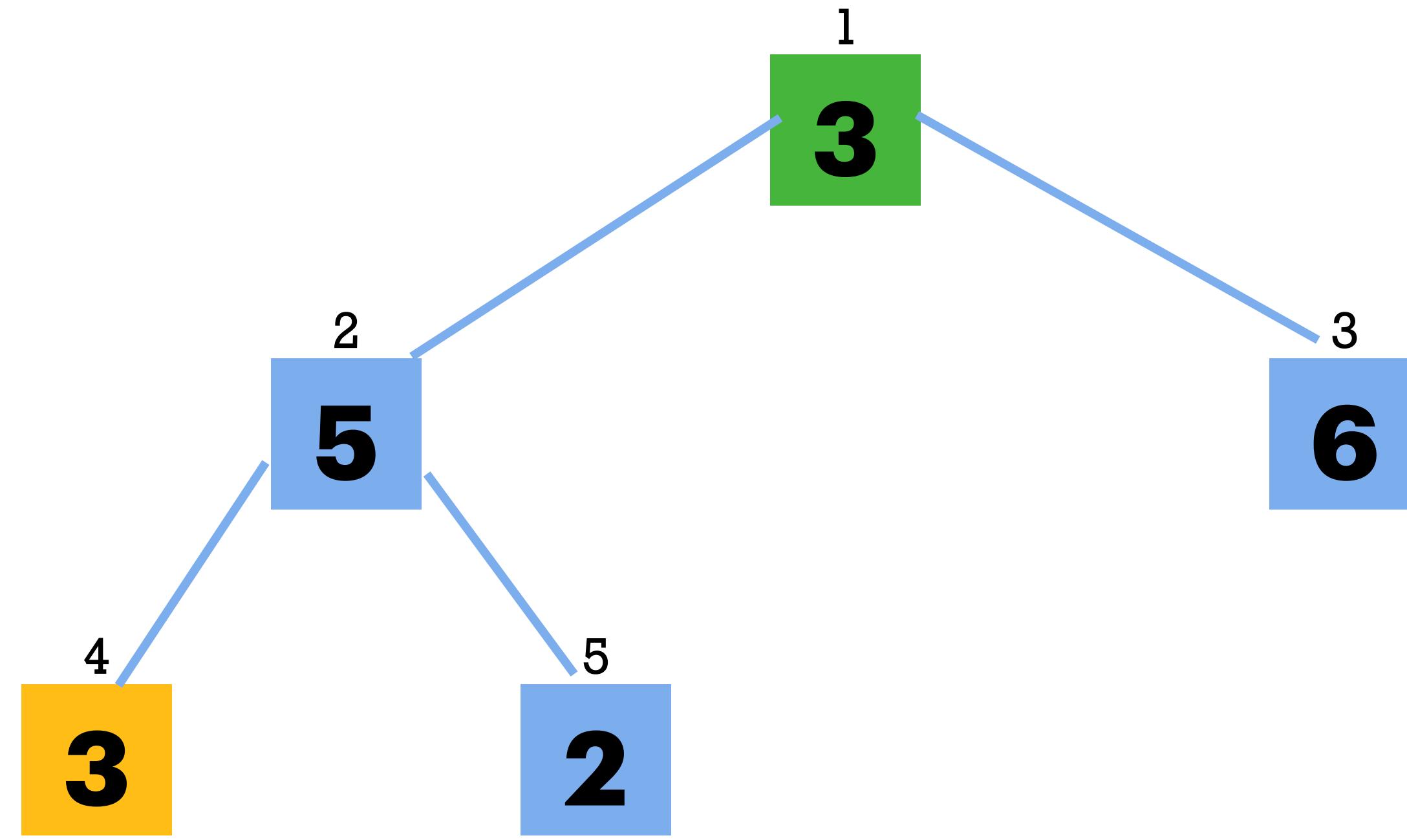
He a p S o r t



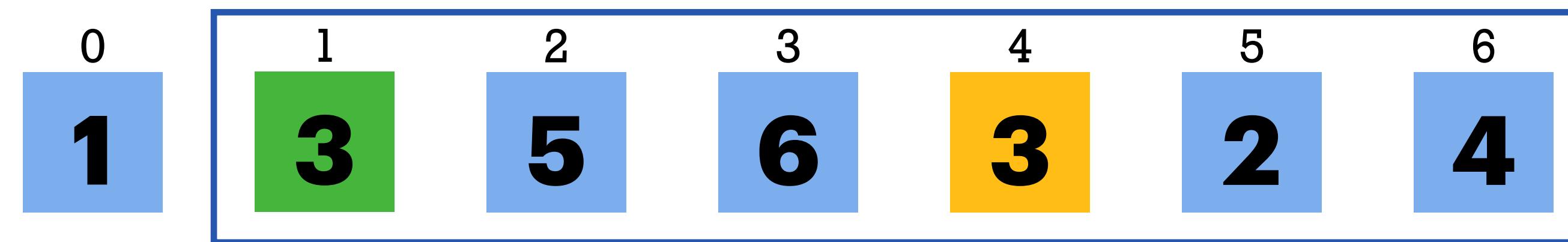
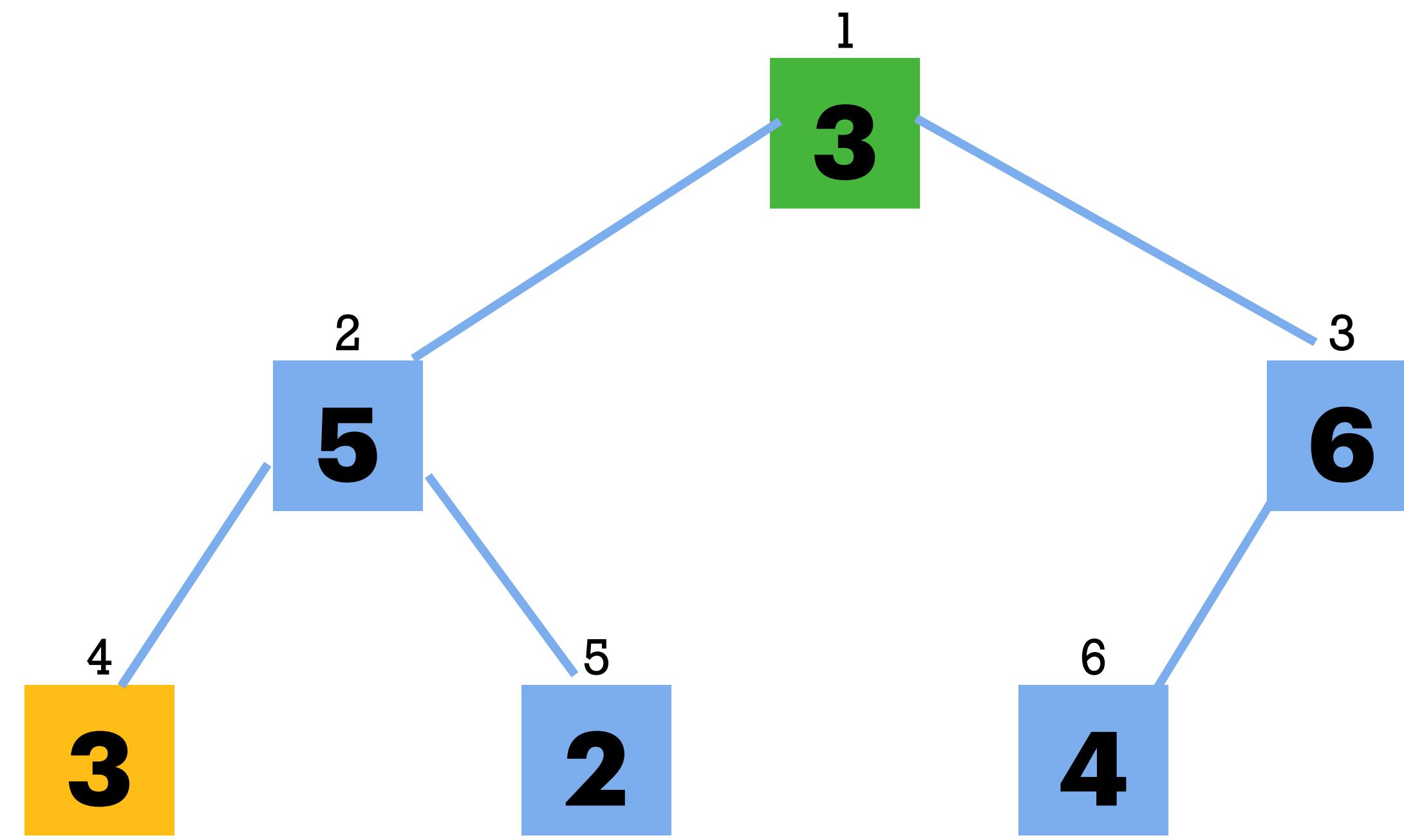
He a p S o r t



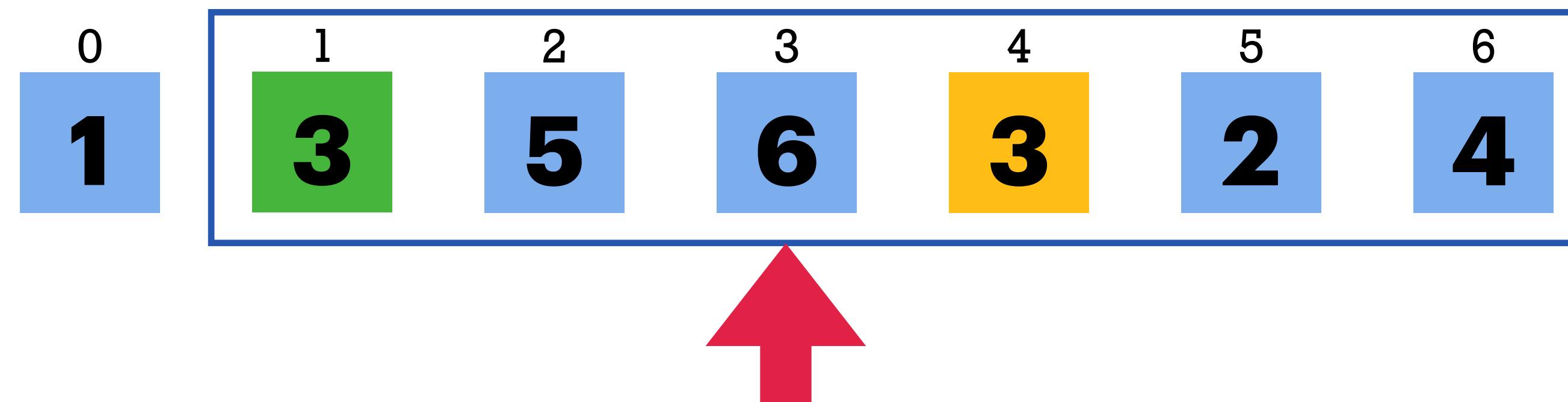
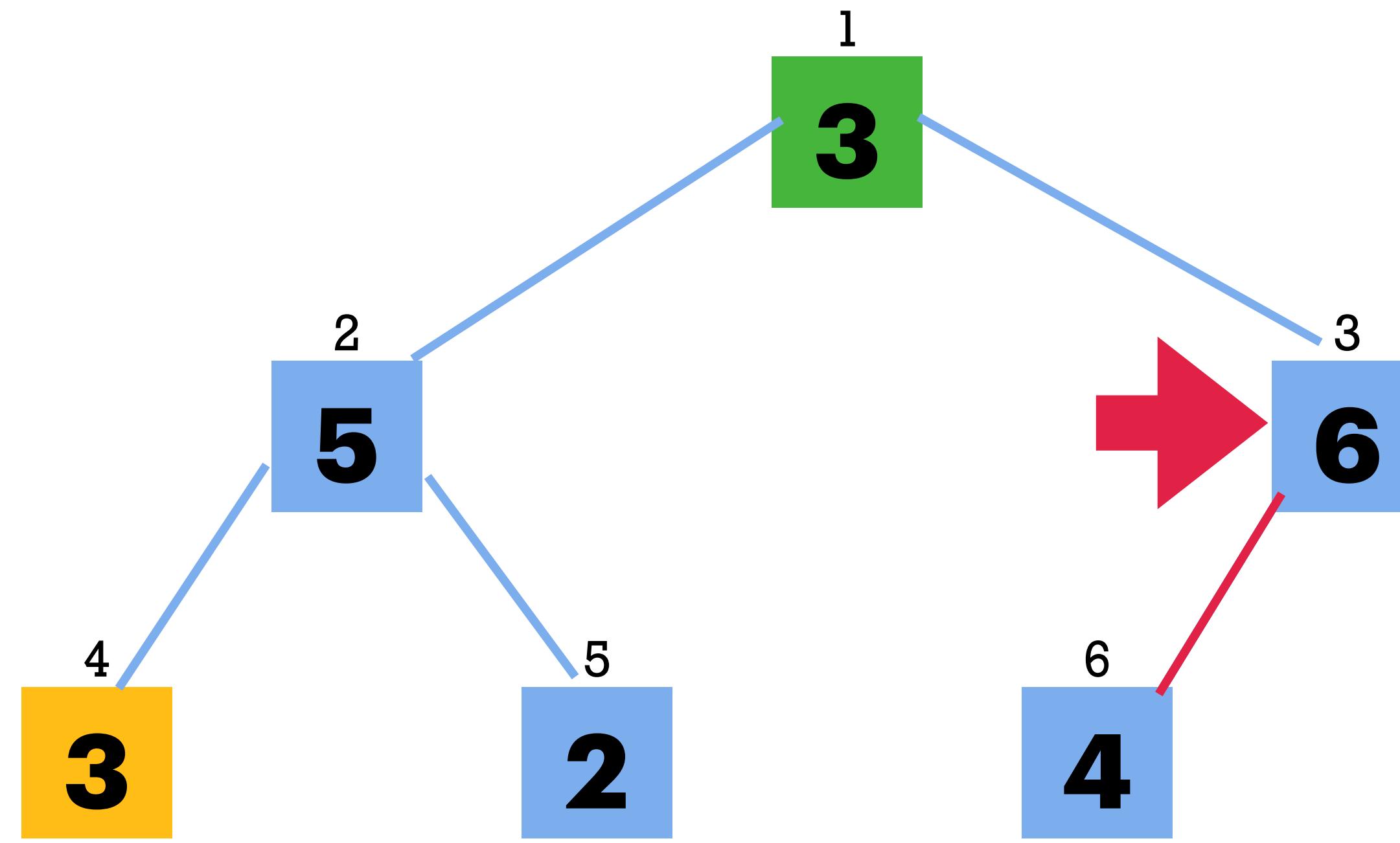
He a p S o r t



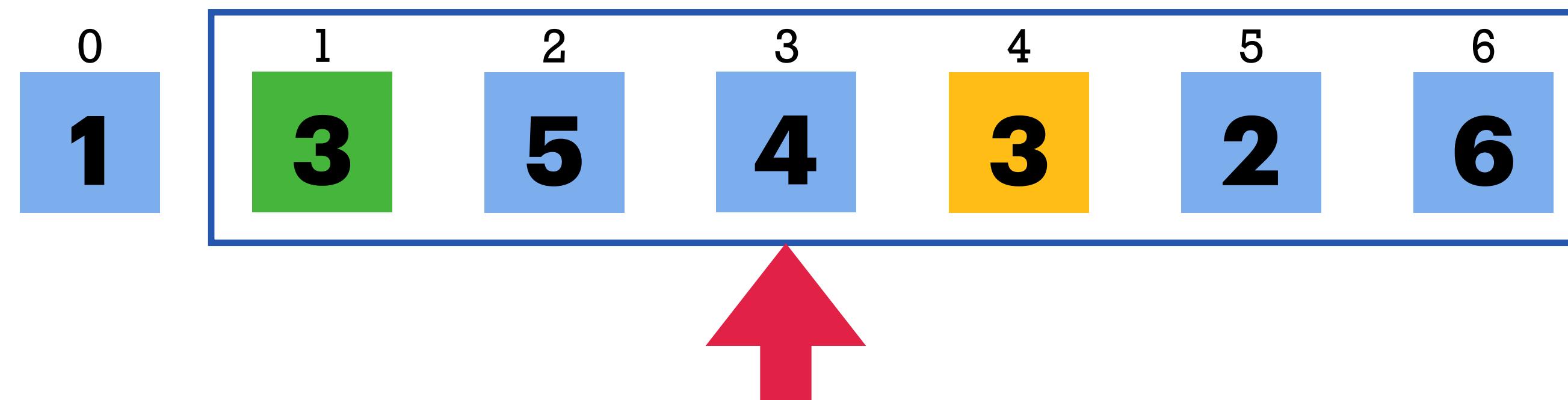
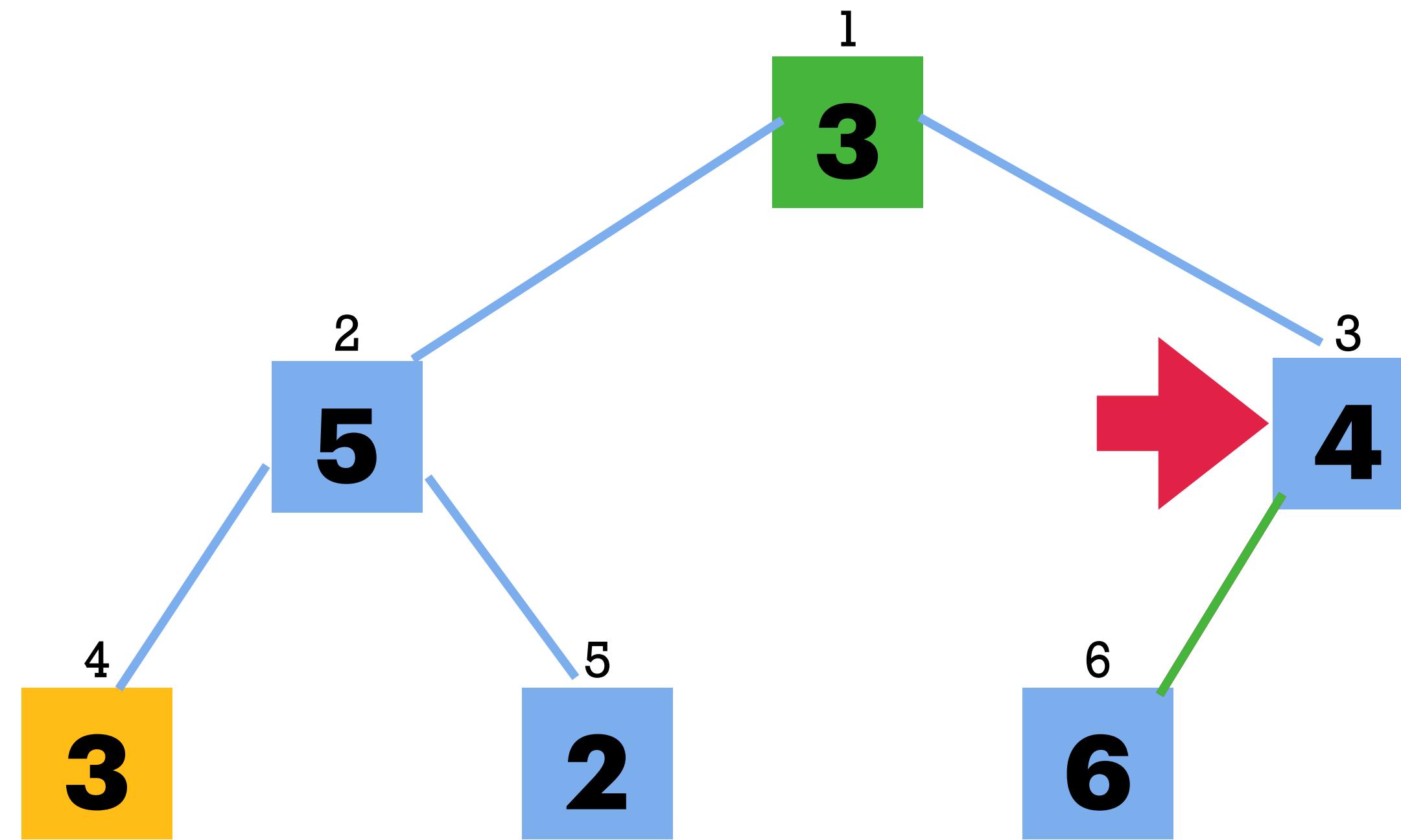
He a p S o r t



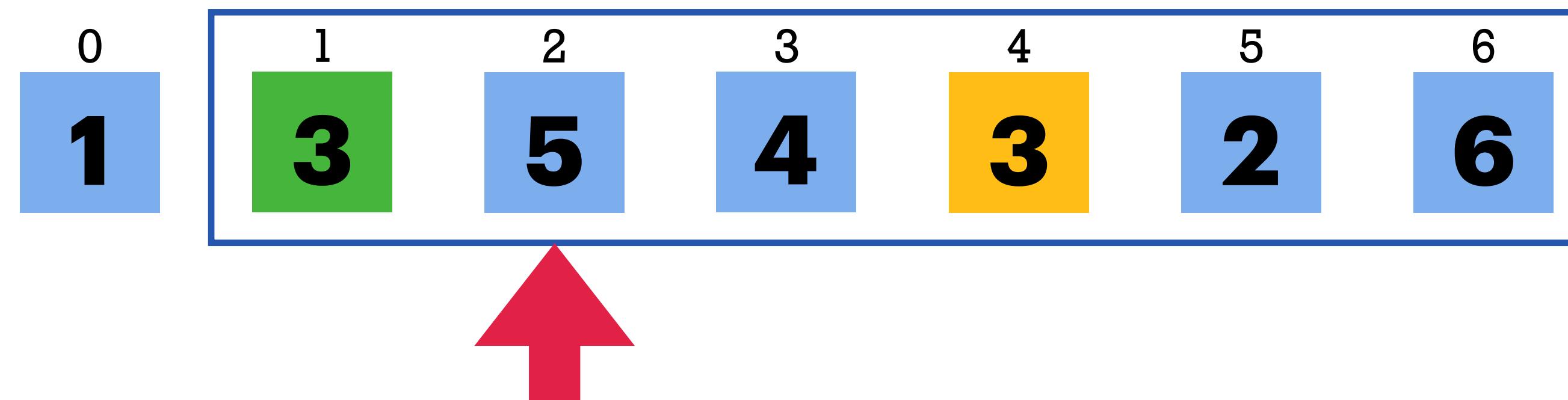
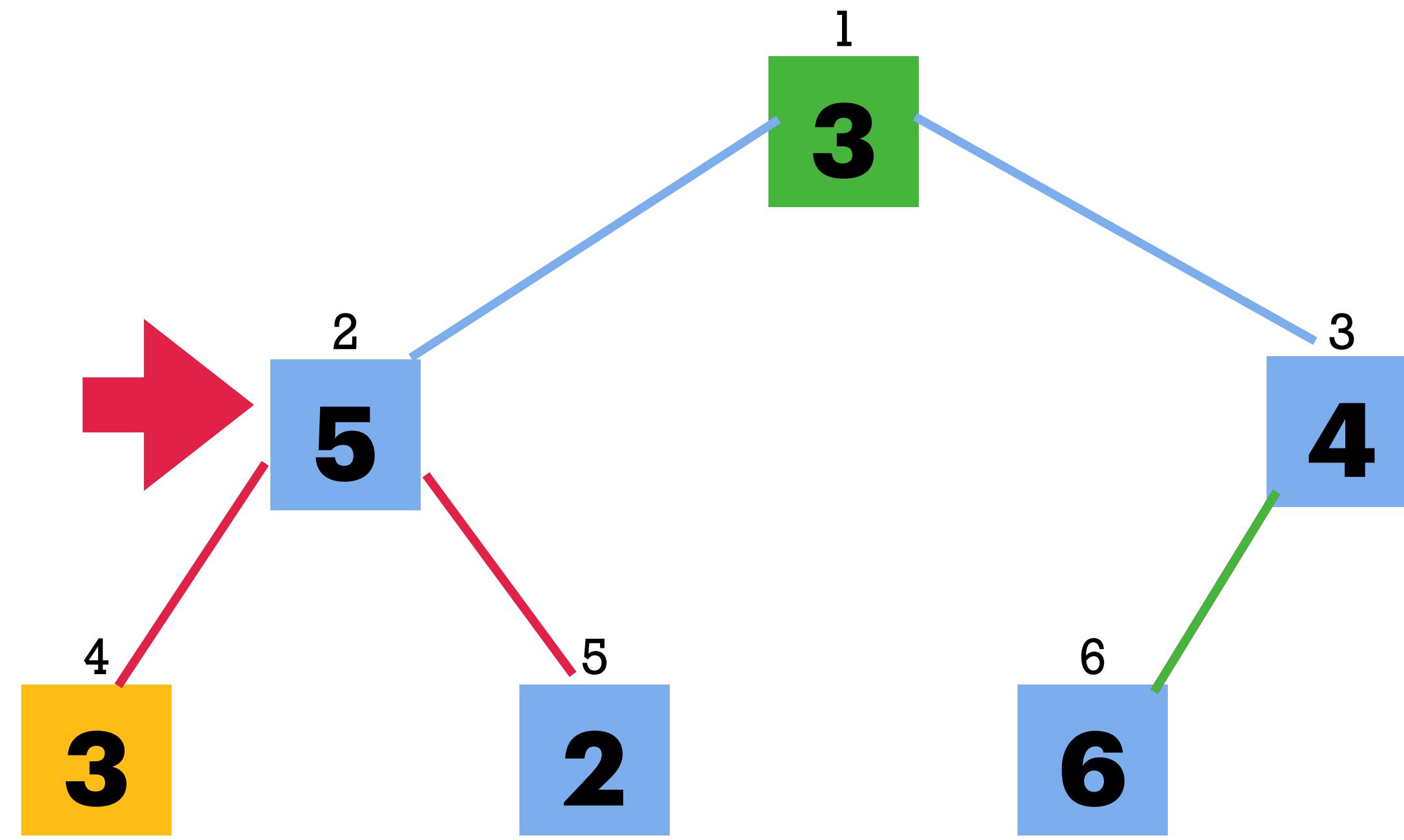
He a p S o r t



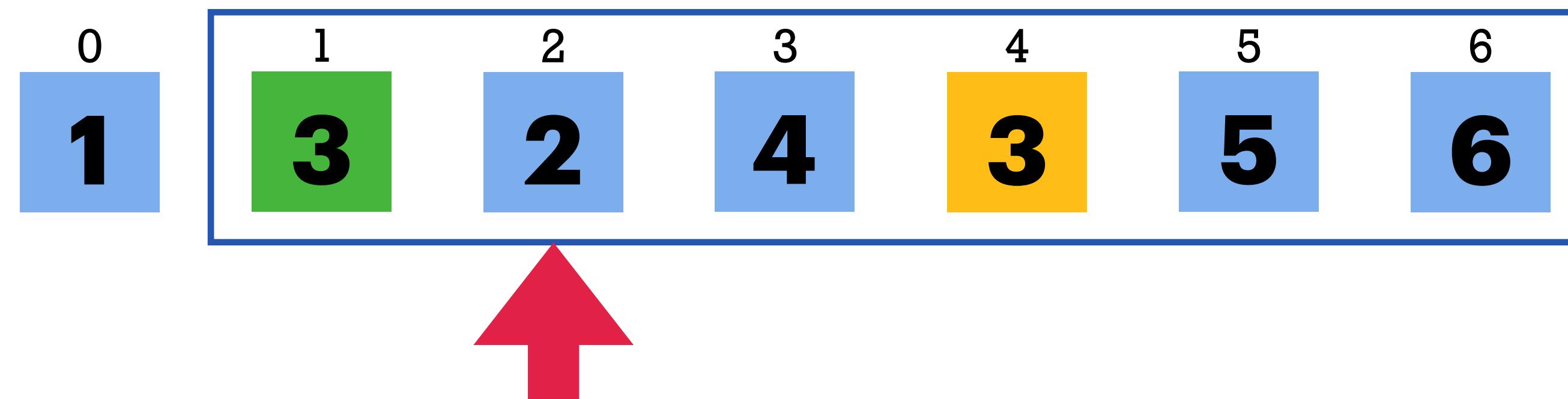
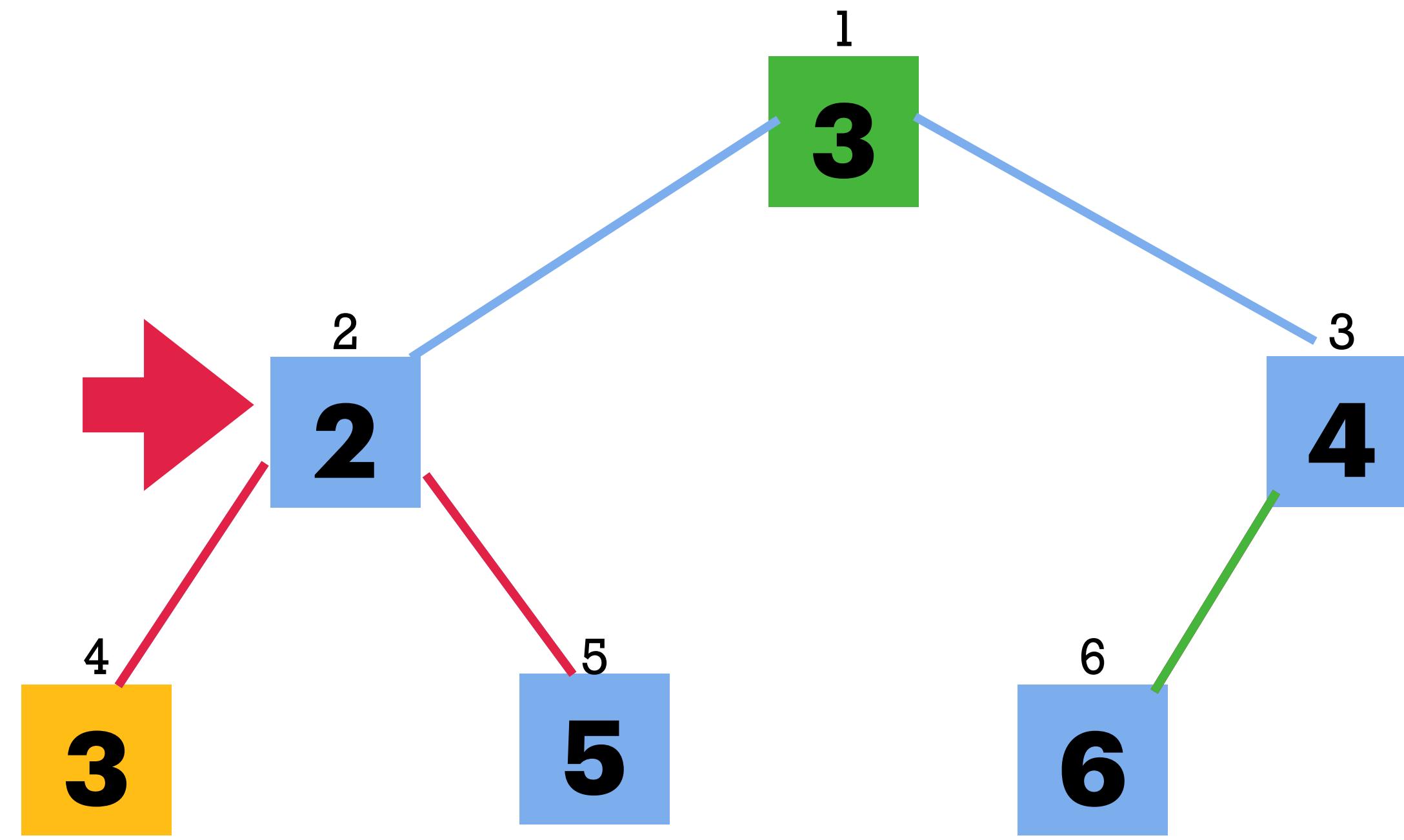
He a p S o r t



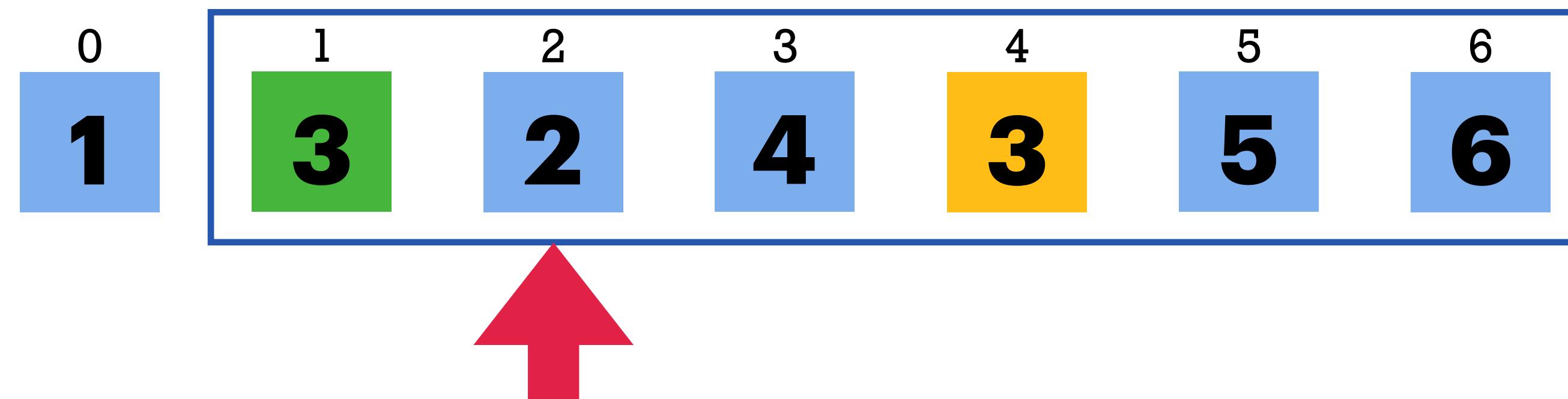
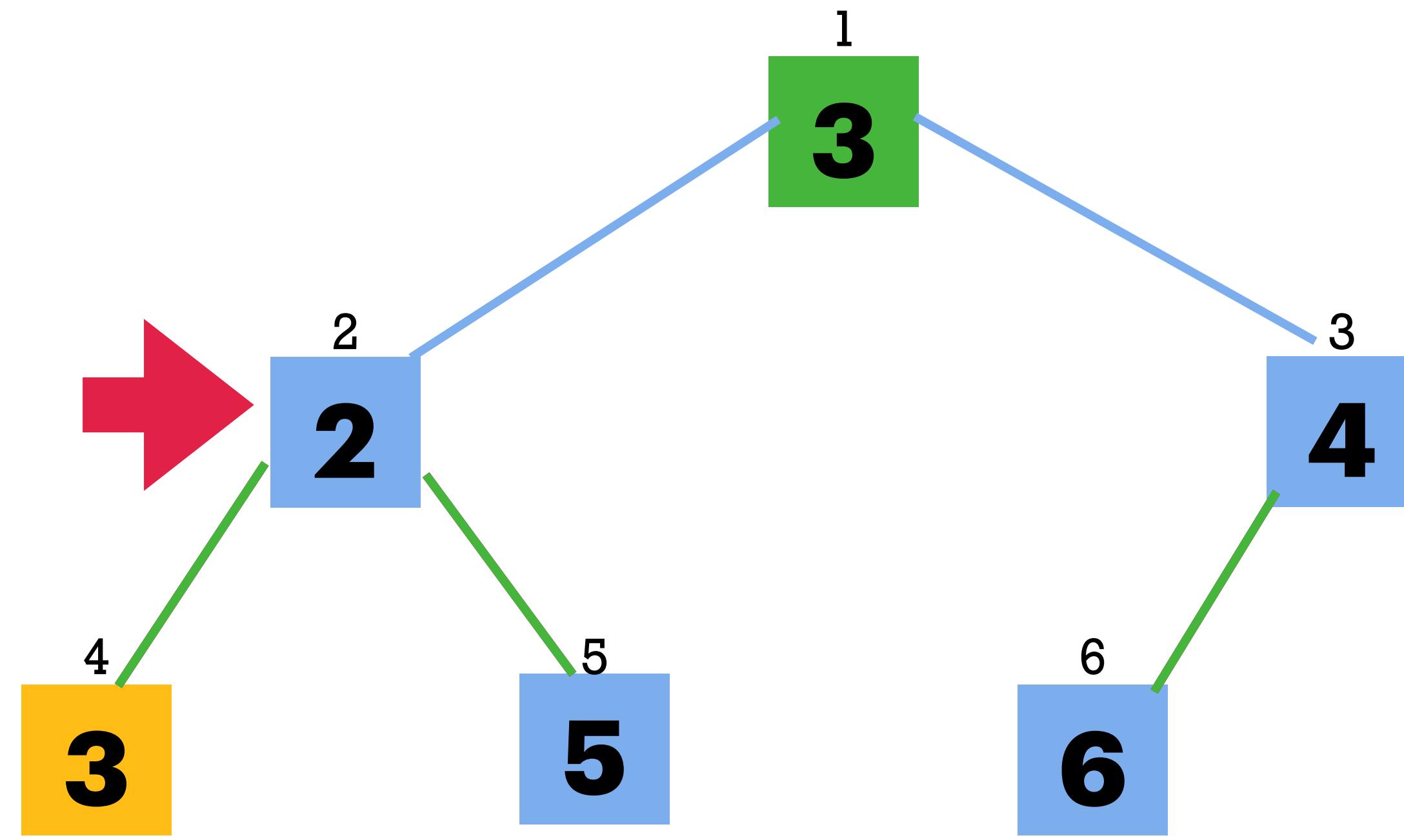
He a p S o r t



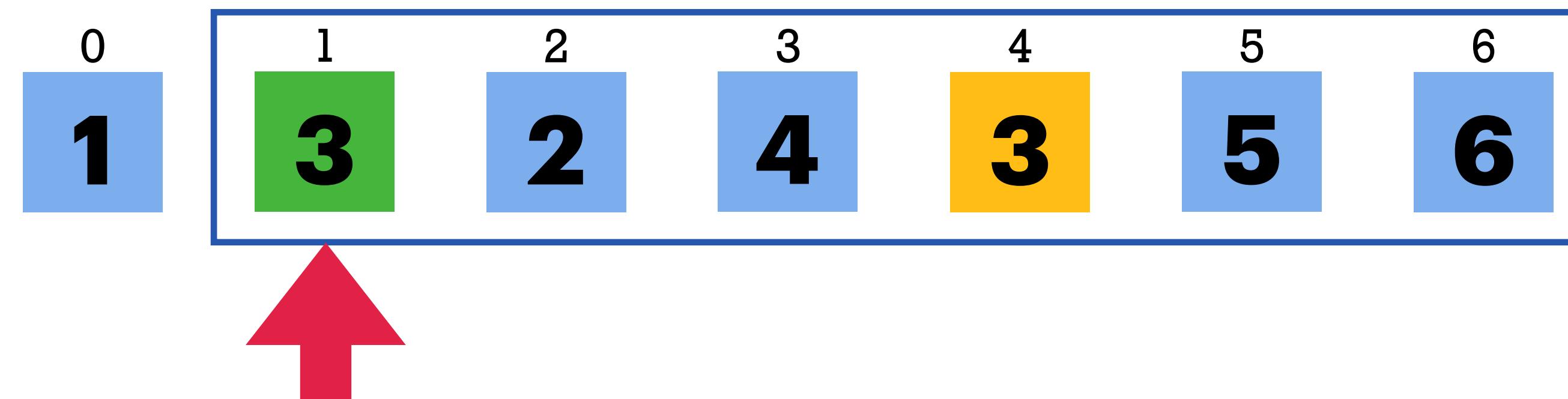
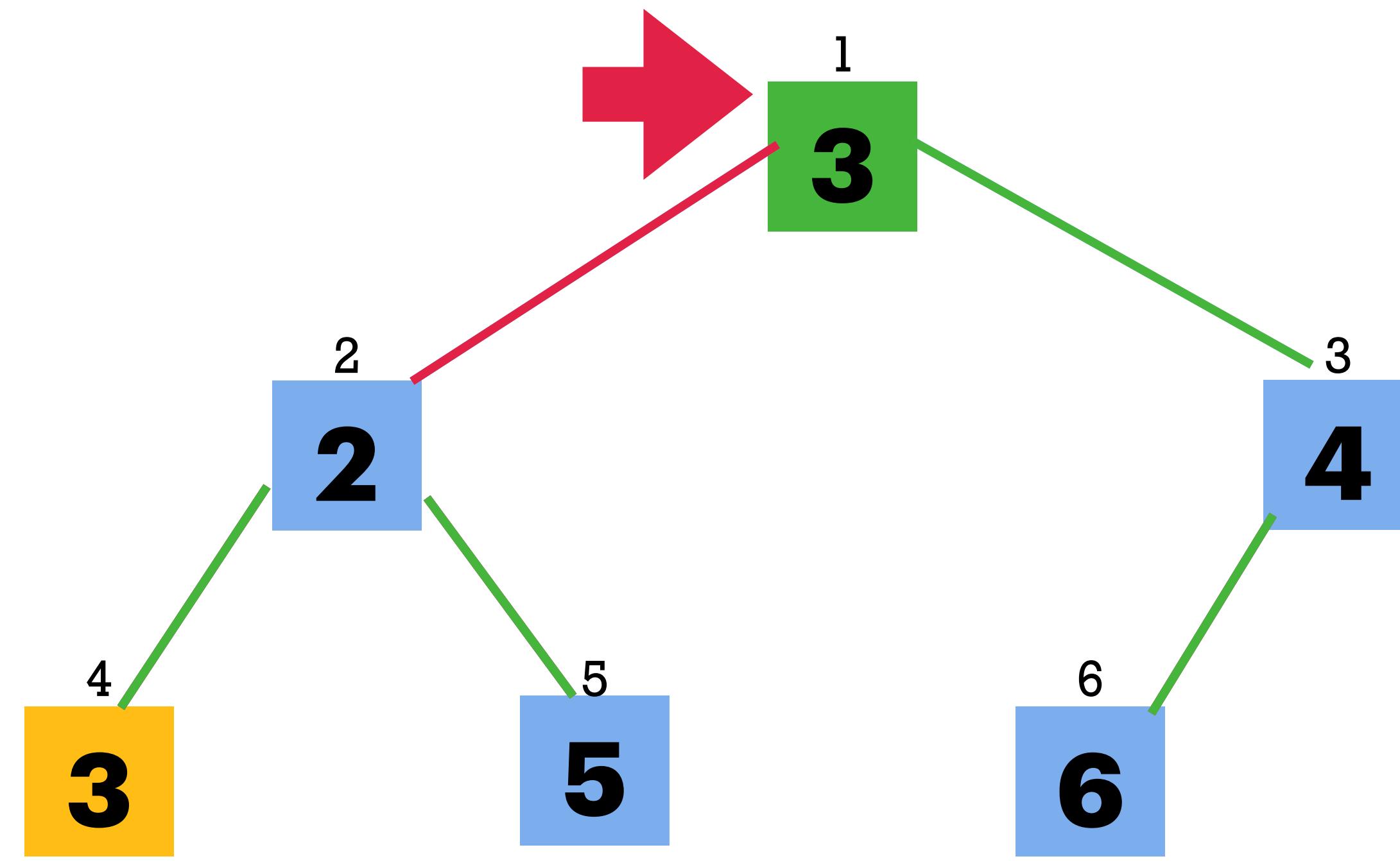
He a p S o r t



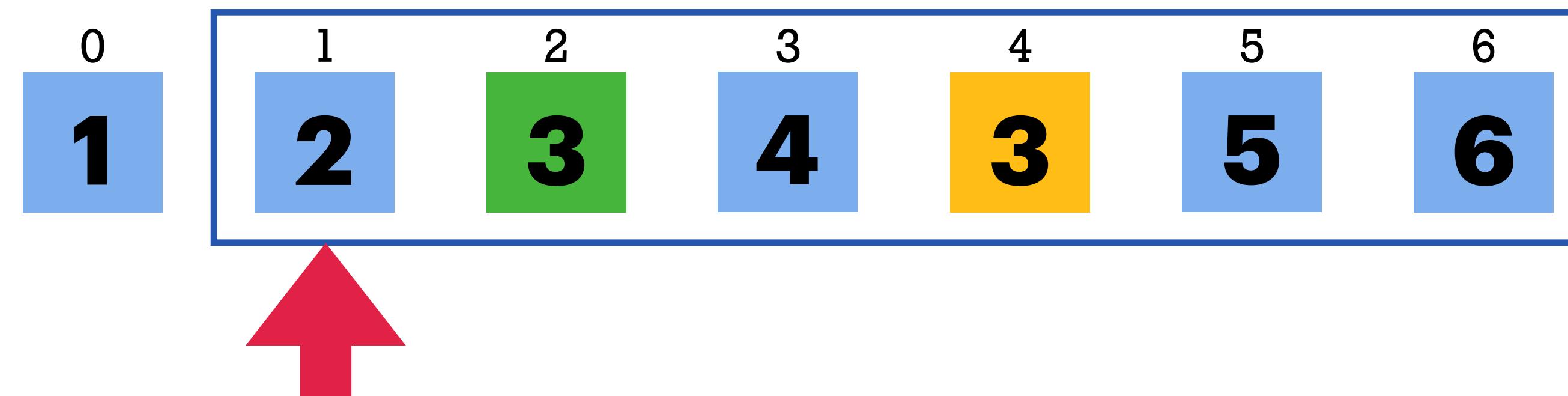
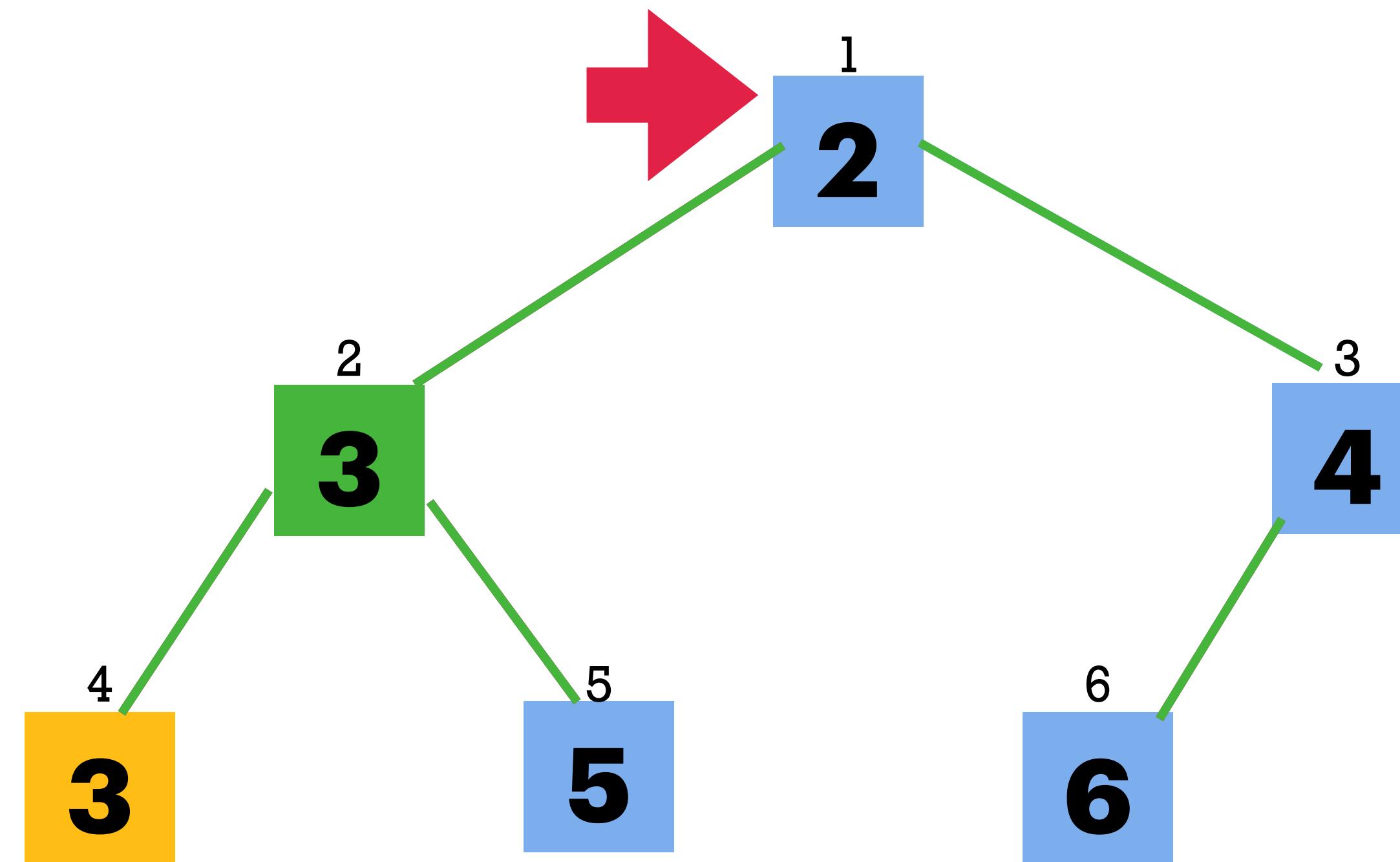
He a p S o r t



He a p S o r t



He a p S o r t



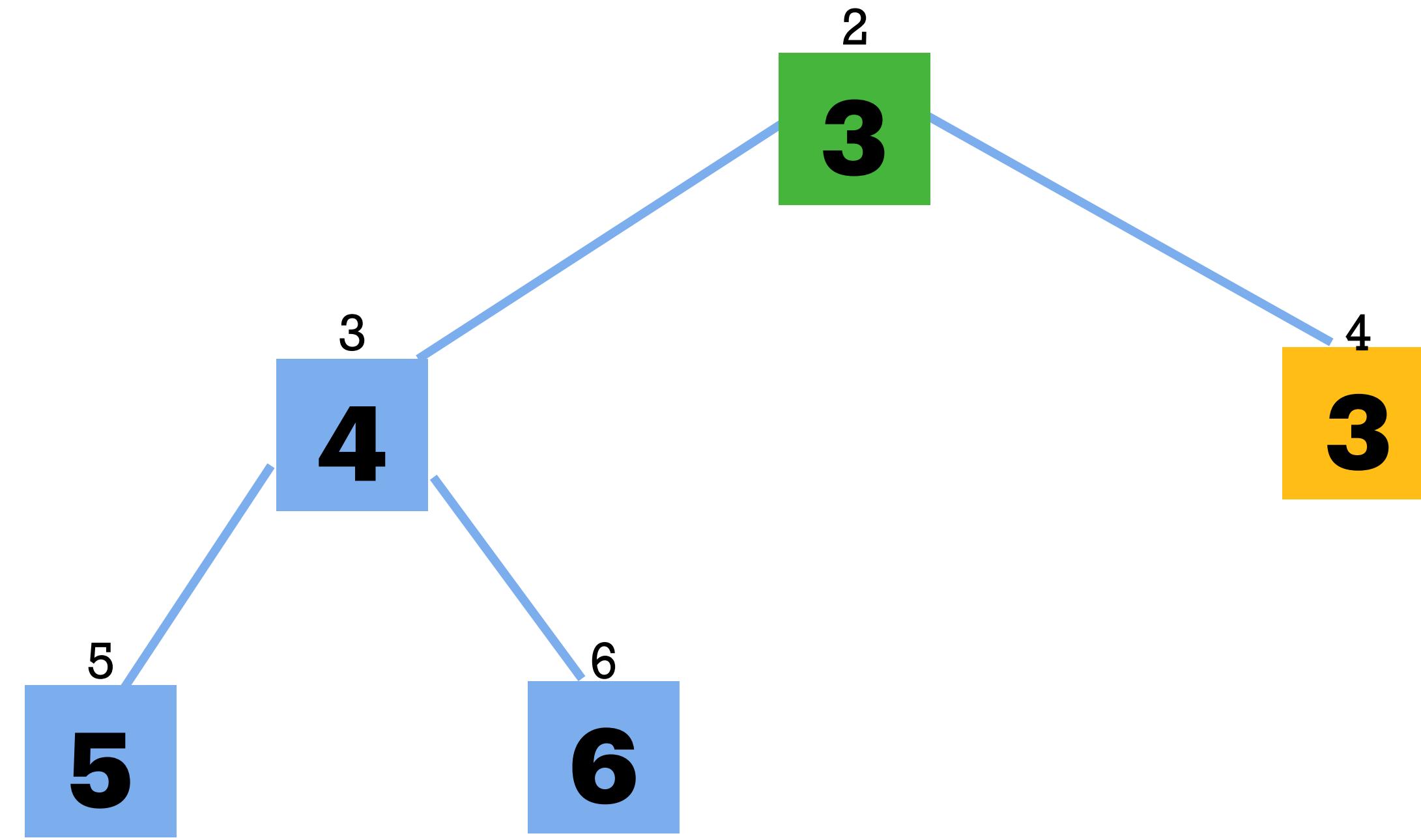
He a p S o r t



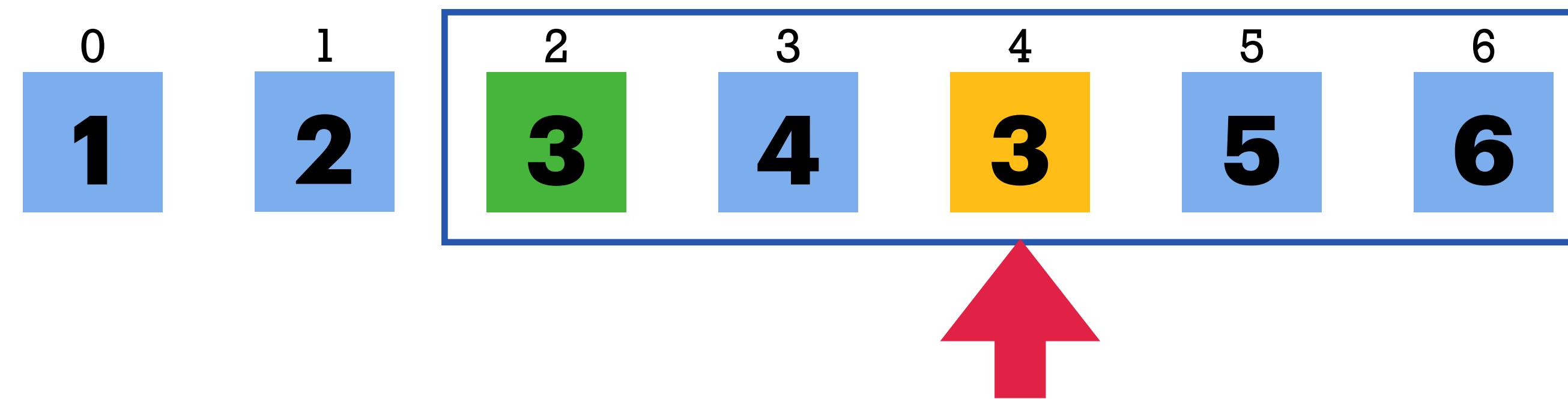
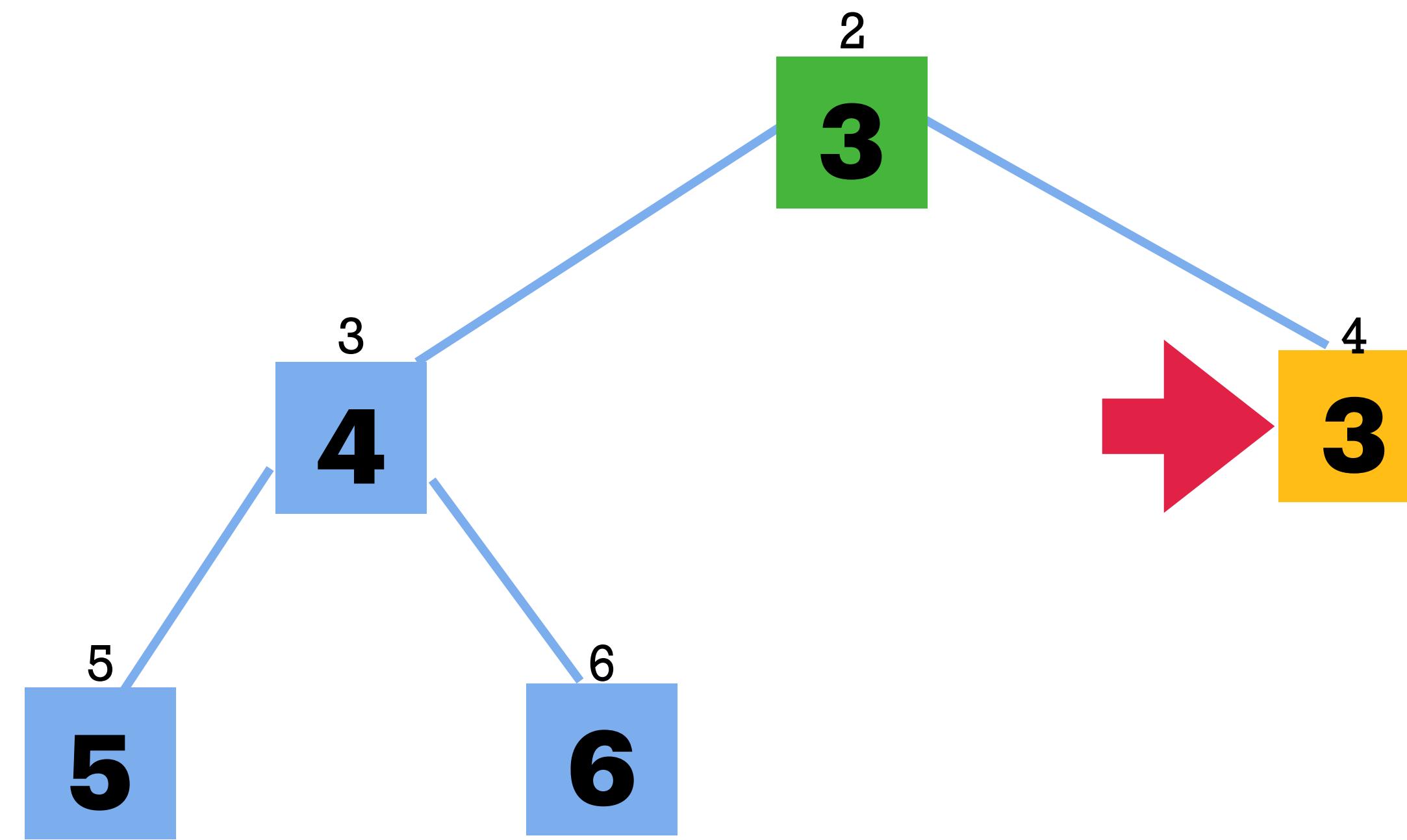
He a p S o r t



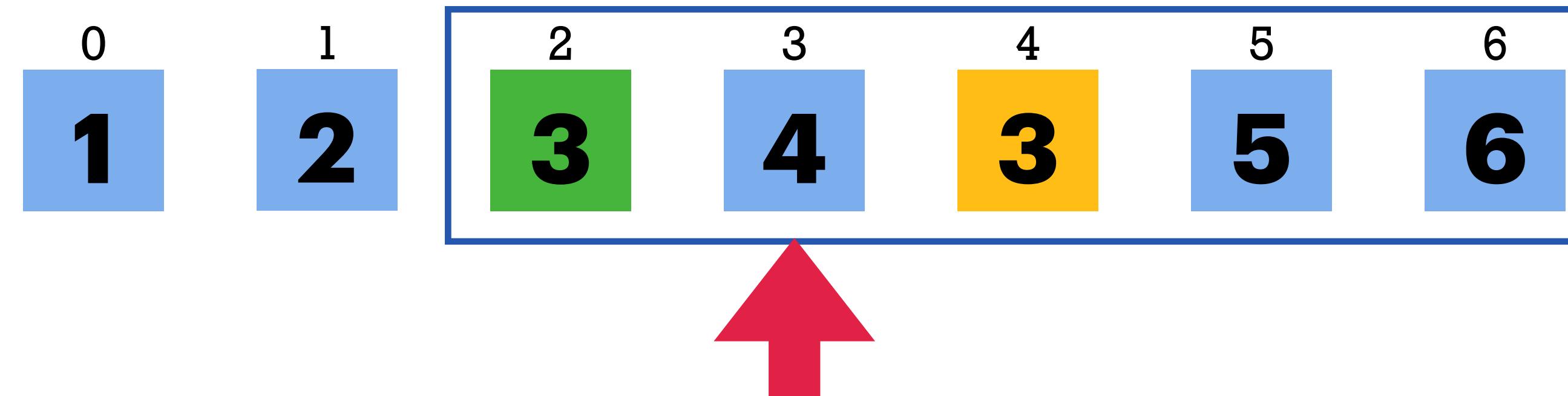
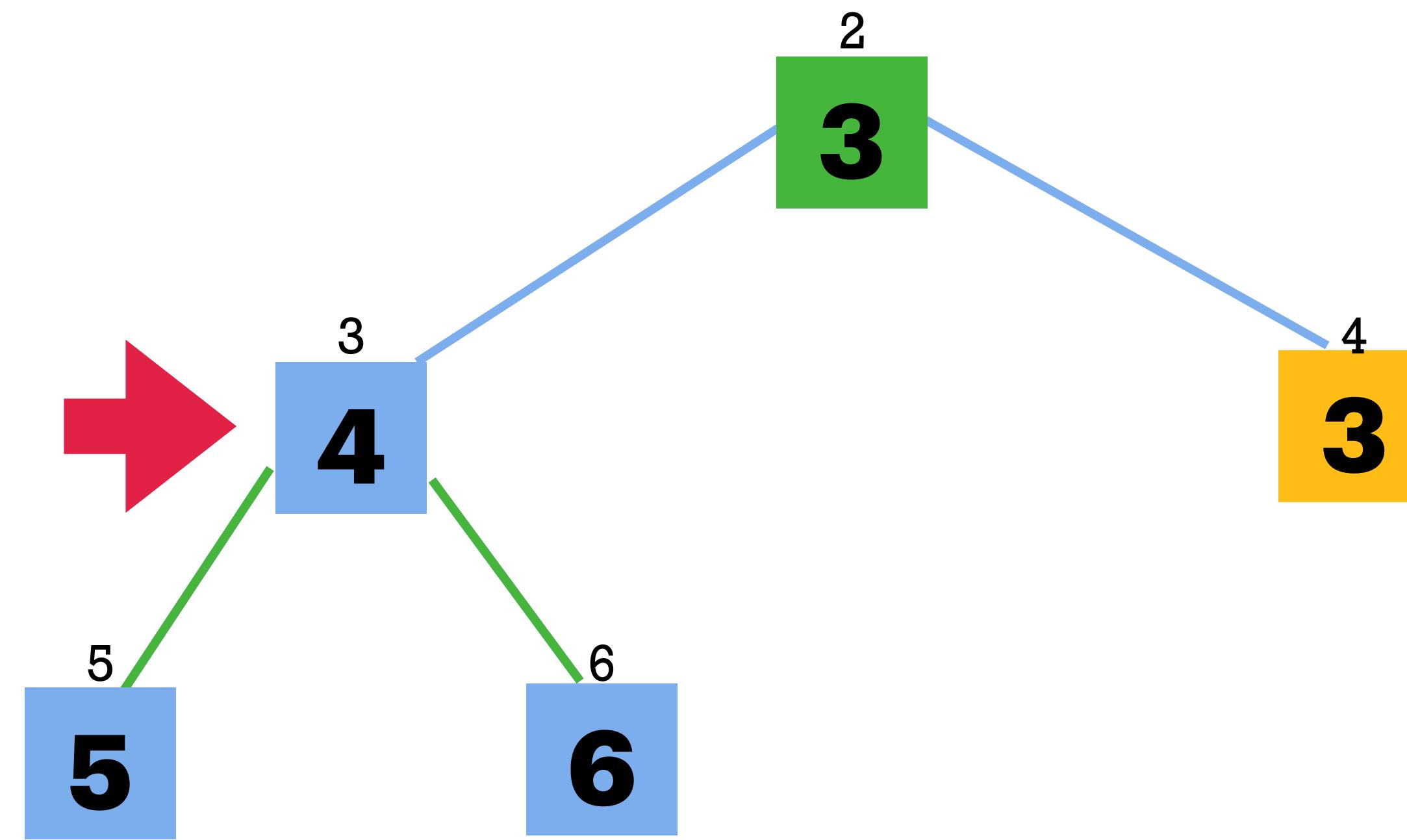
He a p S o r t



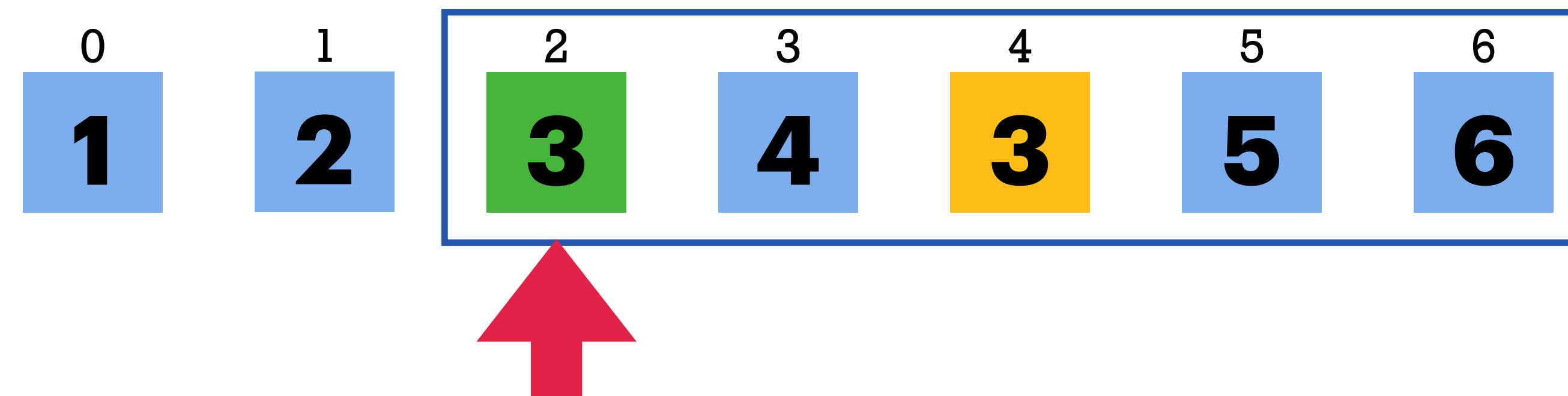
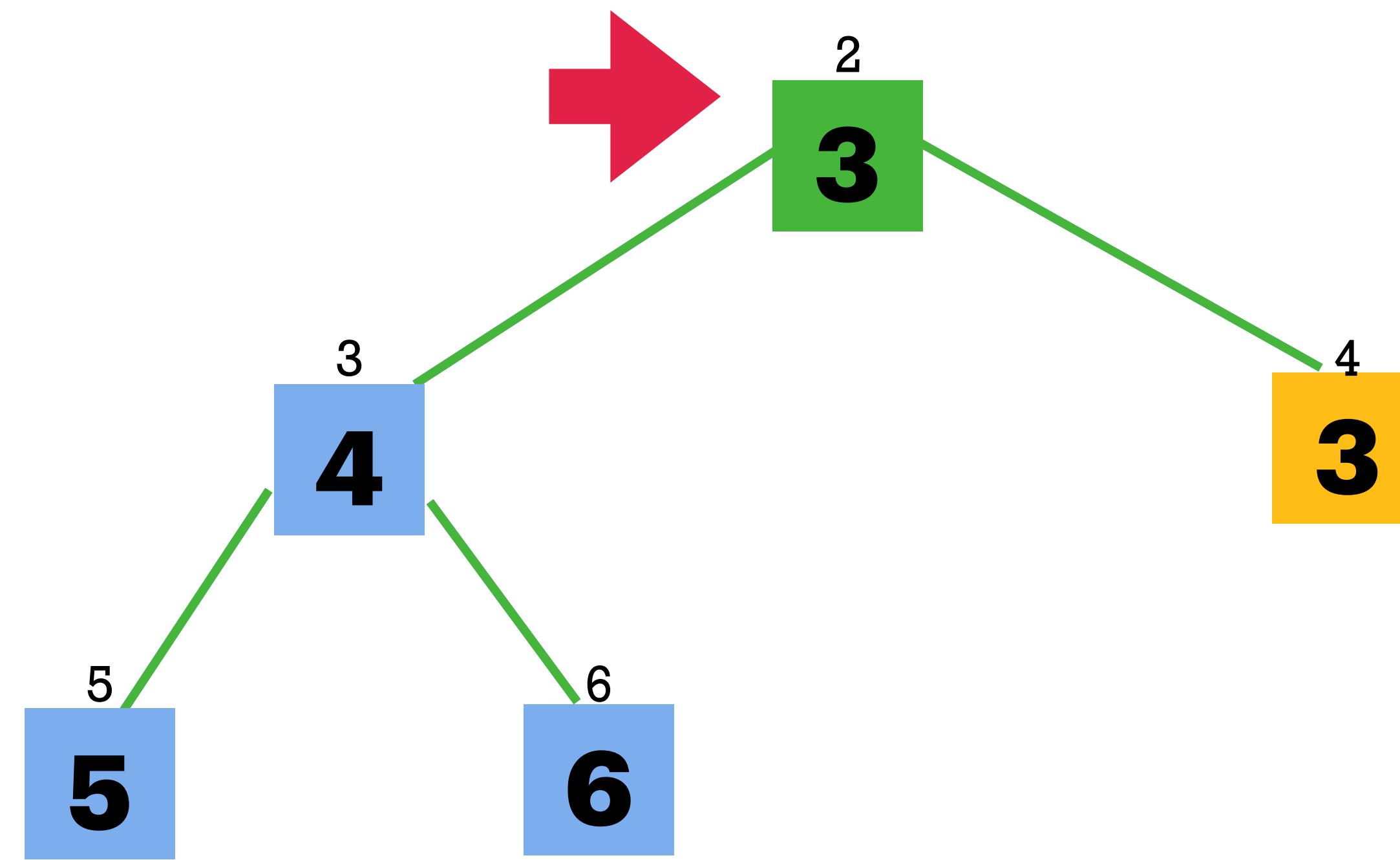
He a p S o r t



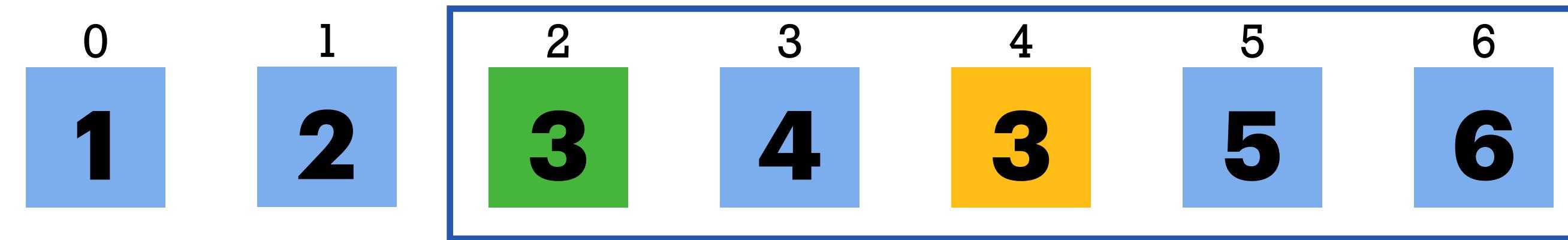
He a p S o r t



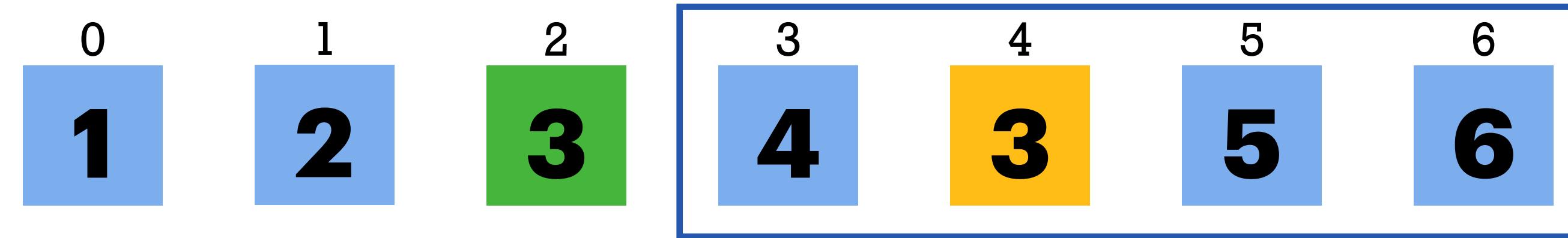
He a p S o r t



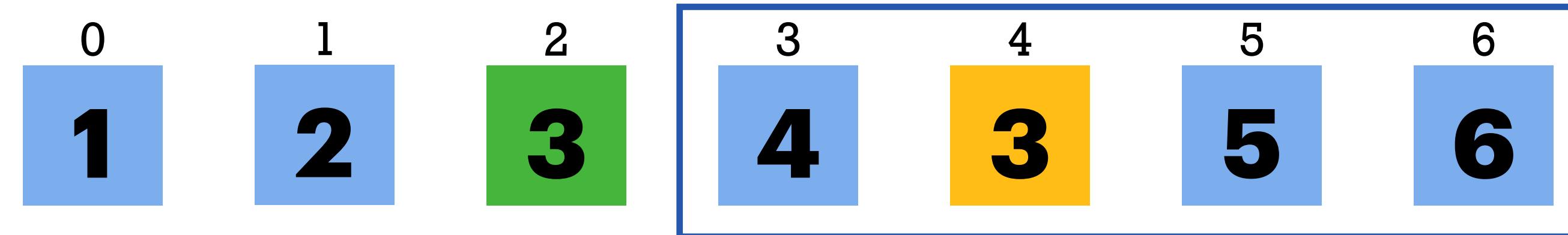
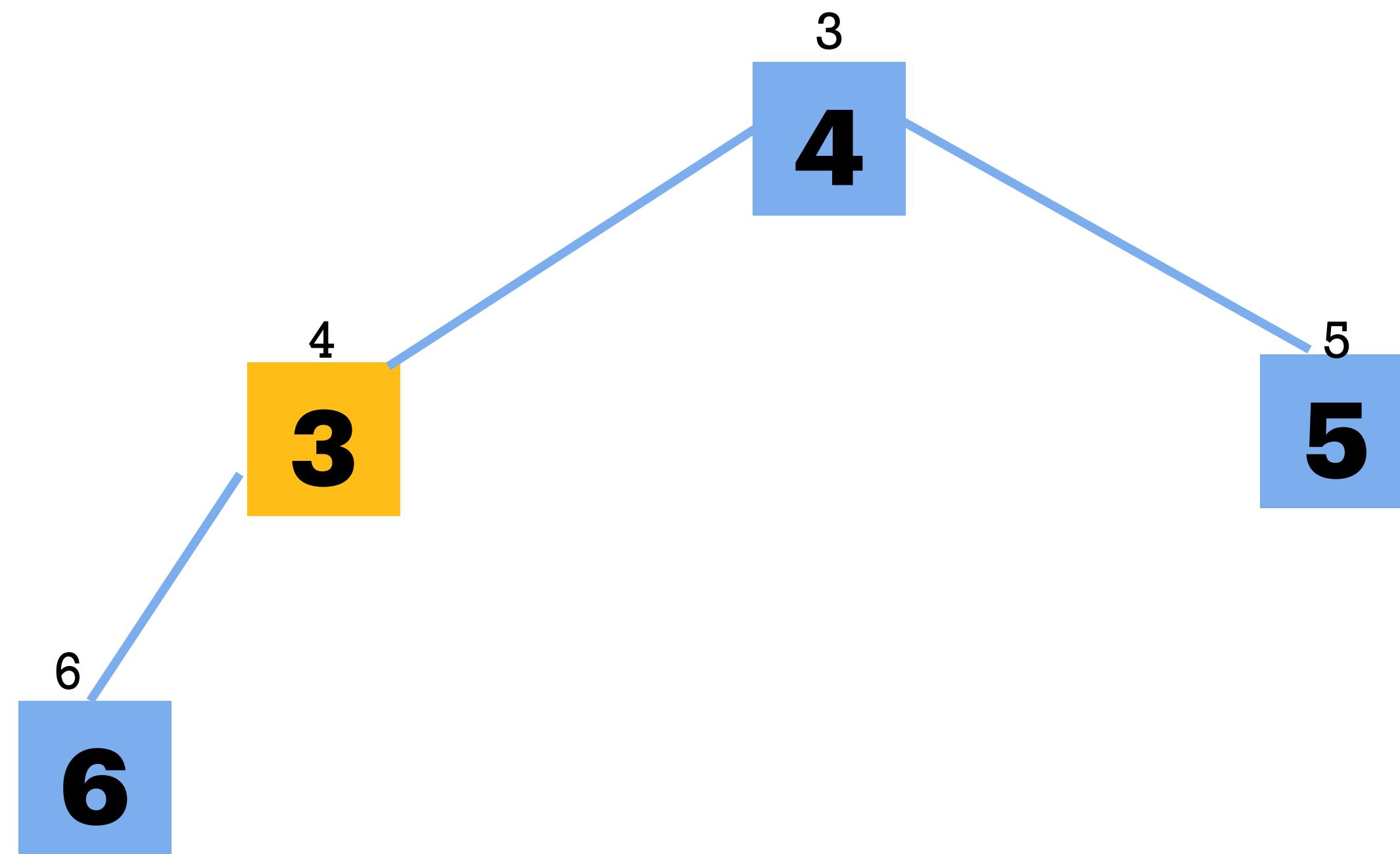
He a p S o r t



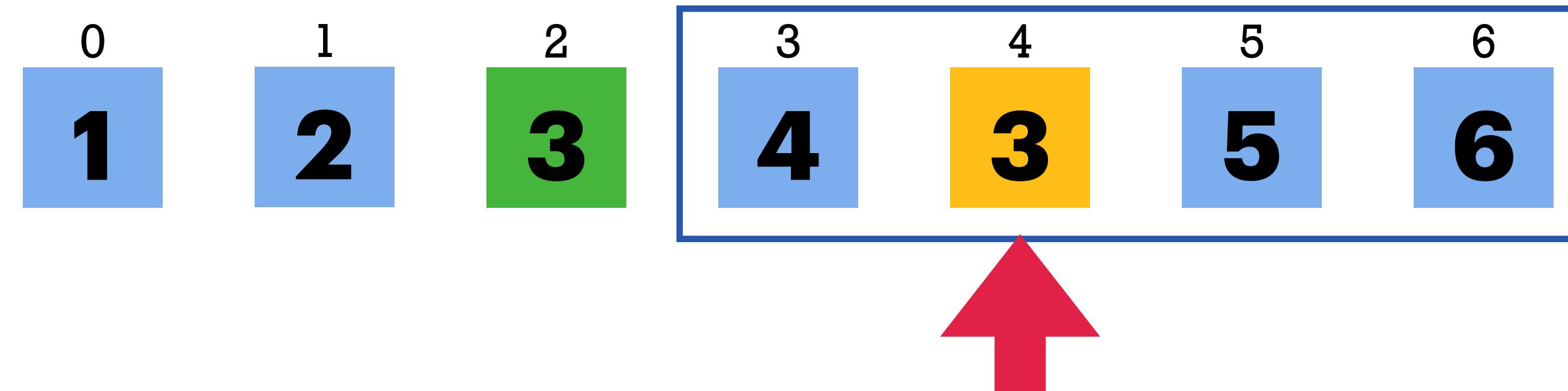
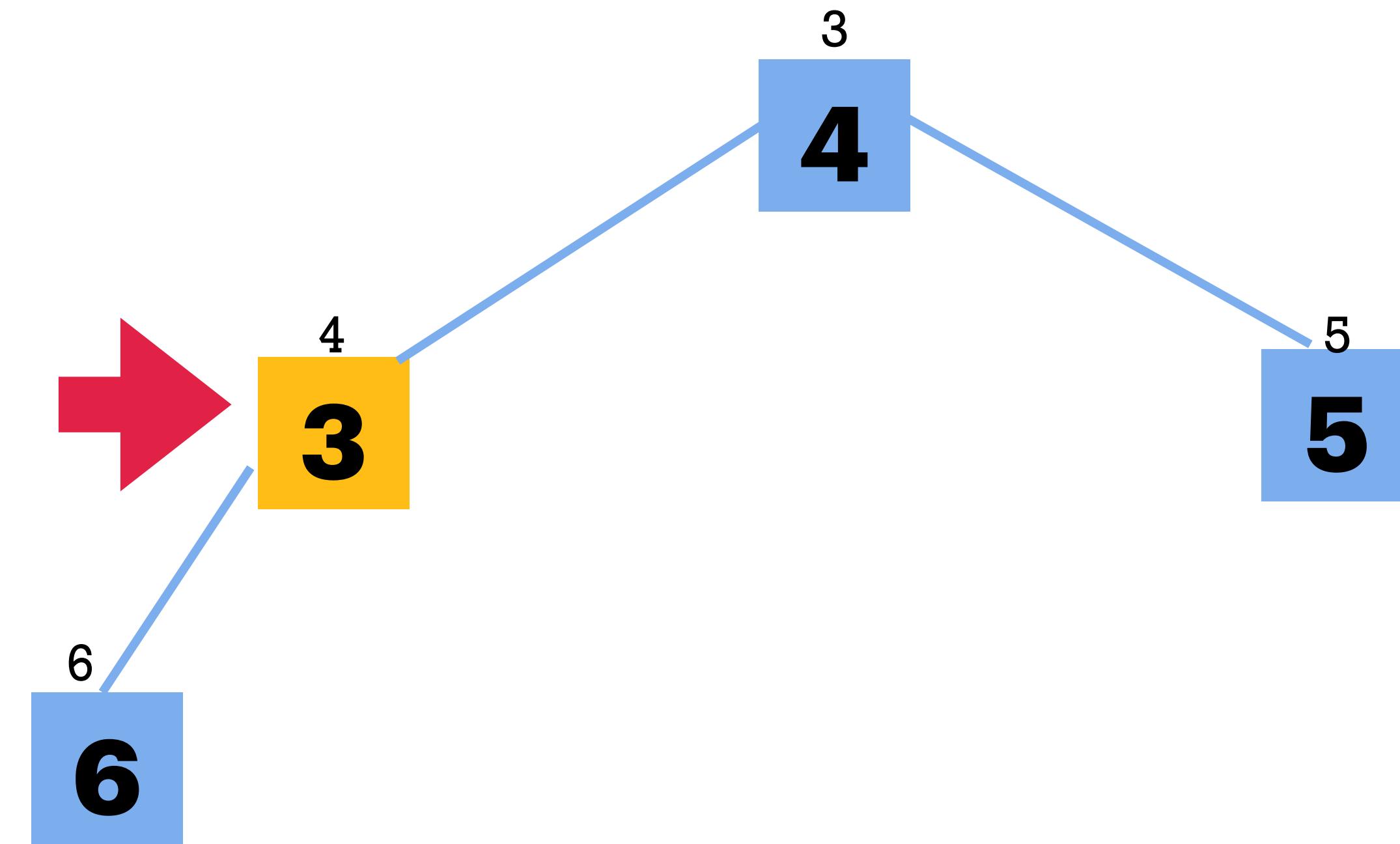
He a p S o r t



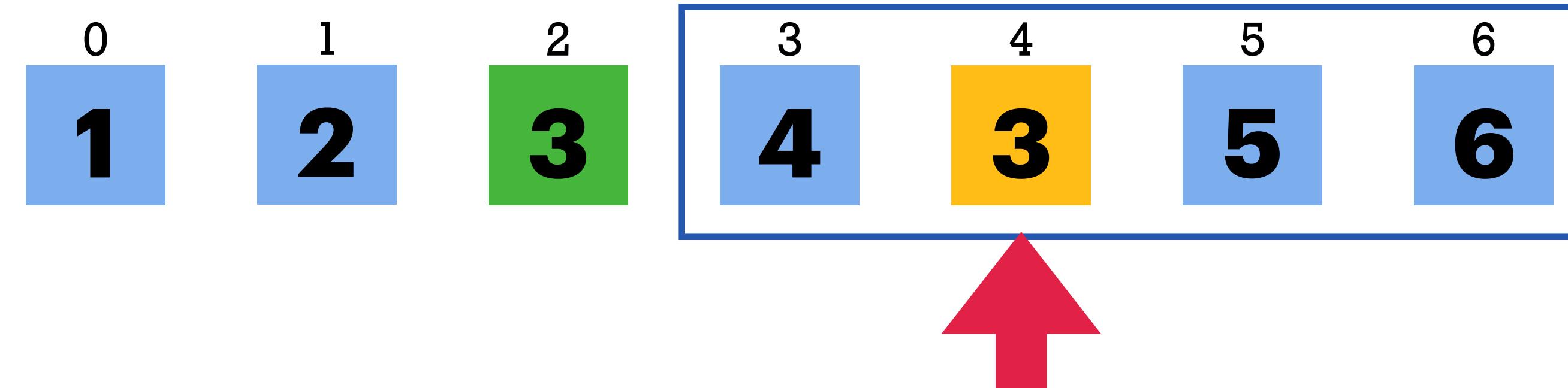
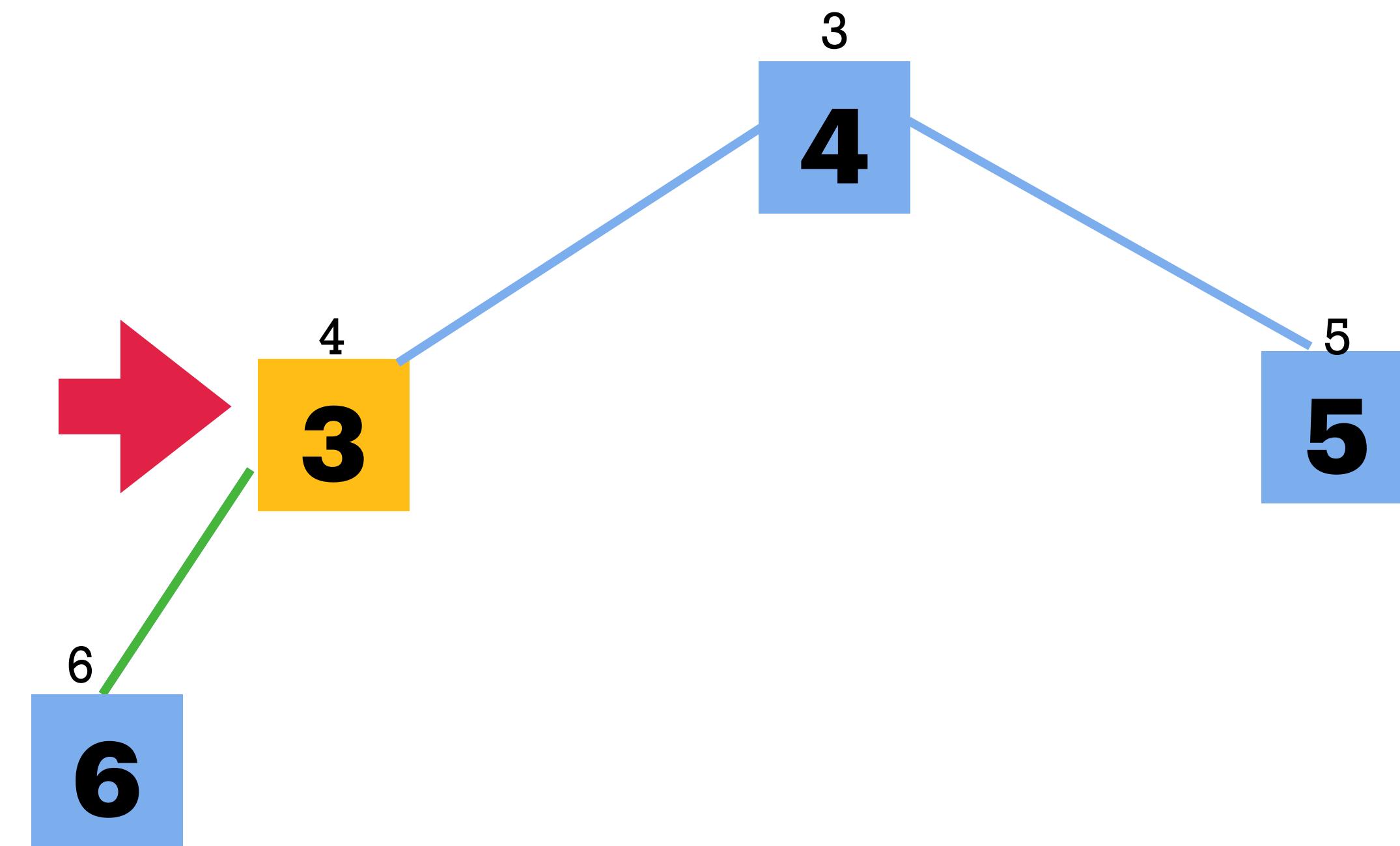
He a p S o r t



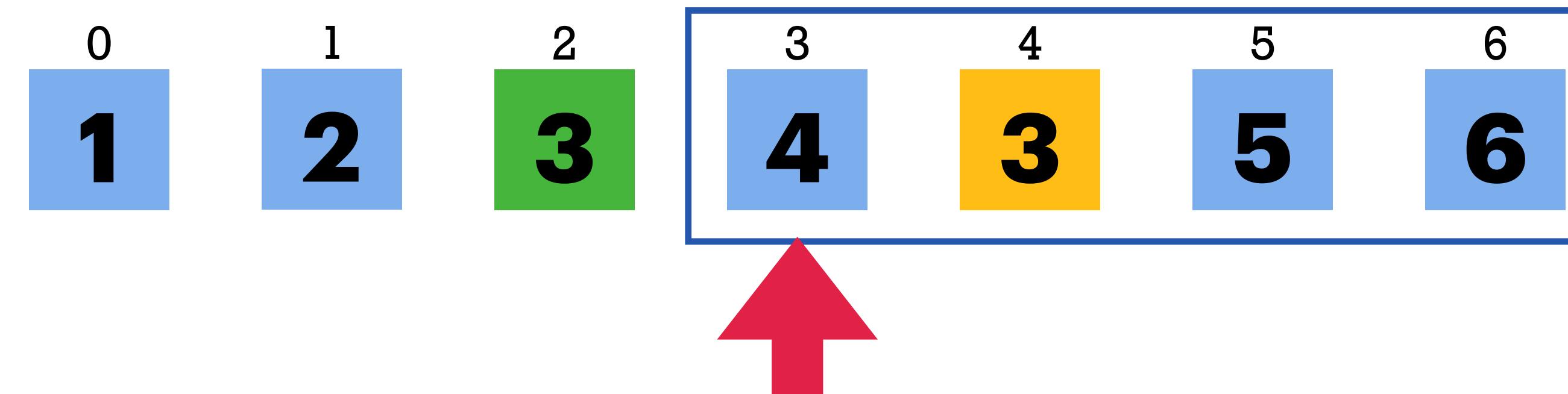
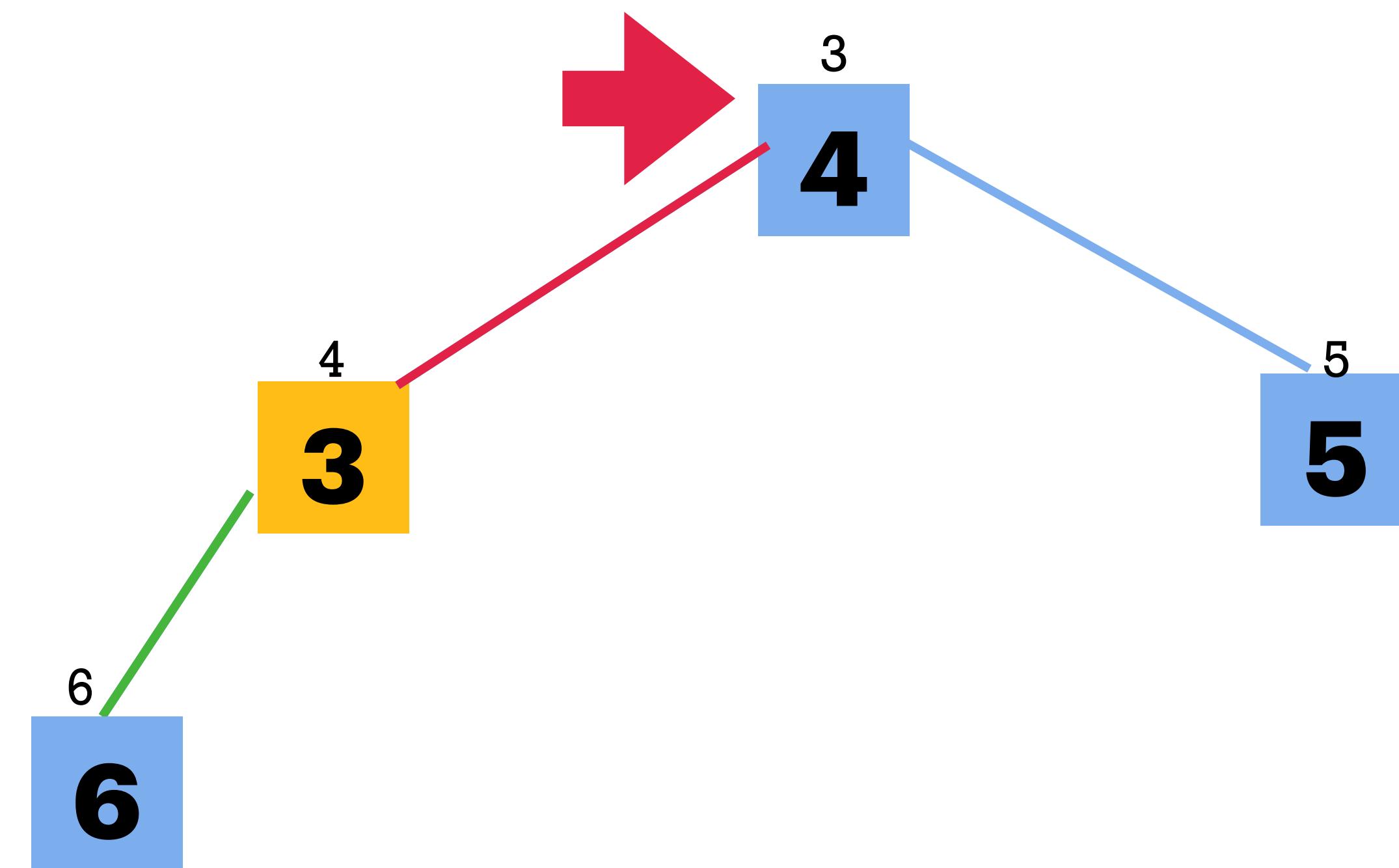
He a p S o r t



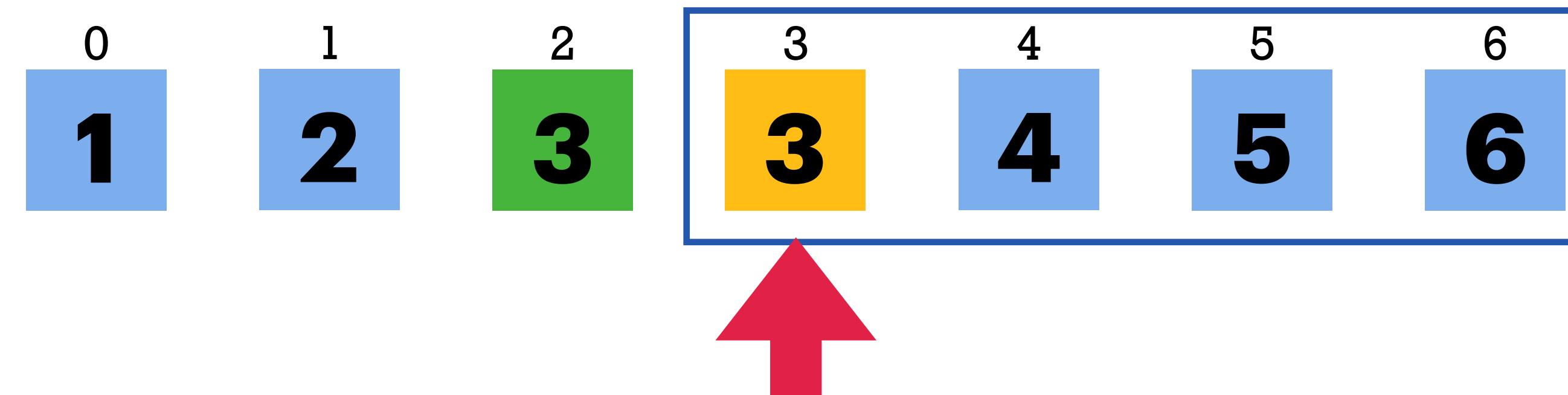
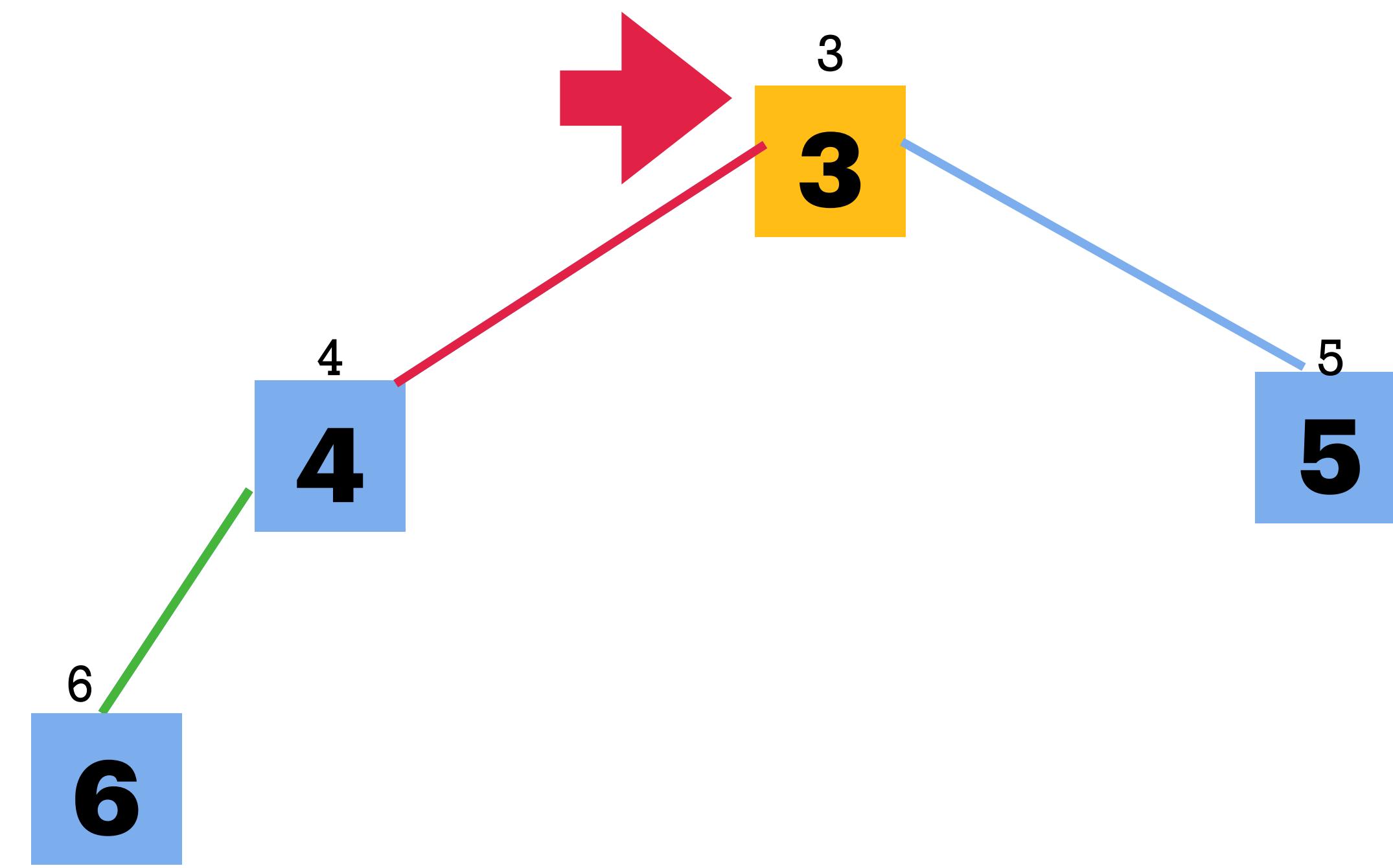
He a p S o r t



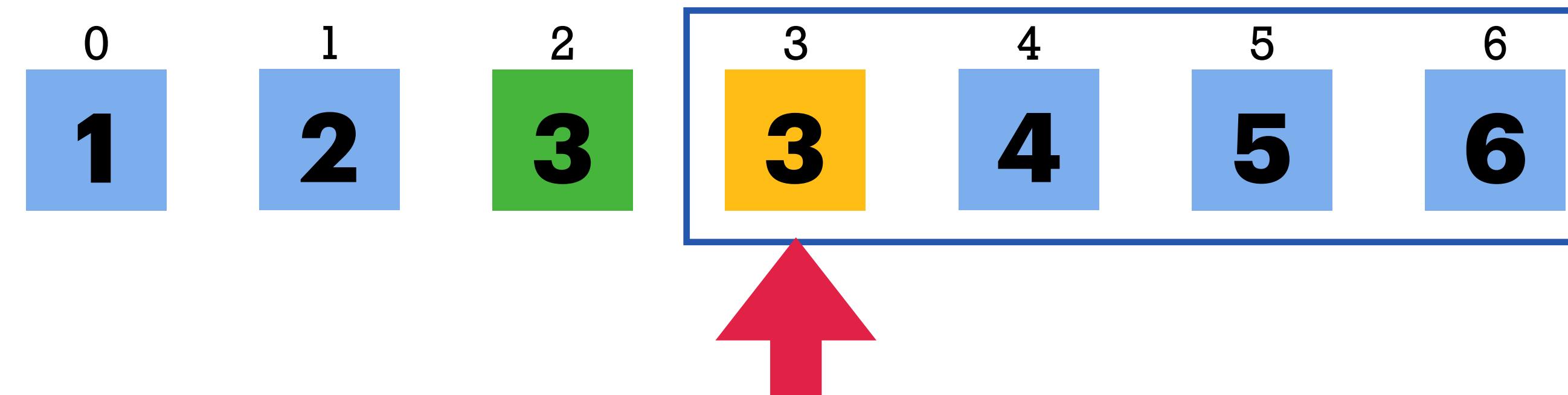
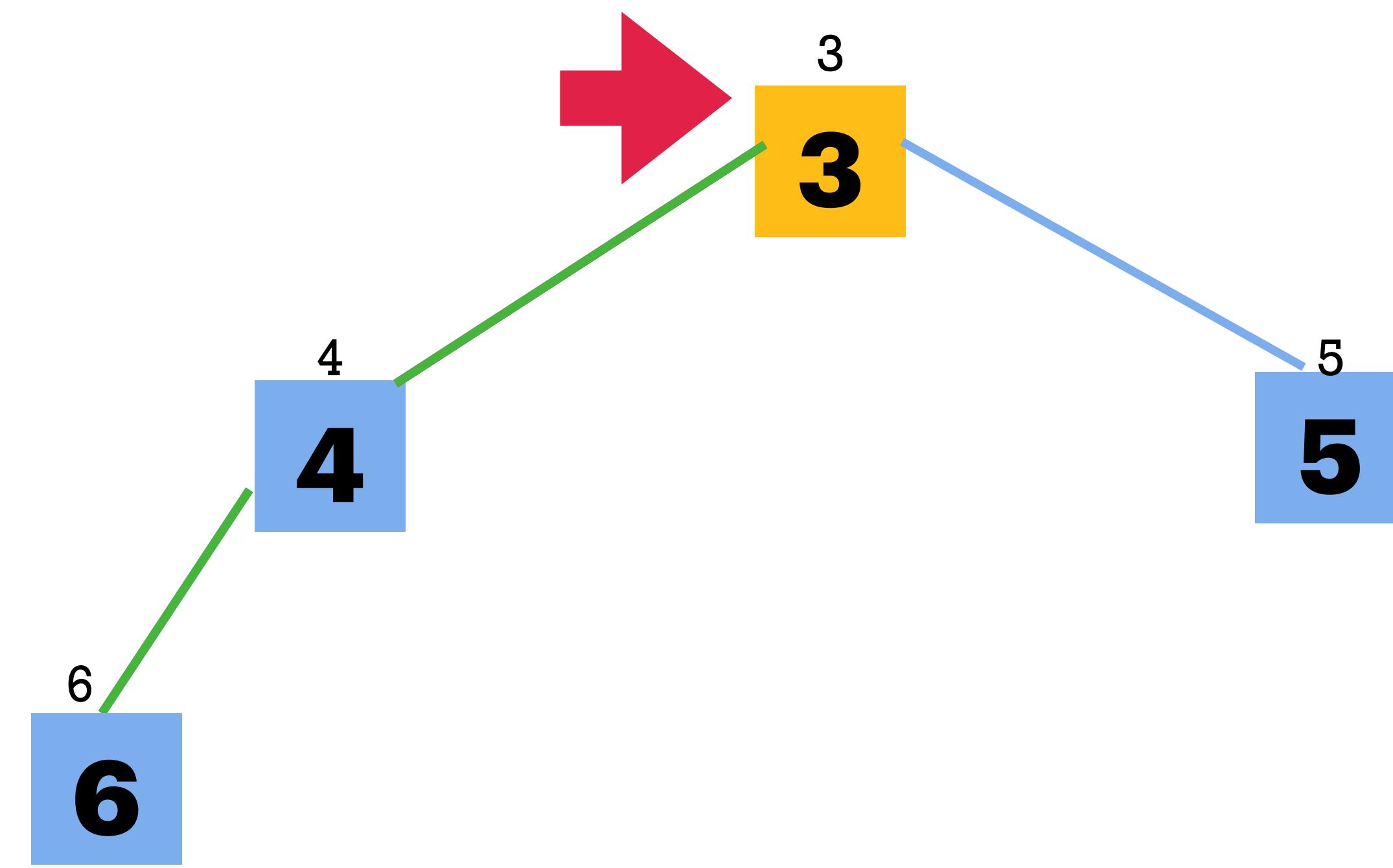
He a p S o r t



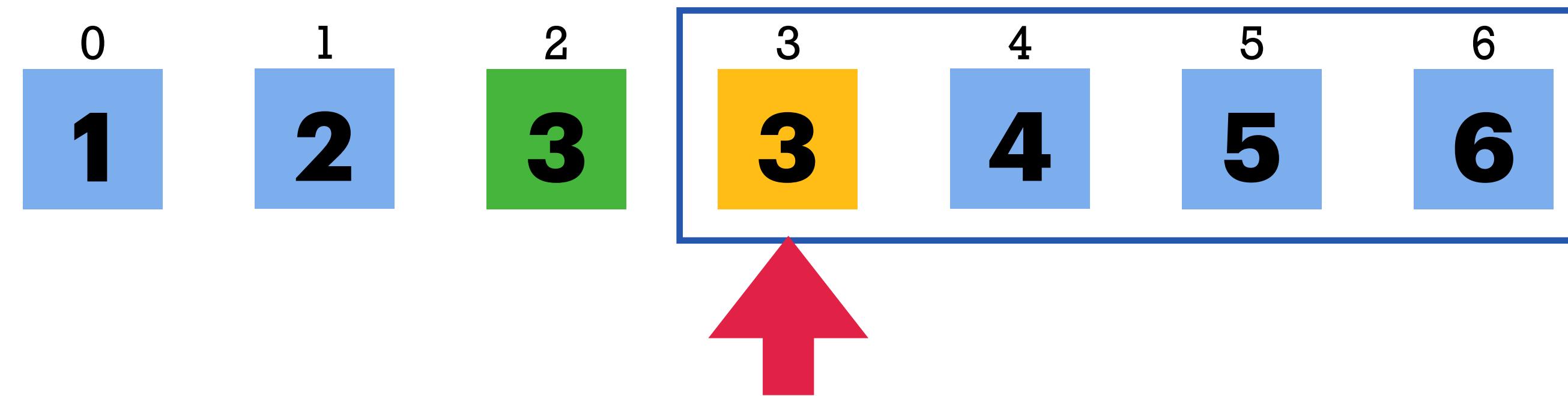
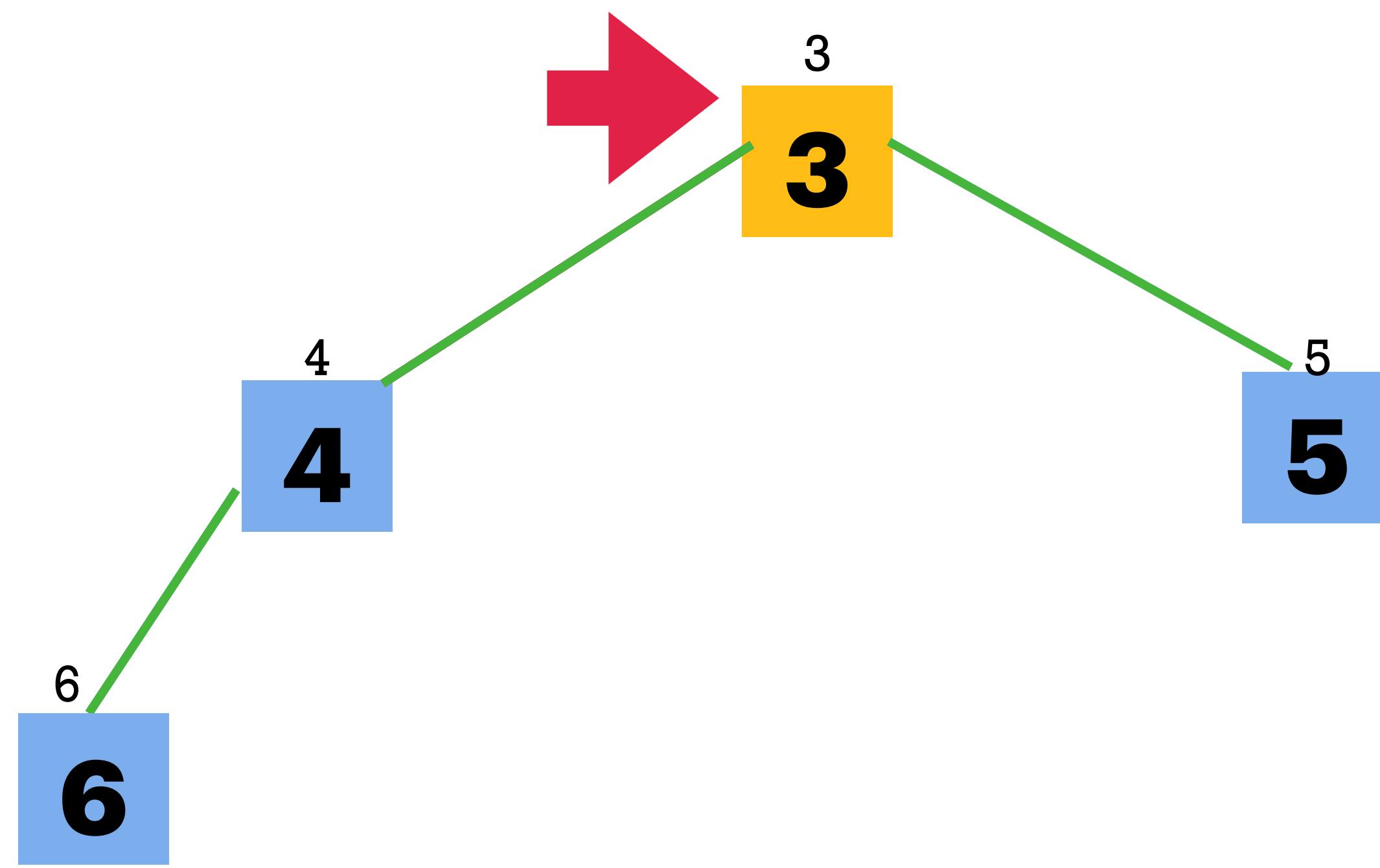
He a p S o r t



He a p S o r t



He a p S o r t



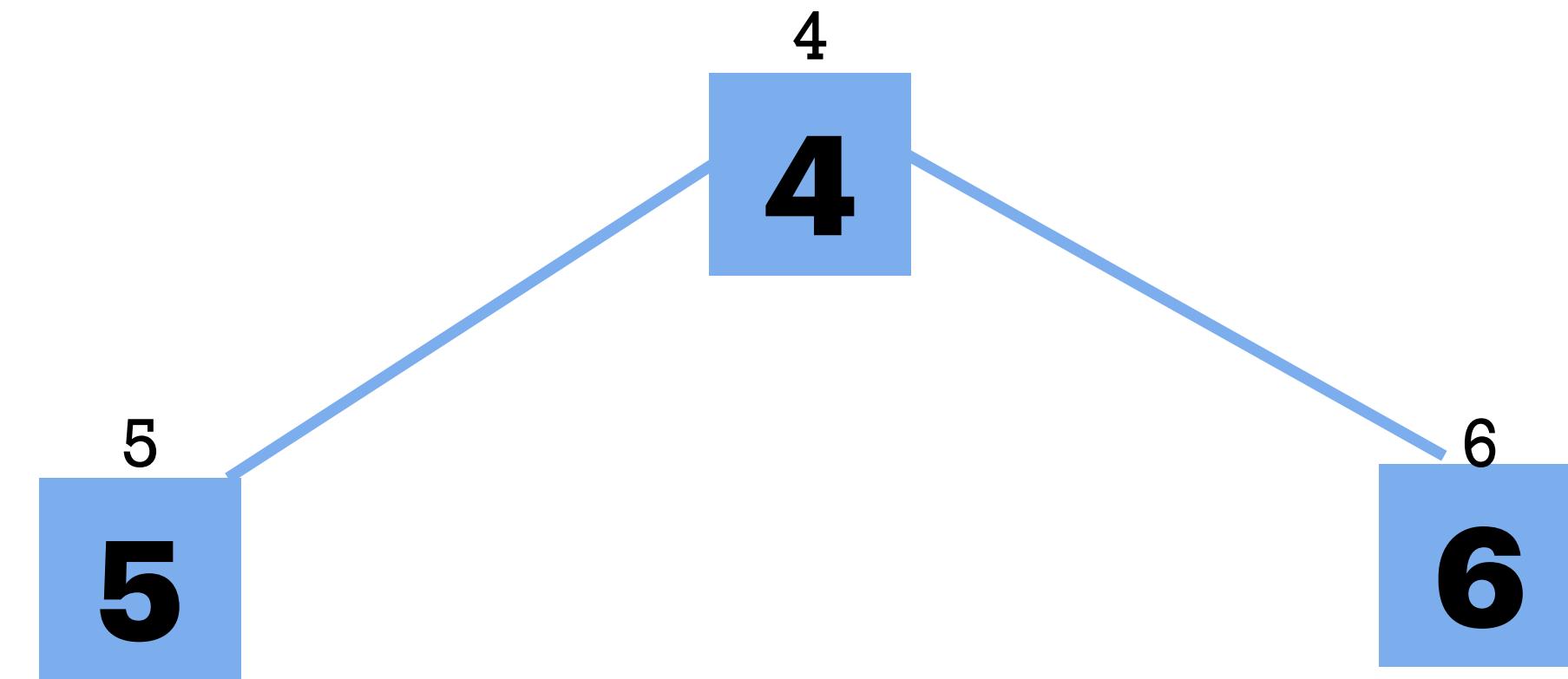
He a p S o r t



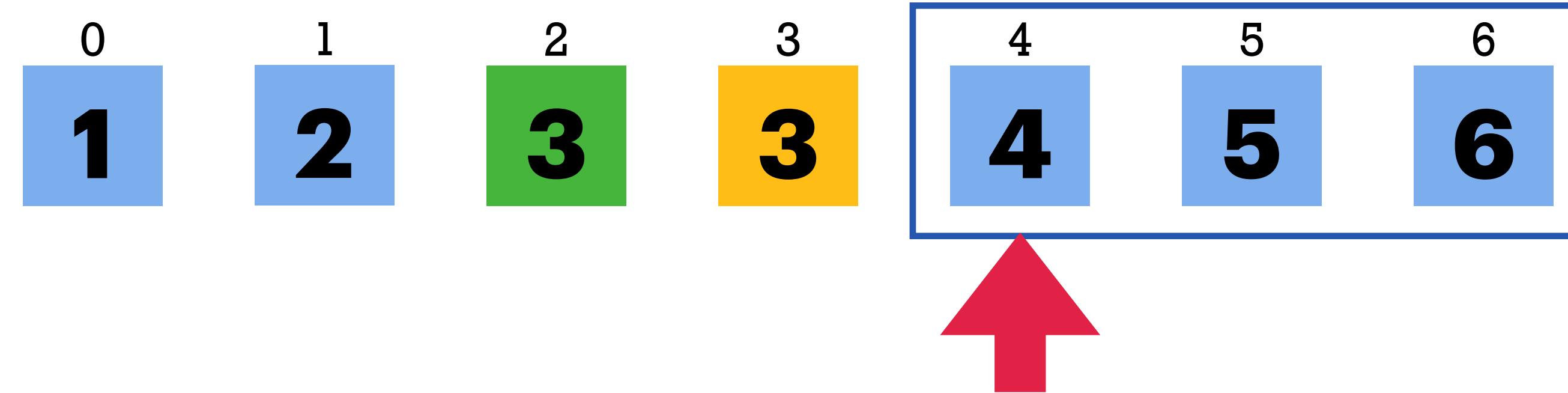
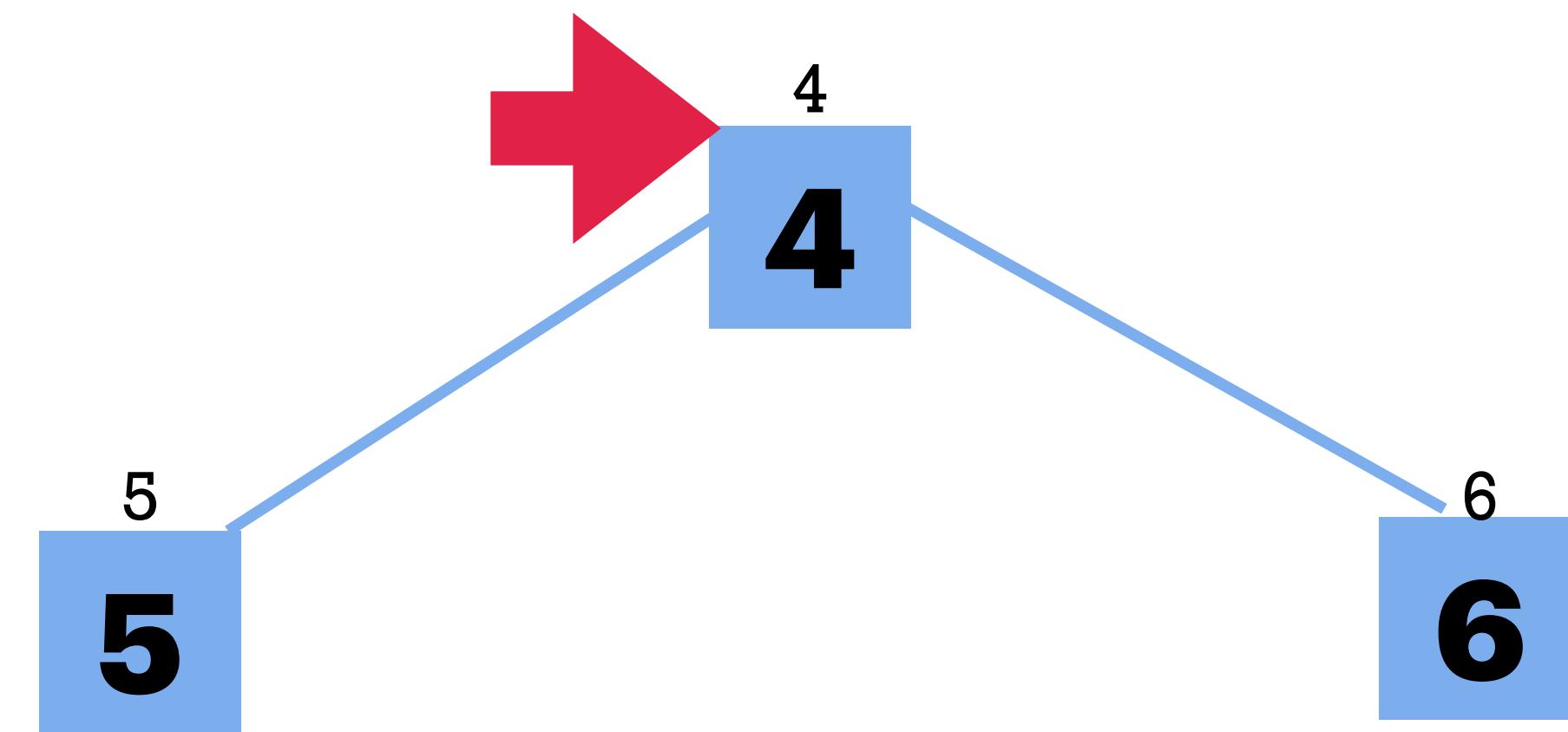
He a p S o r t



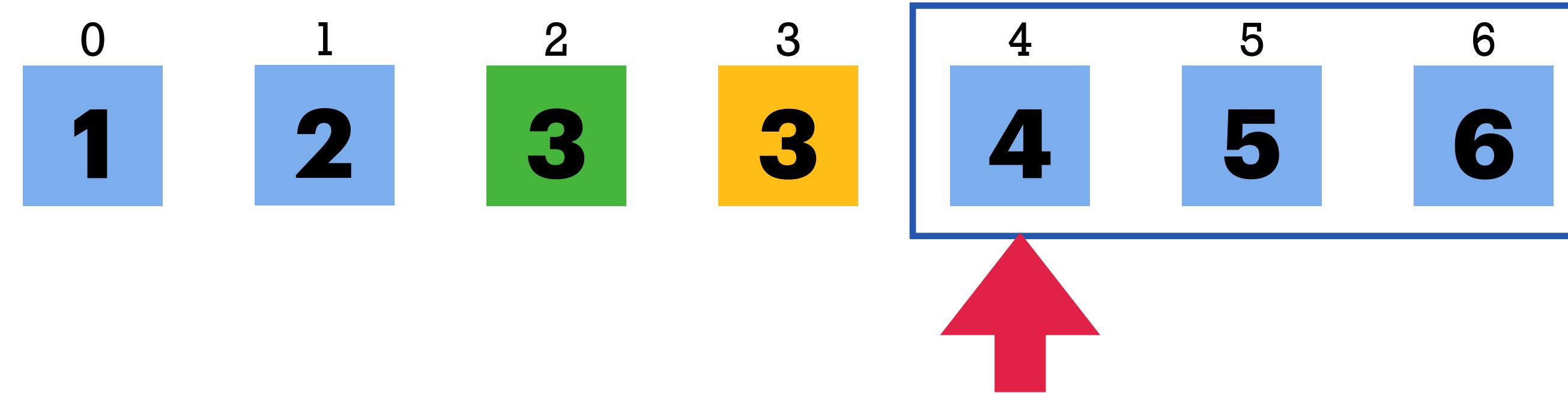
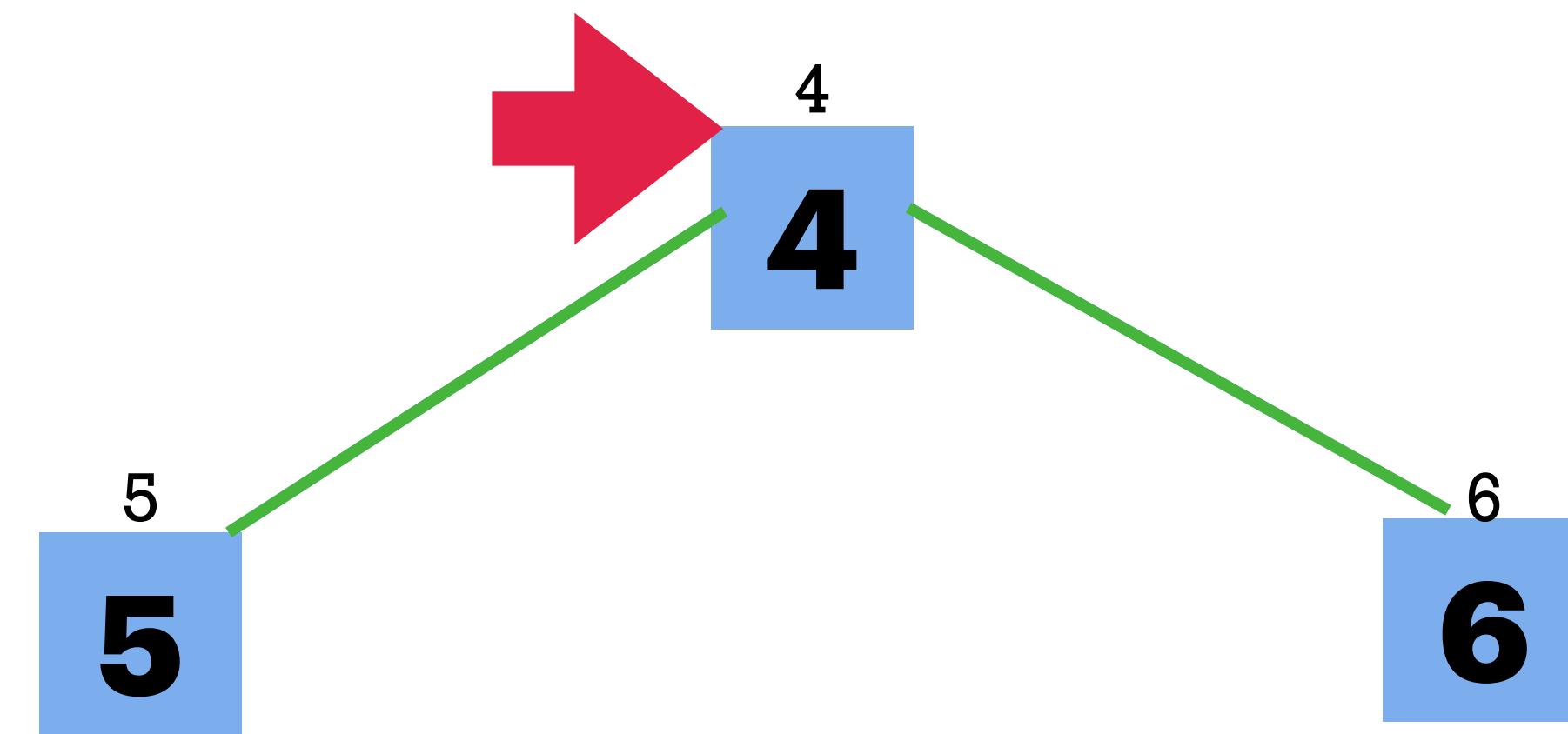
He a p S o r t



He a p S o r t



He a p S o r t



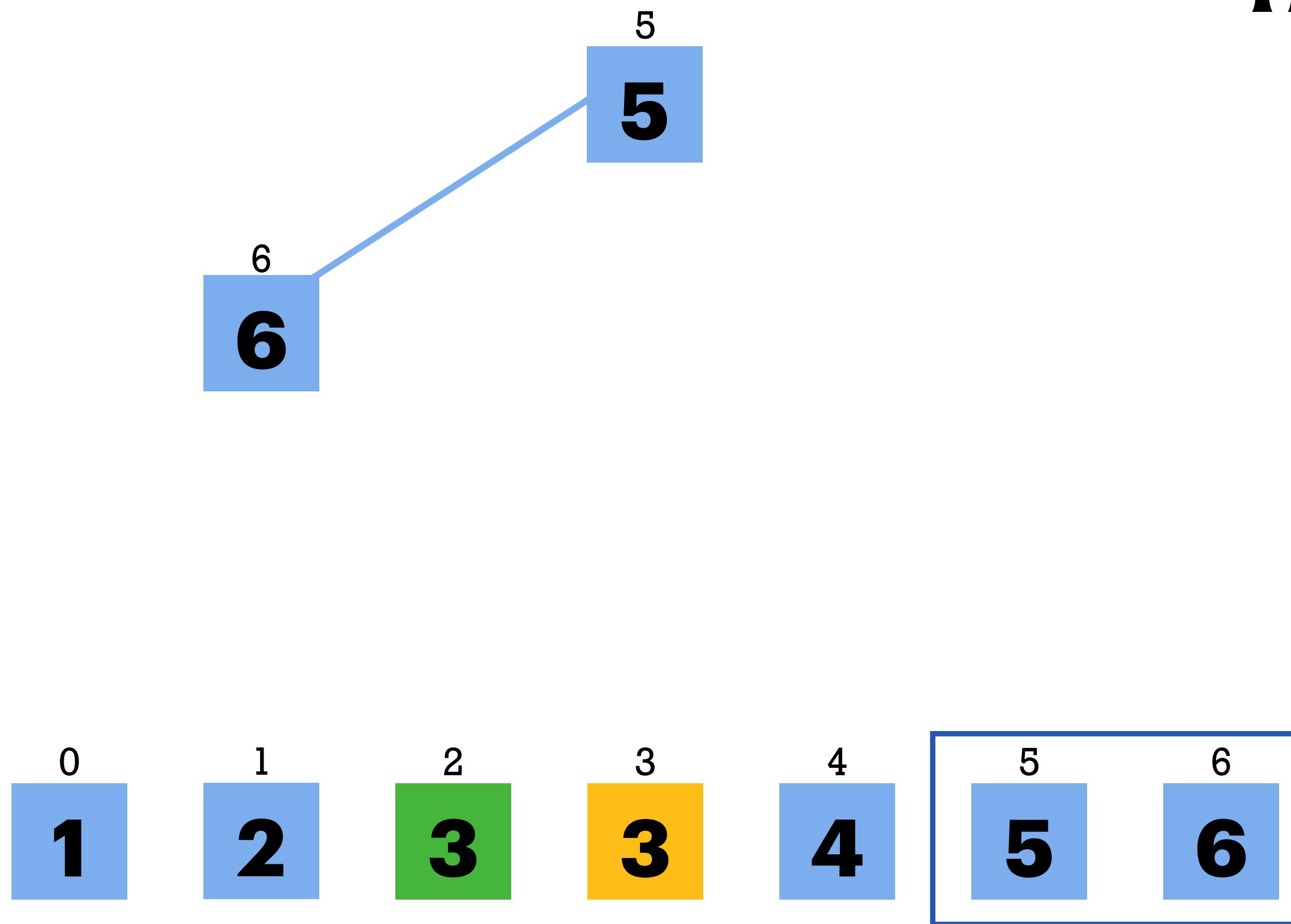
He a p S o r t



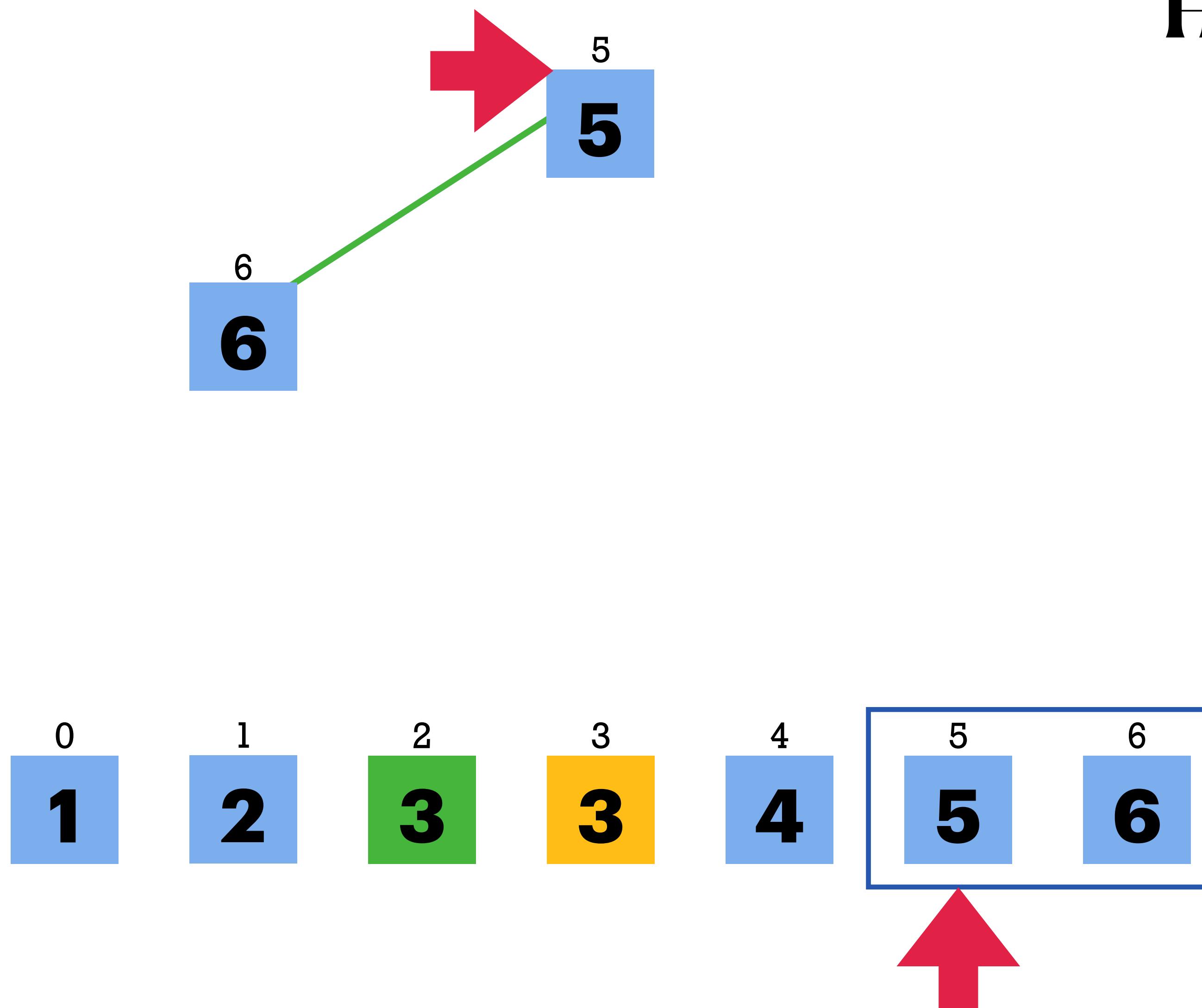
He a p S o r t



He a p S o r t



He a p S o r t



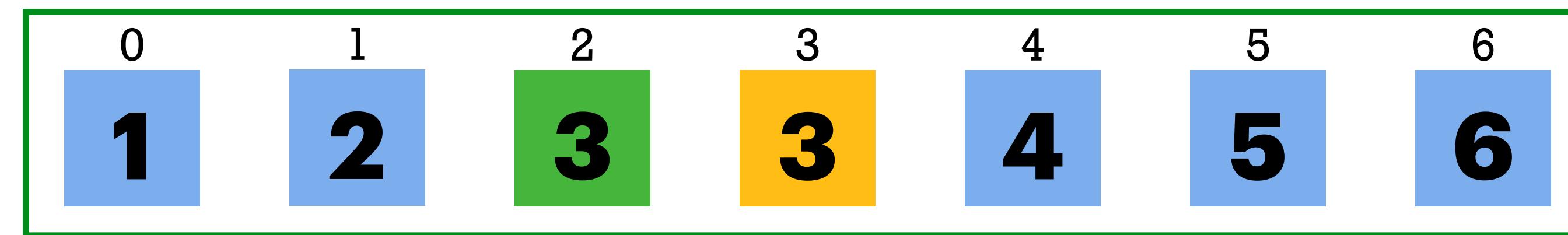
He a p S o r t



He a p S o r t



He a p S o r t



Problema Concordanței

Problema Concordanței = Determinarea frecvenței de apariție a cuvintelor într-un text

Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele

Pentru fiecare cuvânt se verifică dacă este sau nu la prima apariție

Dacă e la prima apariție, cuvântul se inserează în ABP și se initializează contorul cu 1

Dacă nu e la prima apariție se incrementează contorul cu 1

Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele

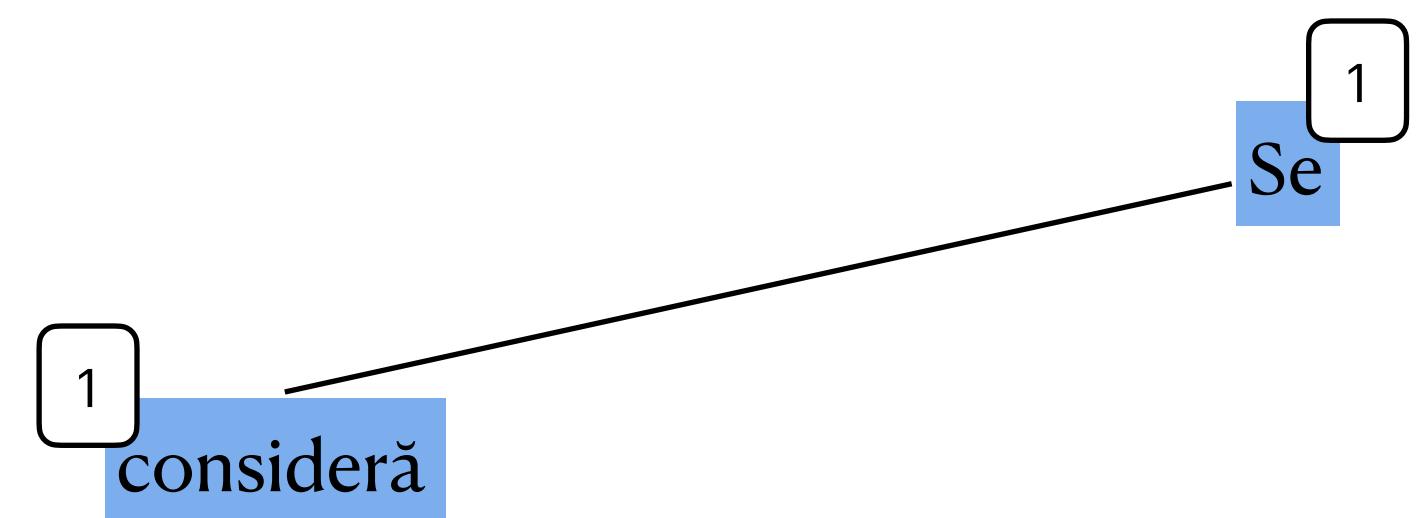
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele

Se 1

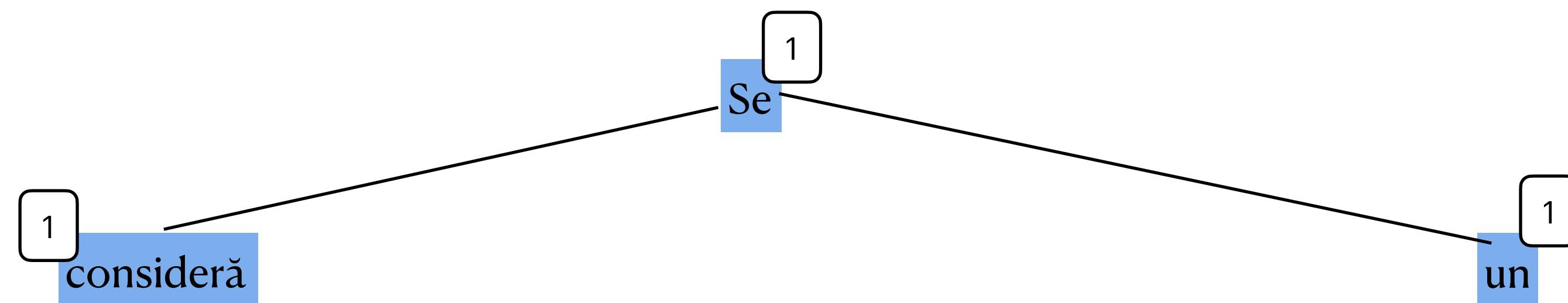
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



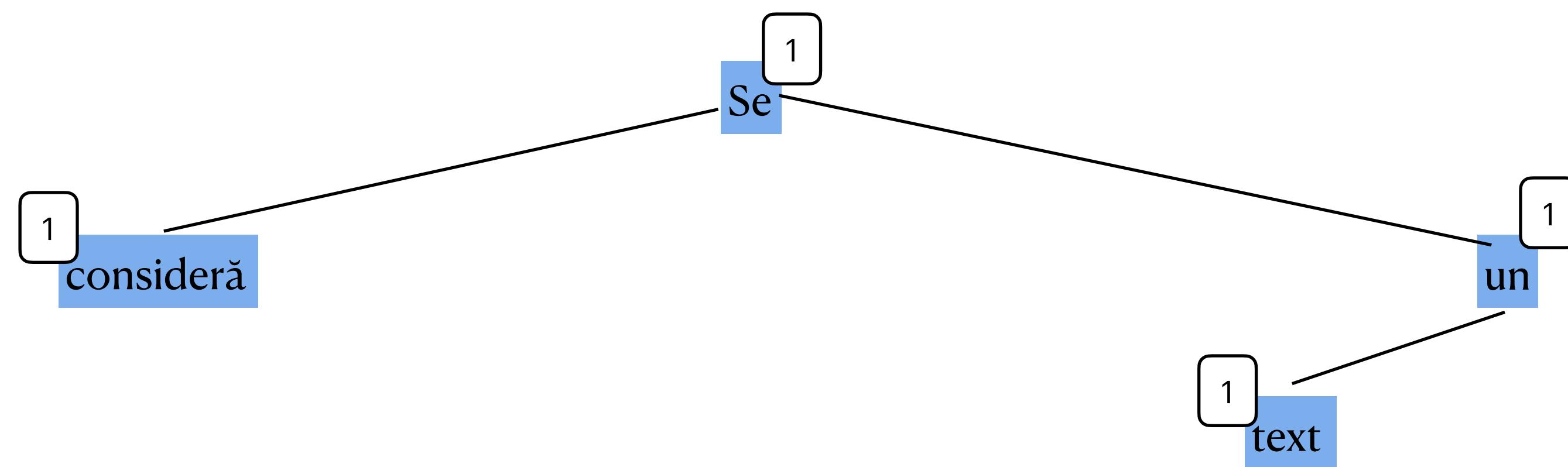
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



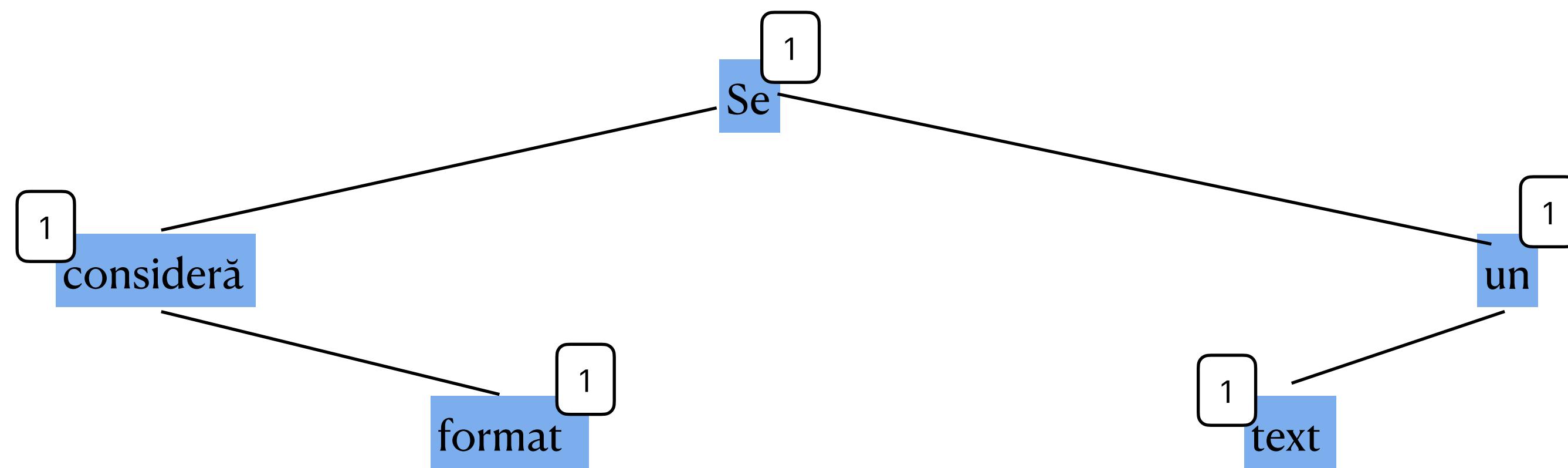
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



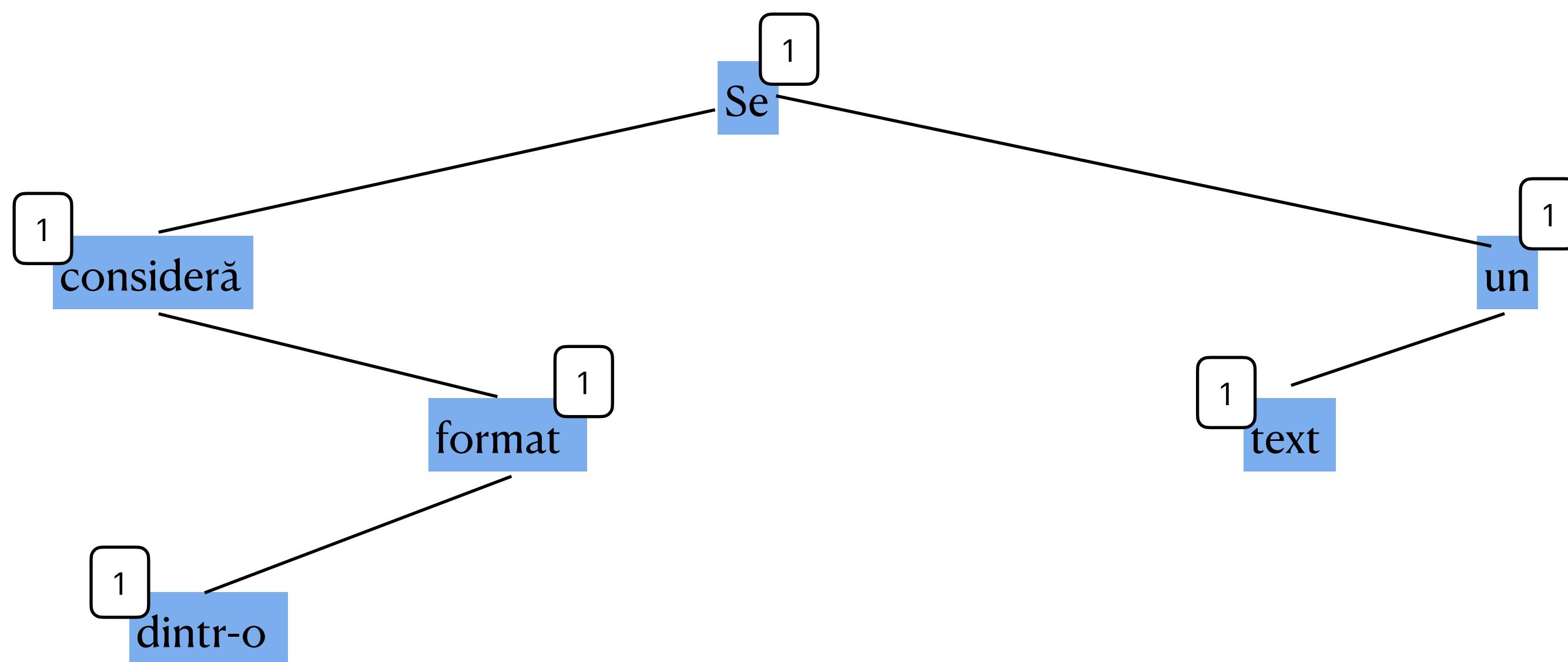
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



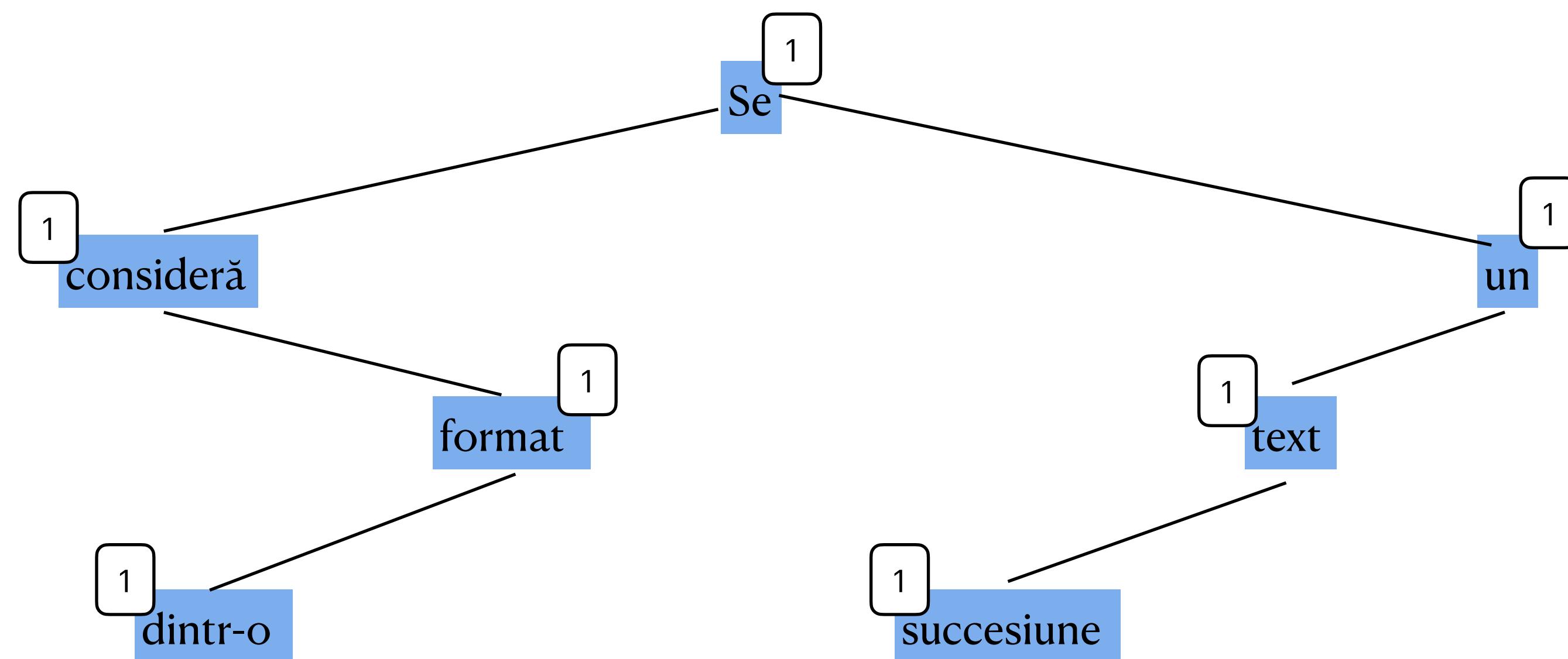
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



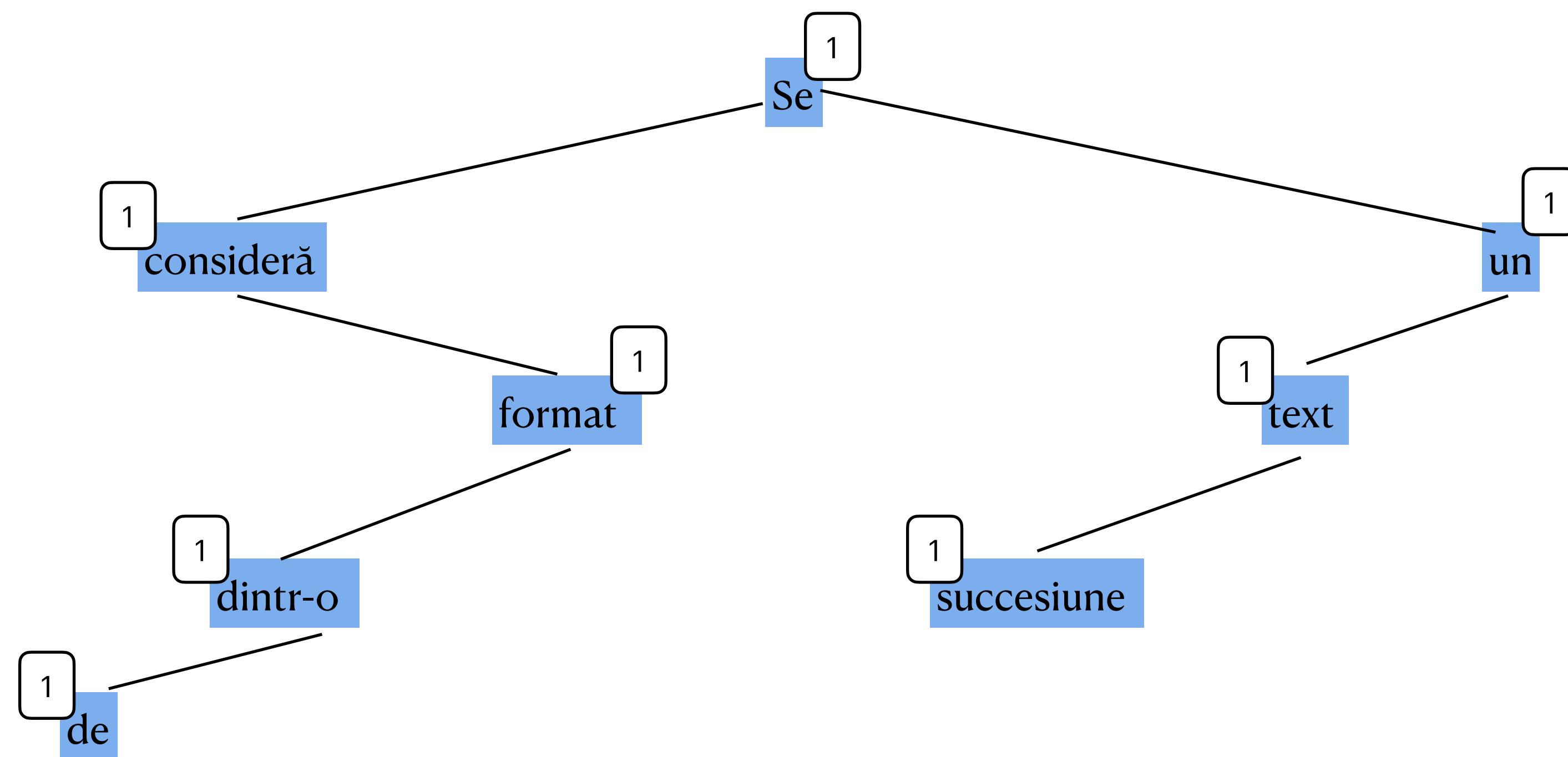
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



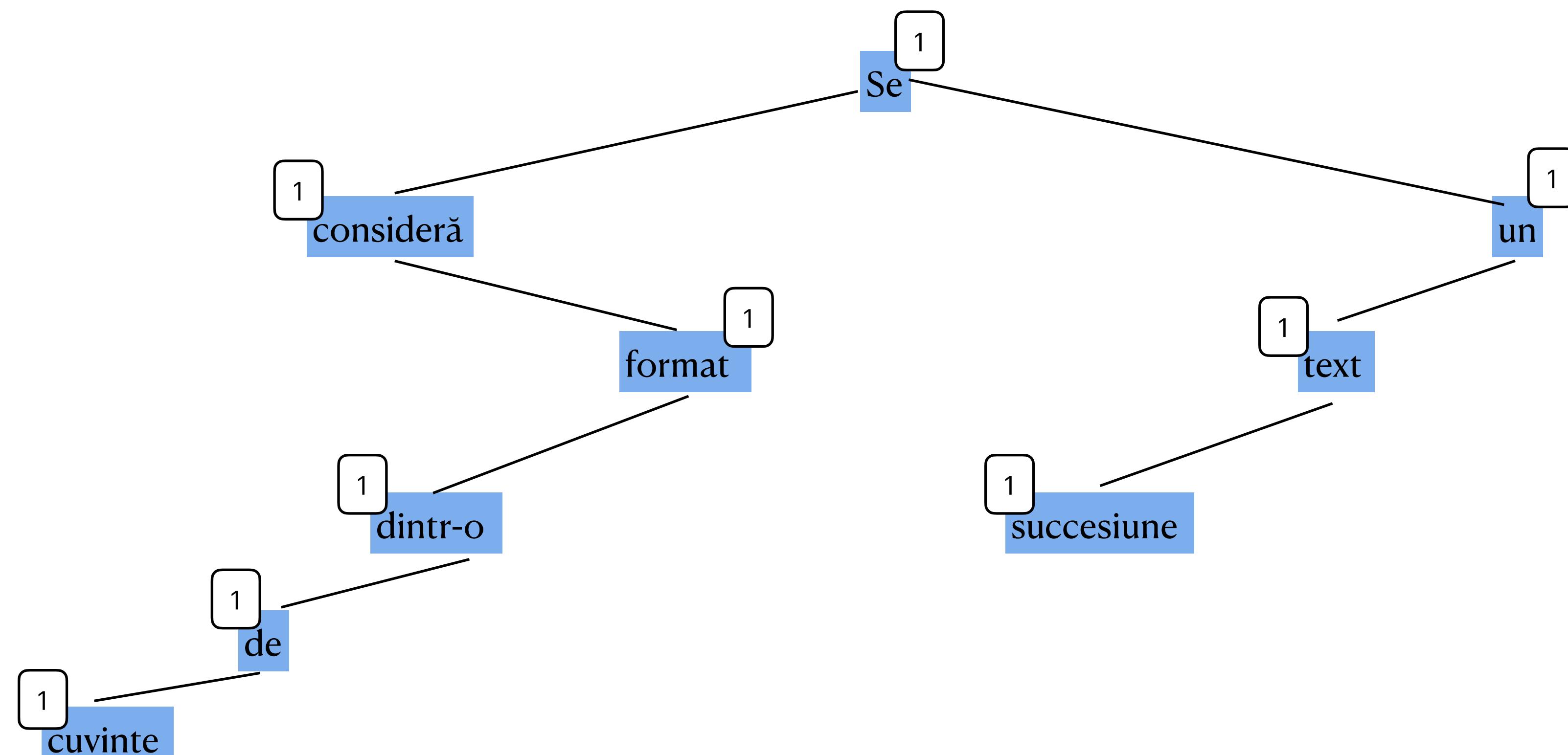
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



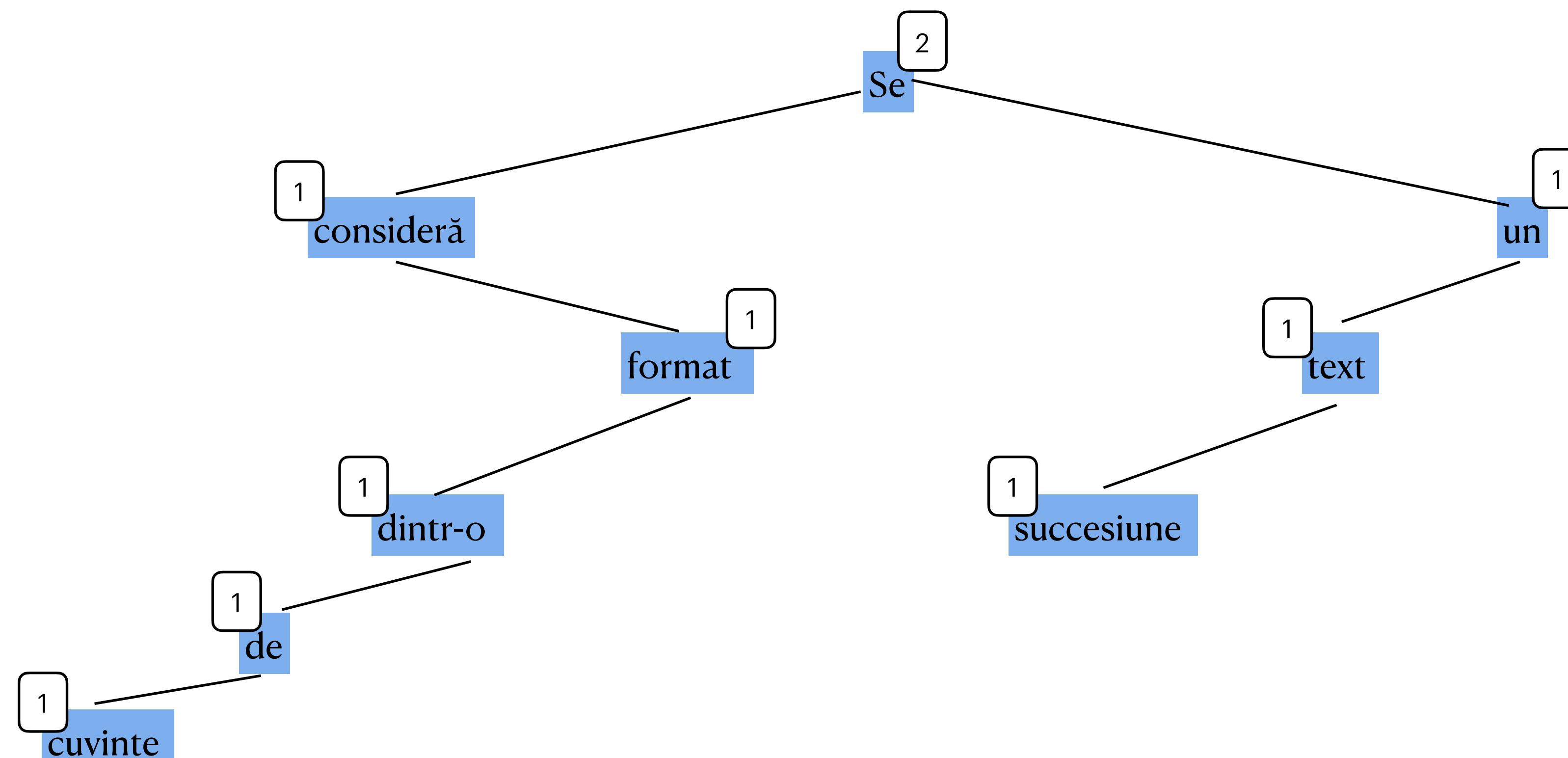
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



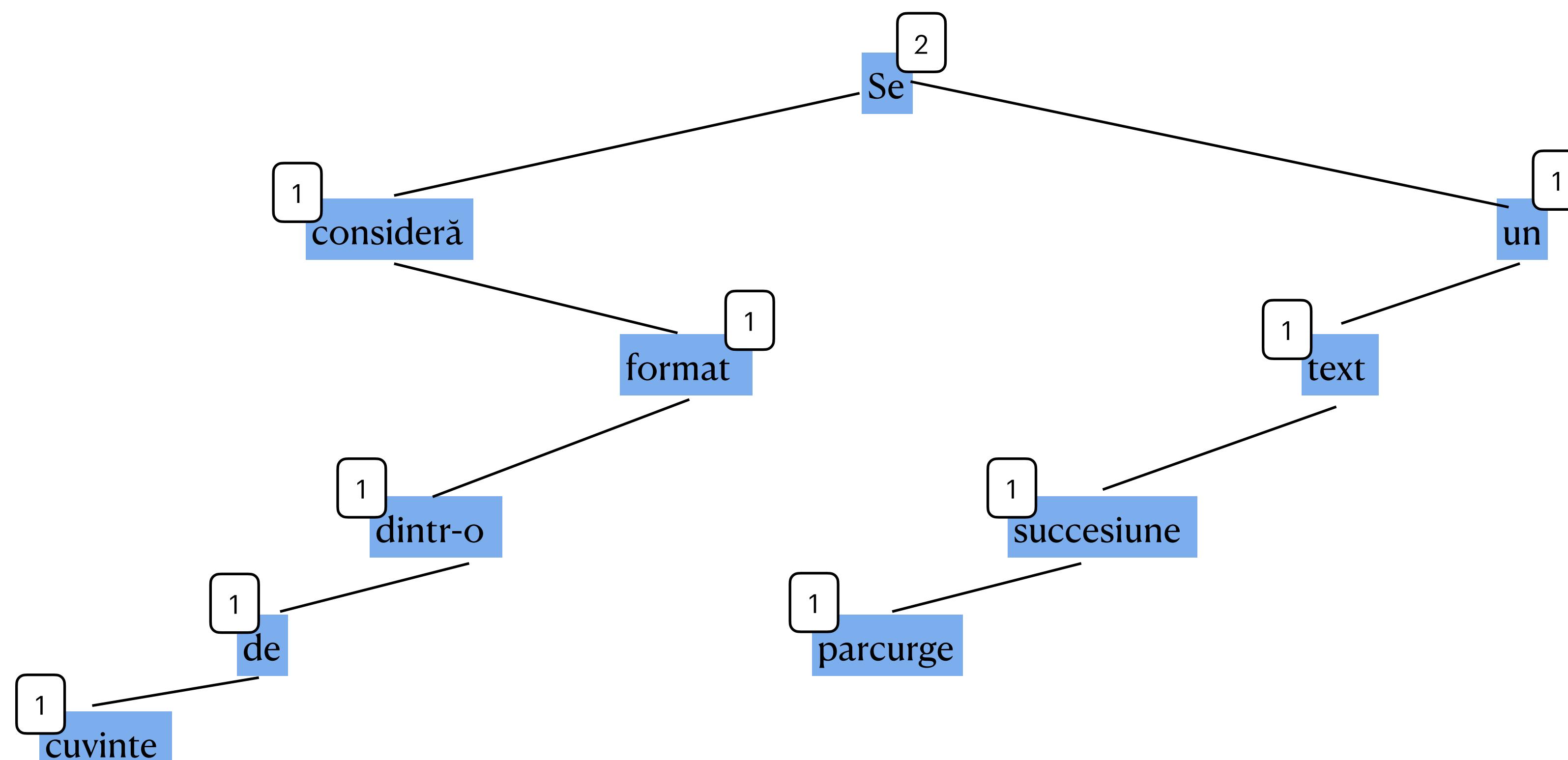
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



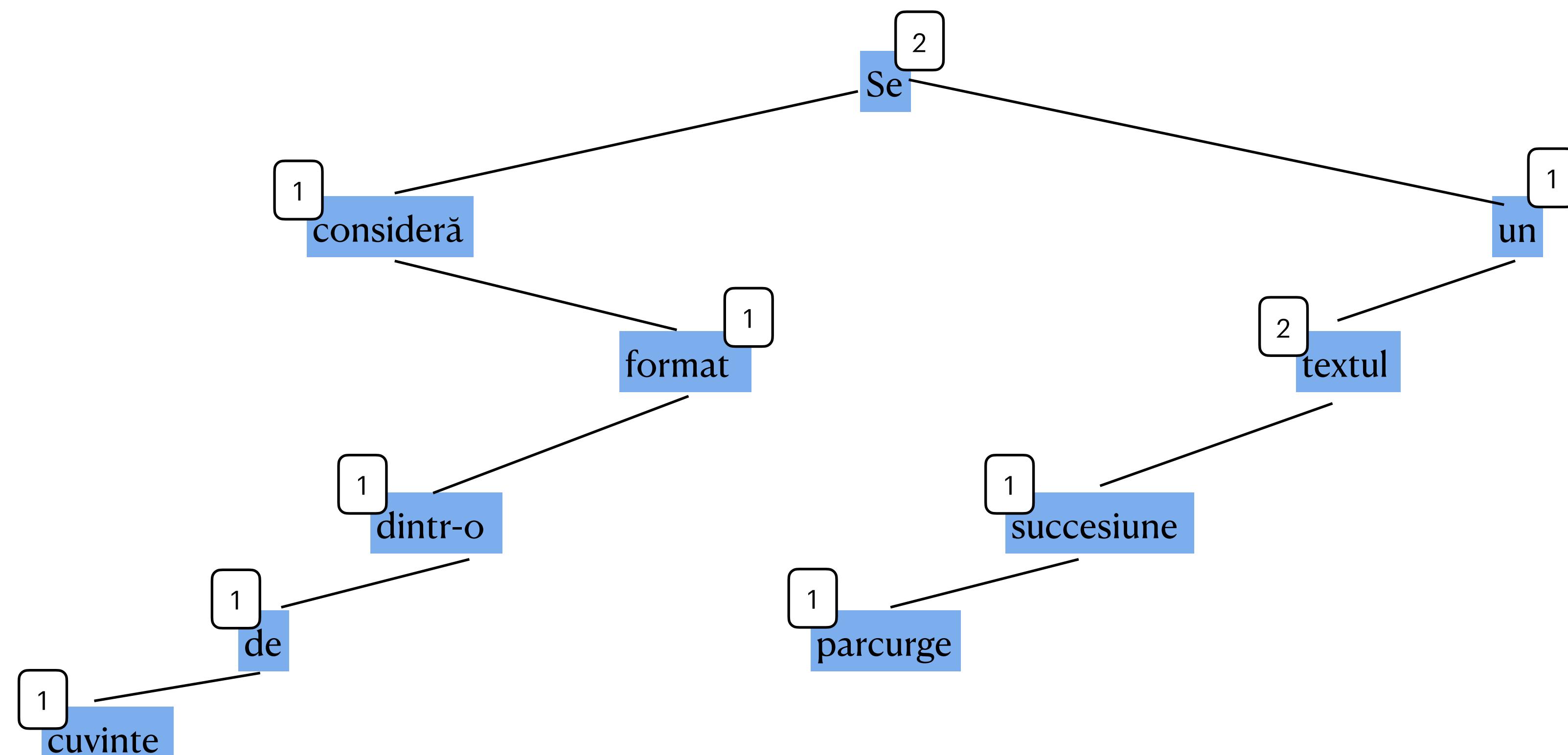
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



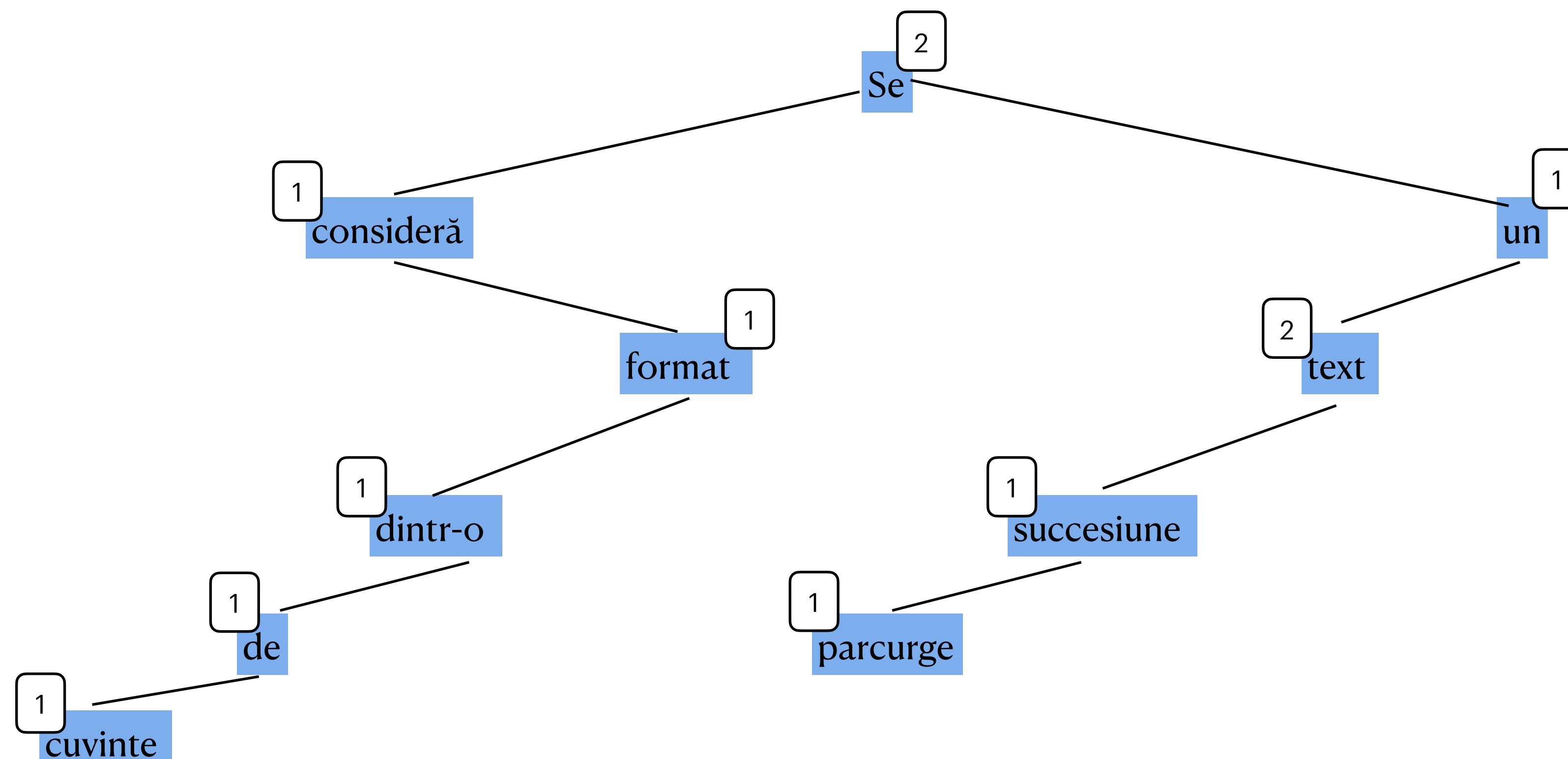
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



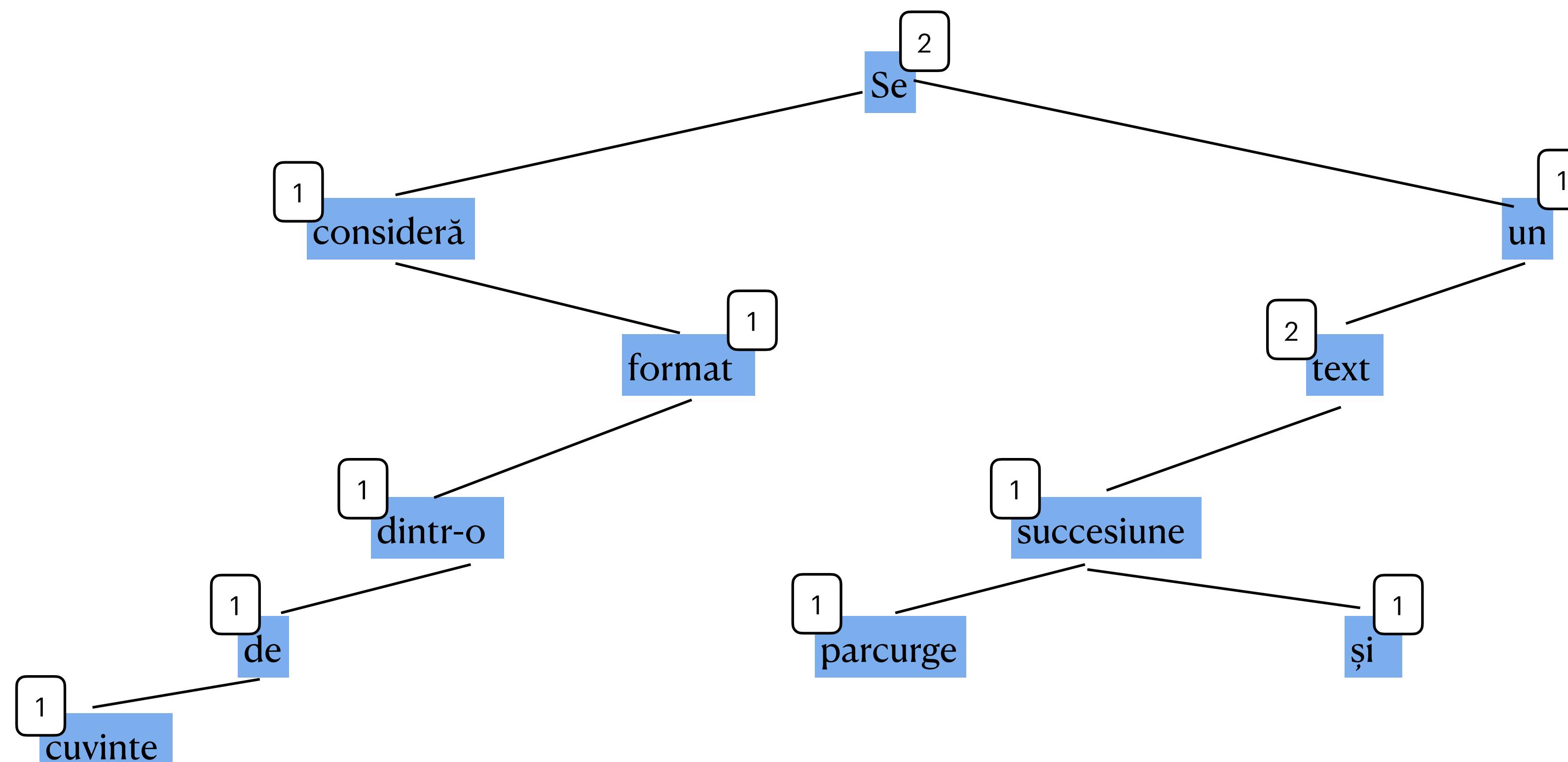
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



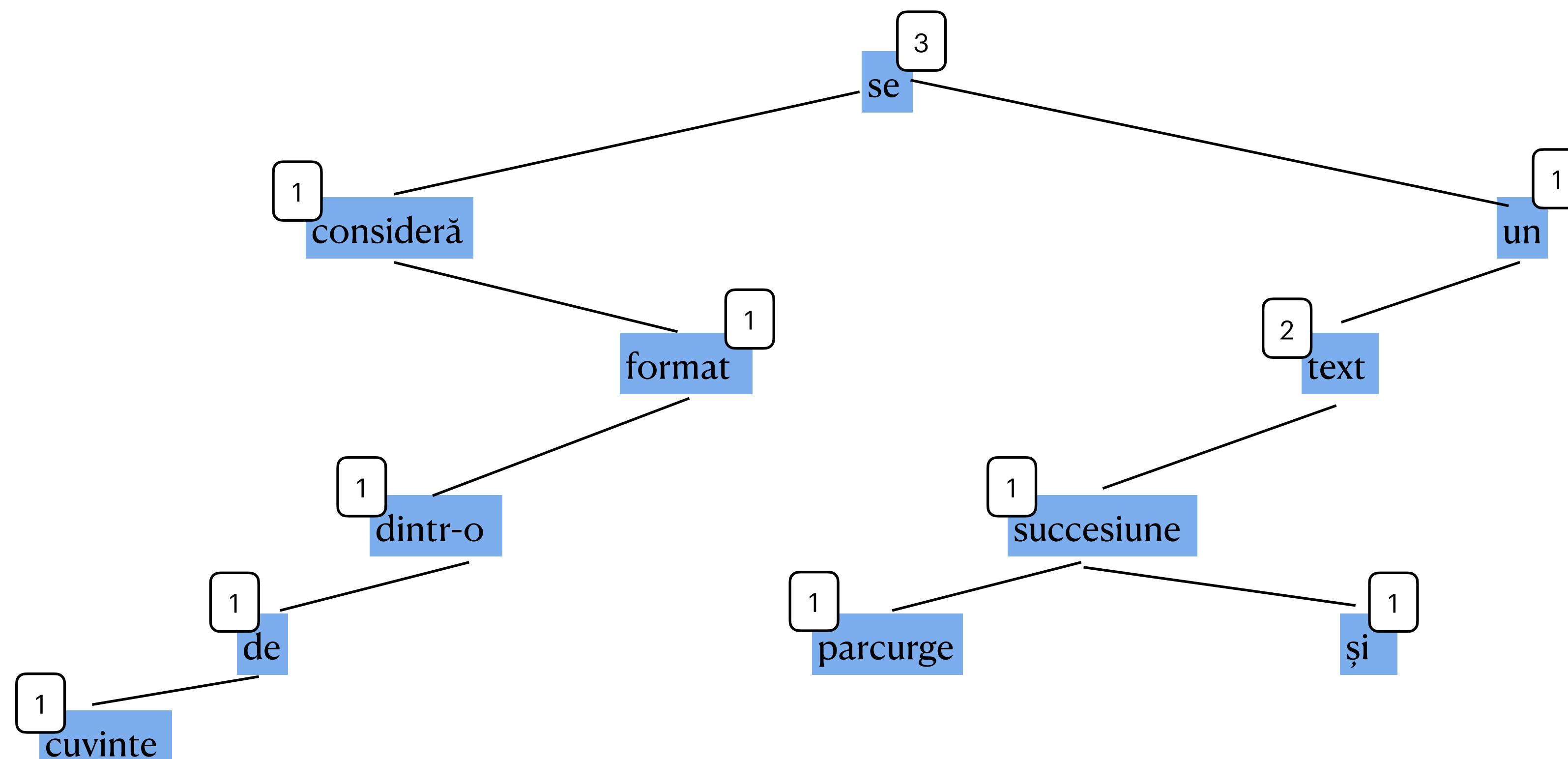
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



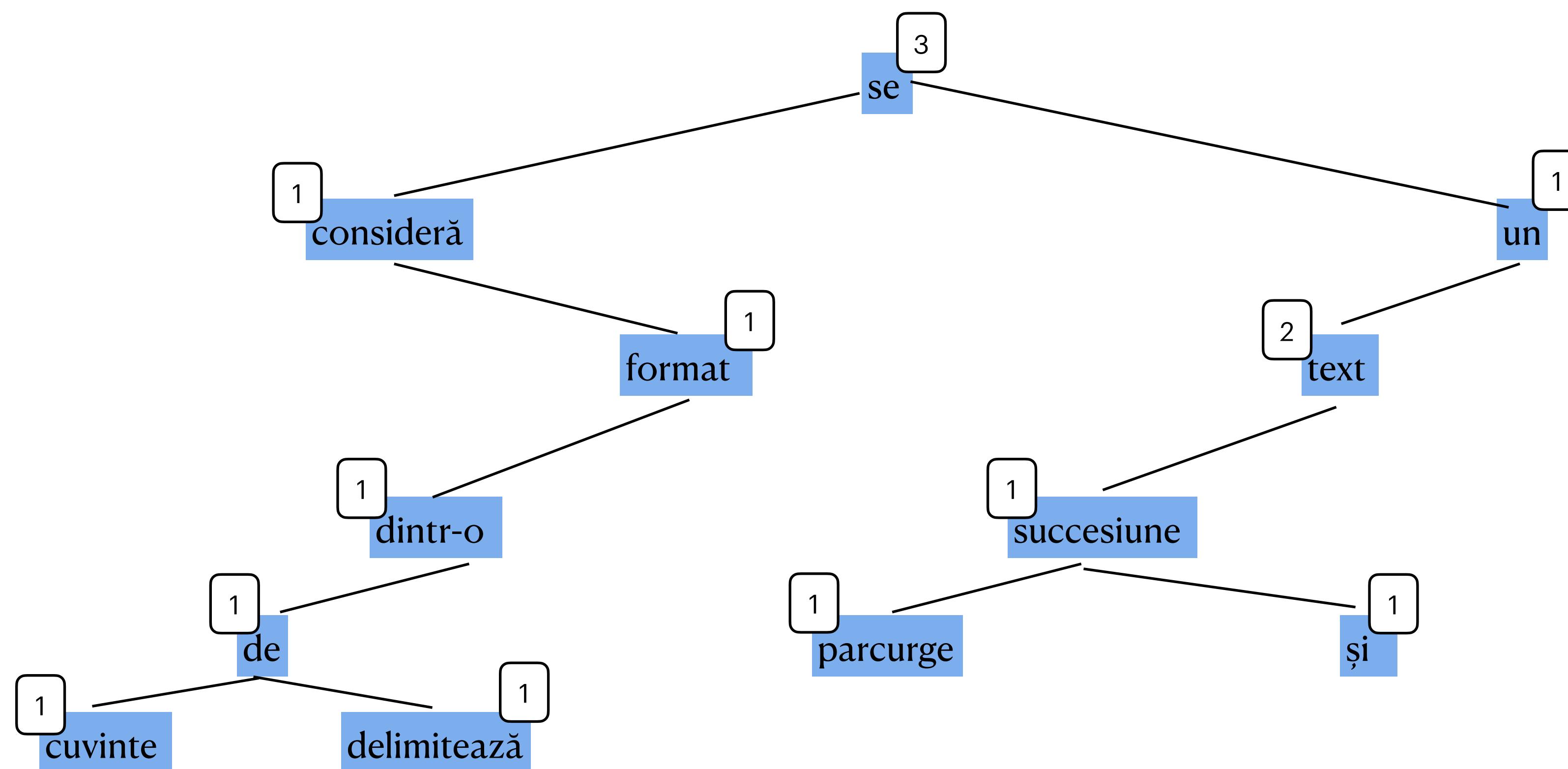
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



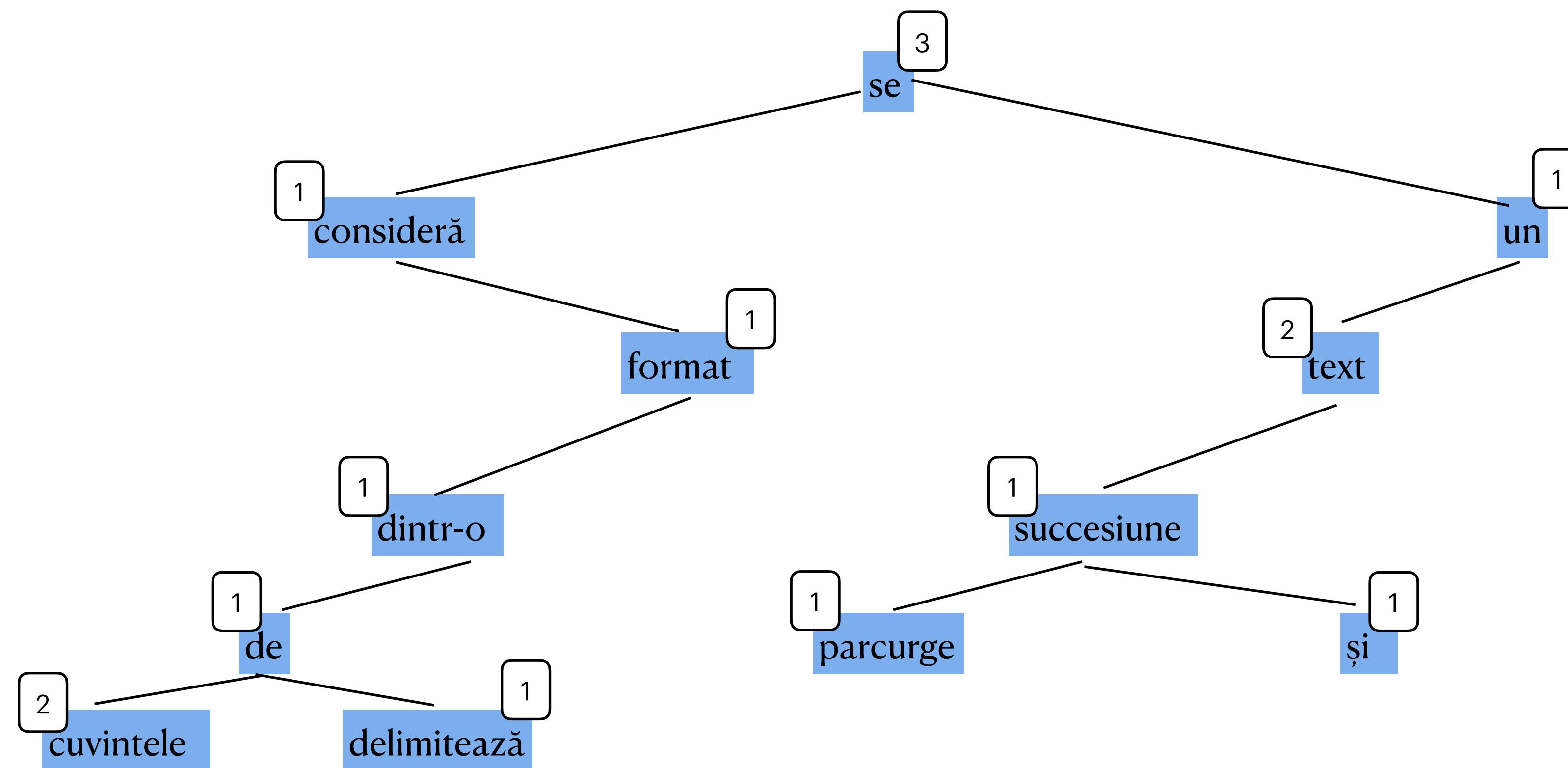
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



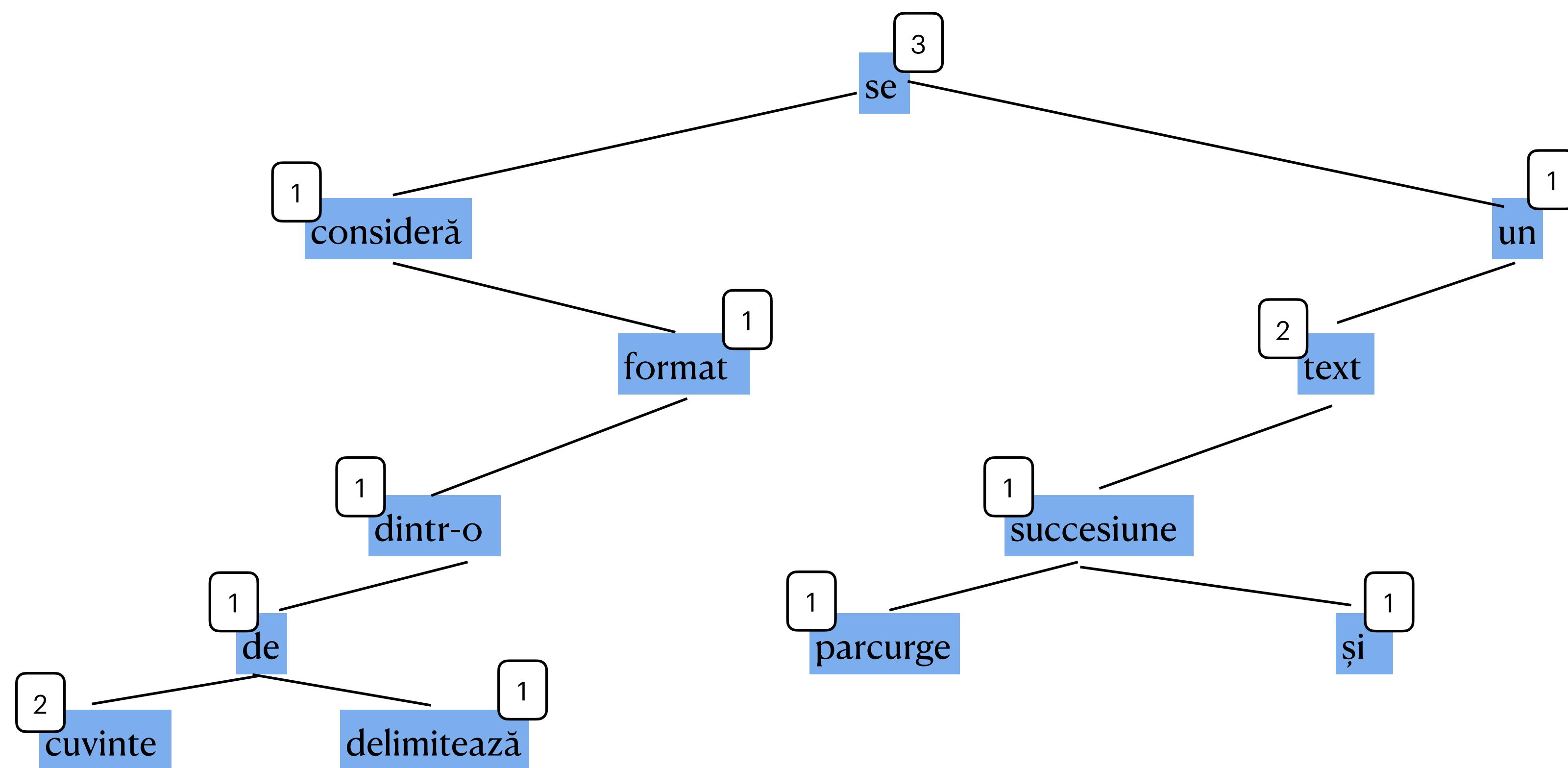
Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



Se consideră un text format dintr-o succesiune de cuvinte

Se parcurge textul și se delimitează cuvintele



Arbore Binari pentru Expresii

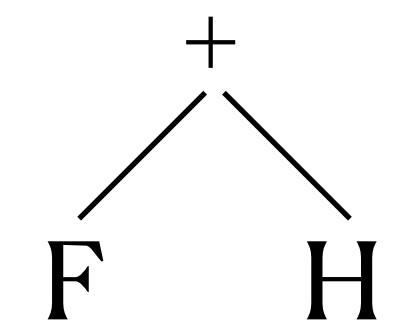
Folosit pentru a reprezenta expresii (algebrice, booleene, etc..)

$$A + B^*C + D \uparrow E^*(F+H)$$

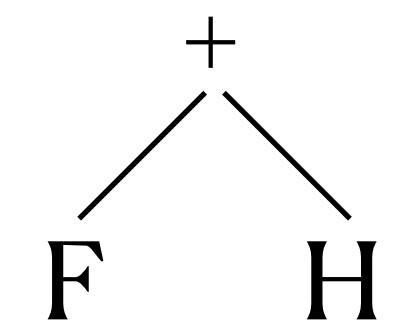
A + B * C + D ↑ E * (F + H)

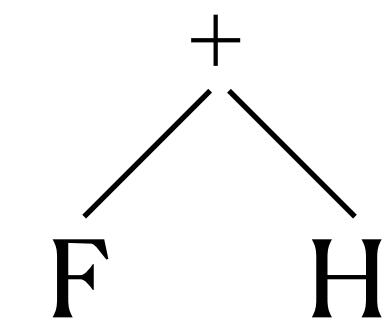
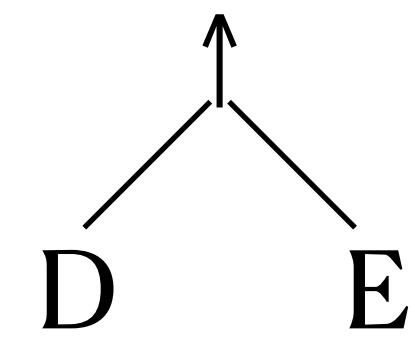
A + B * C + D ↑ E * (F + H)

A + B * C + D ↑ E * (F + H)

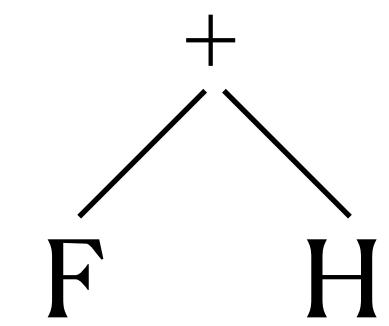
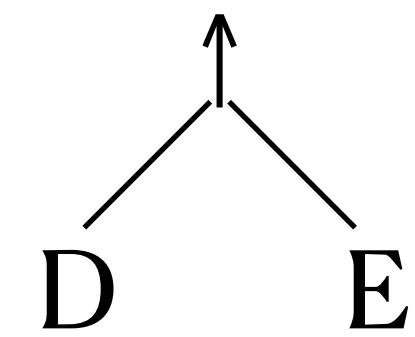


A + B * C + D ↑ E * (F + H)

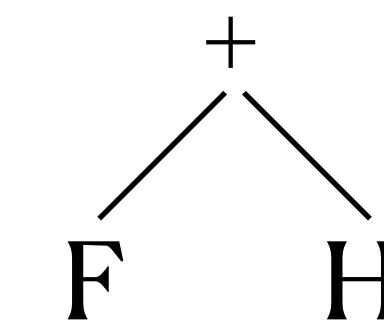
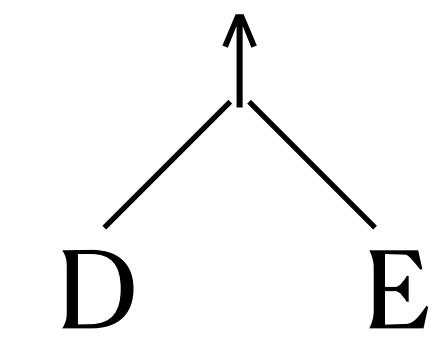
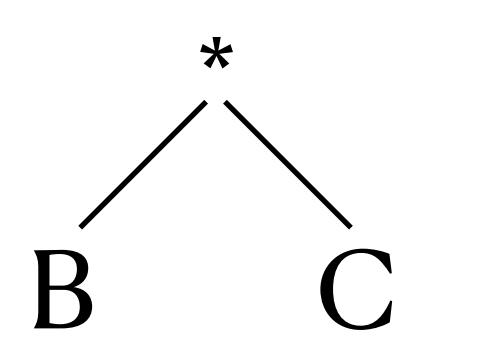


$$A + B * C + D \uparrow E * (F + H)$$


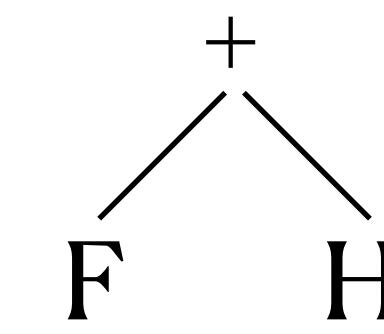
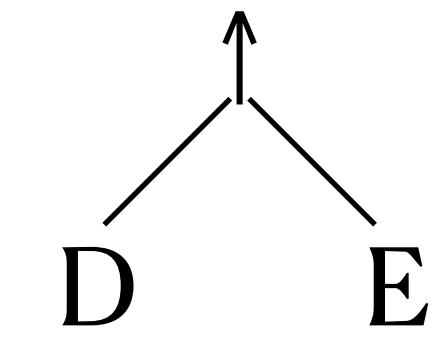
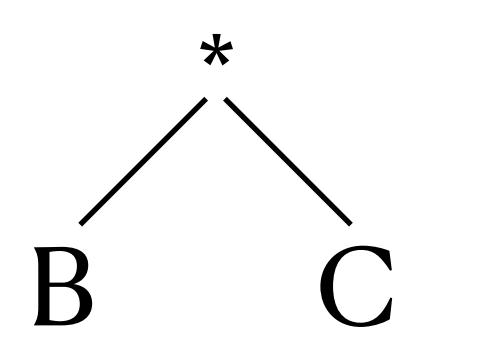
A + B * C + D ↑ E * (F + H)



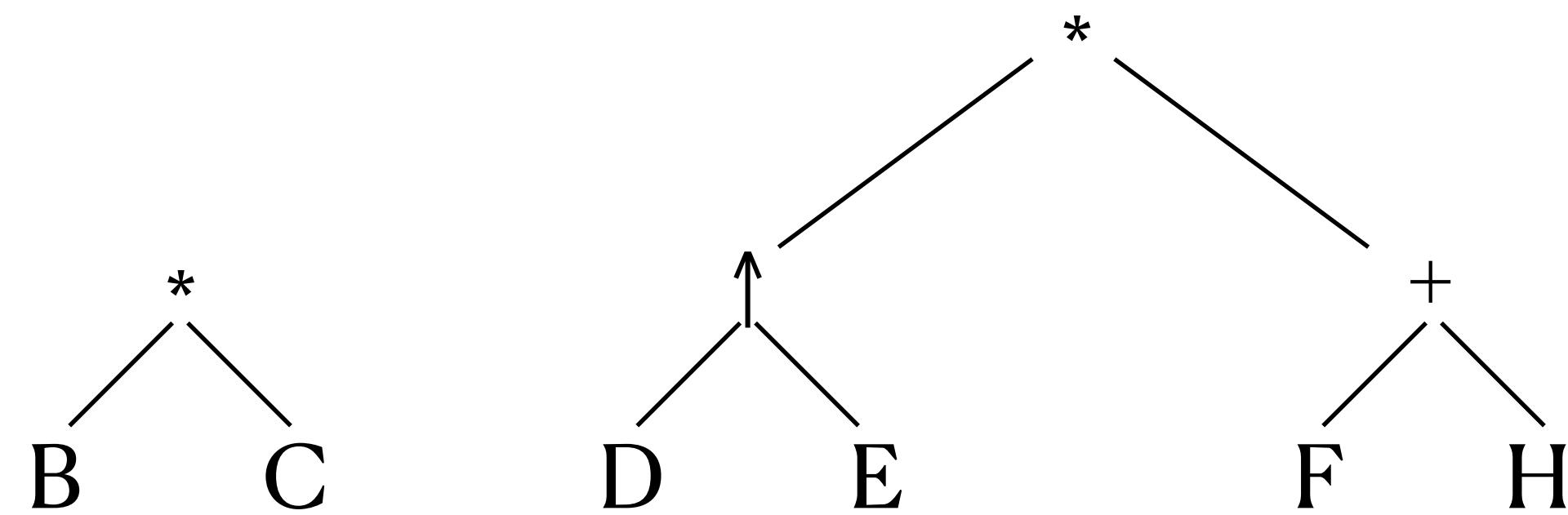
A + B * C + D ↑ E * (F + H)



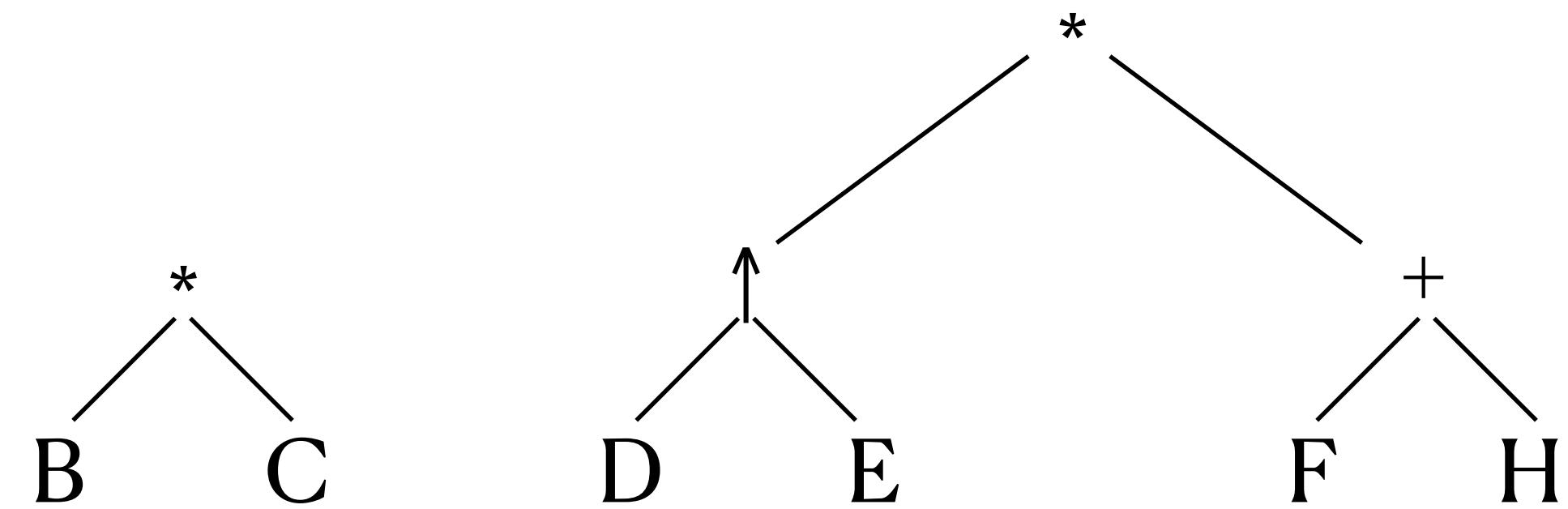
A + B * C + D ↑ E * (F + H)



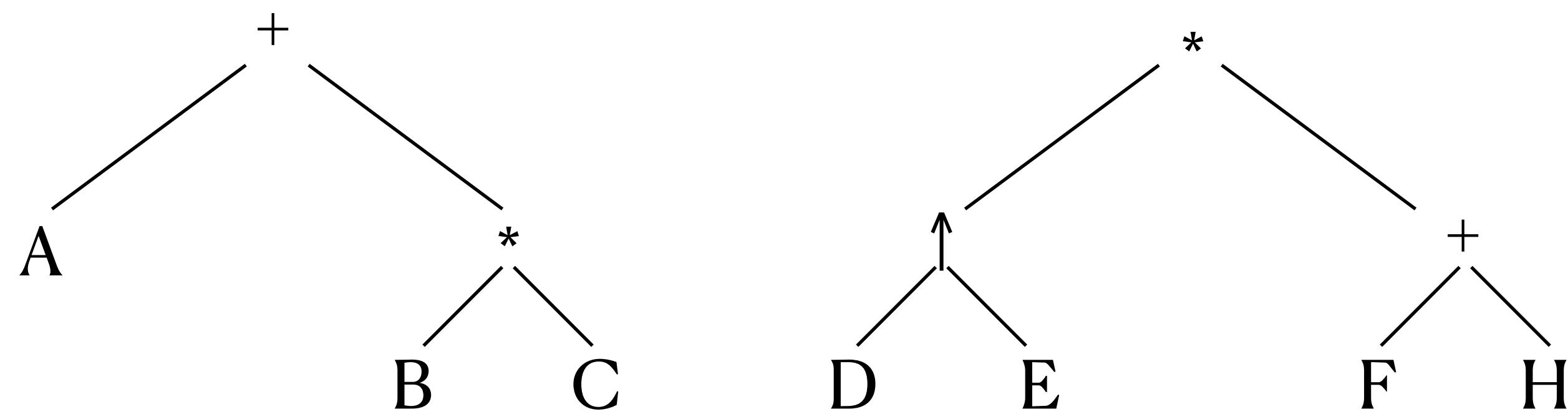
A + B * C + D ↑ E * (F + H)



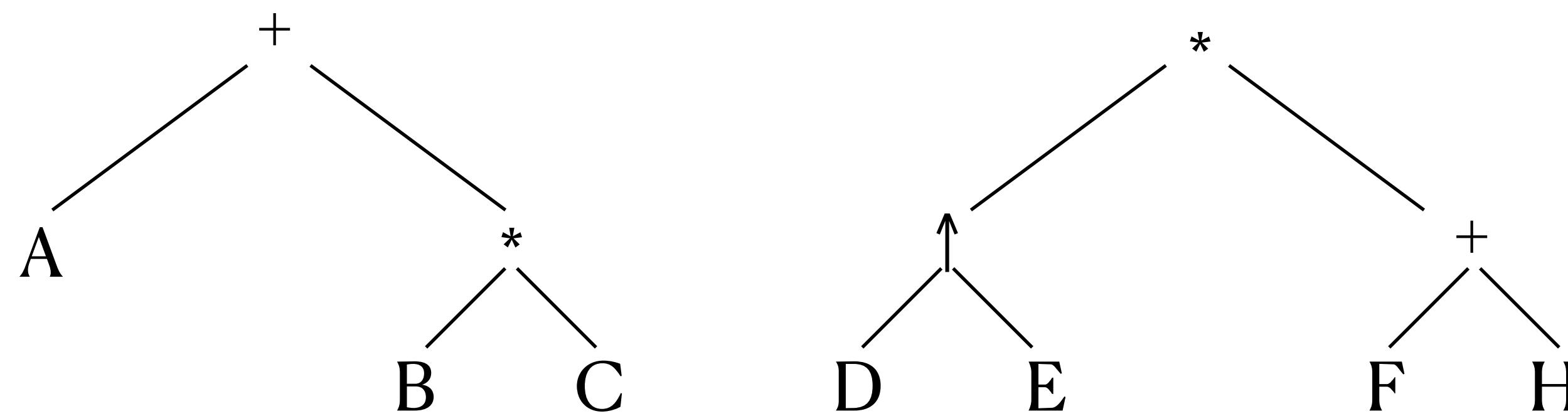
A + B * C + D ↑ E * (F + H)



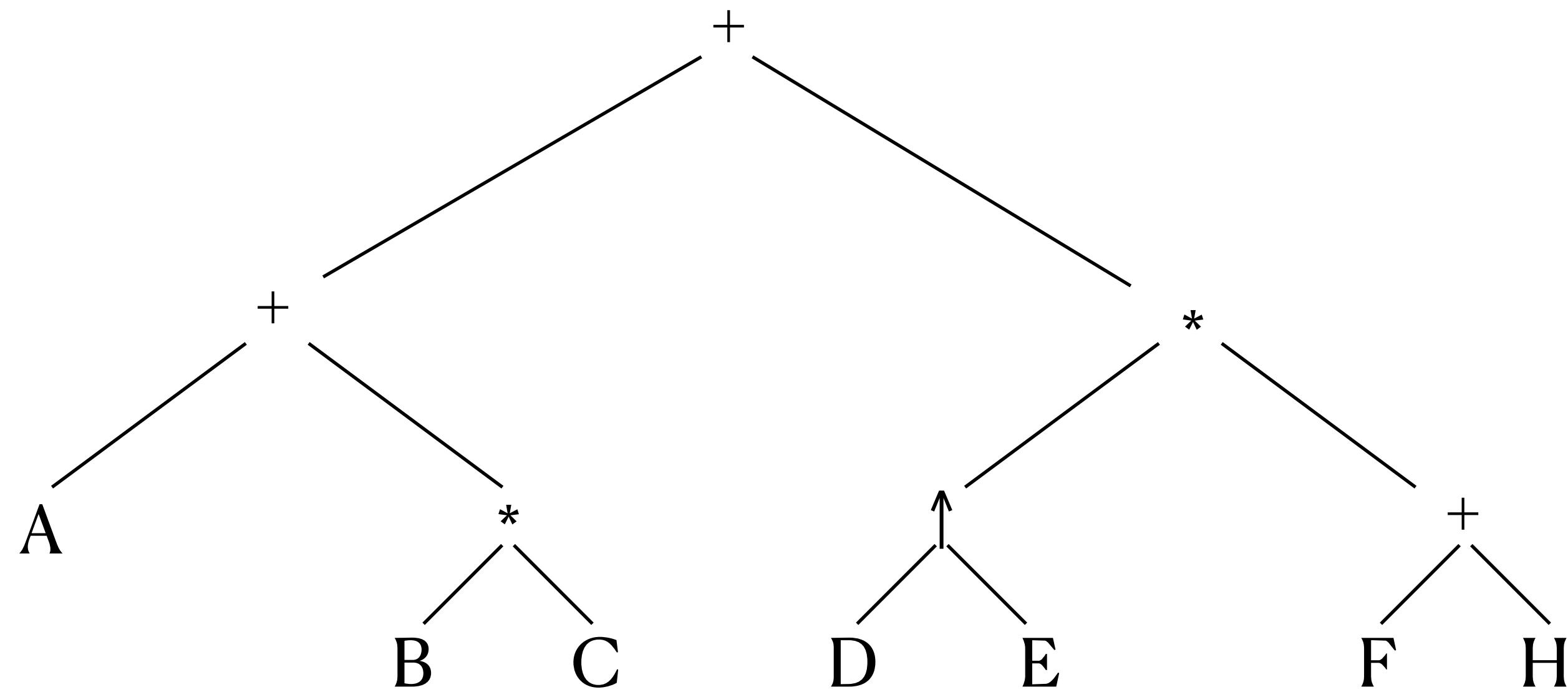
A + B * C + D ↑ E * (F + H)



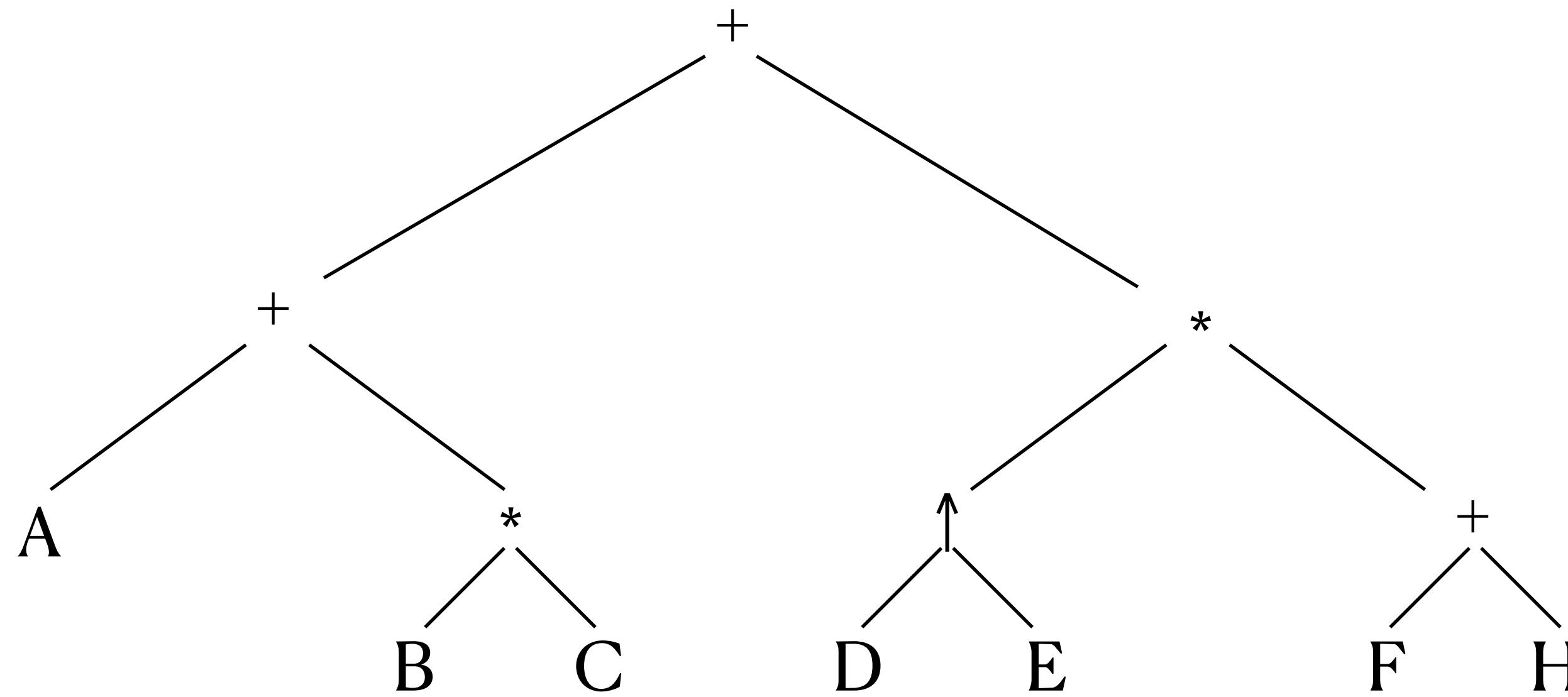
A + B * C + D ↑ E * (F + H)



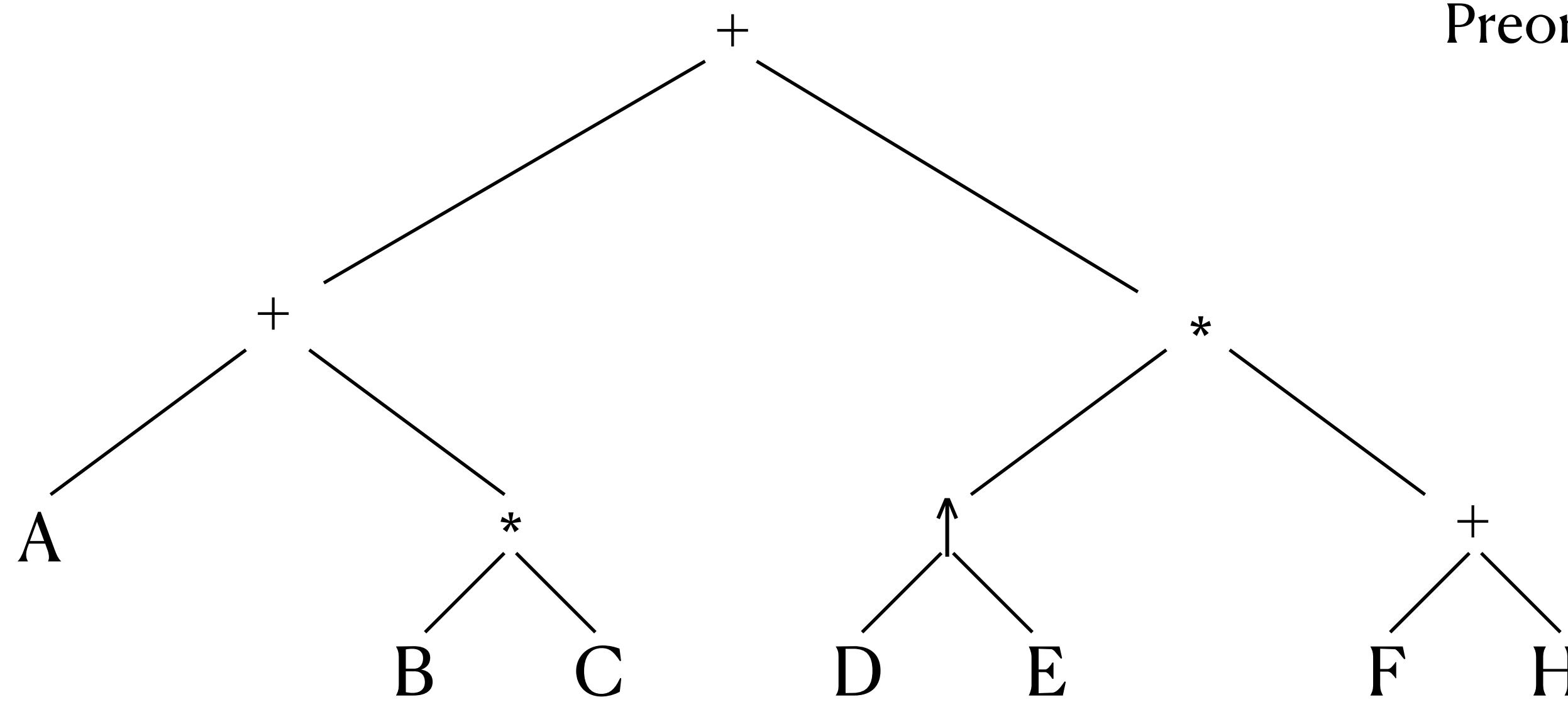
A + B * C + D ↑ E * (F + H)



A + B * C + D ↑ E * (F + H)

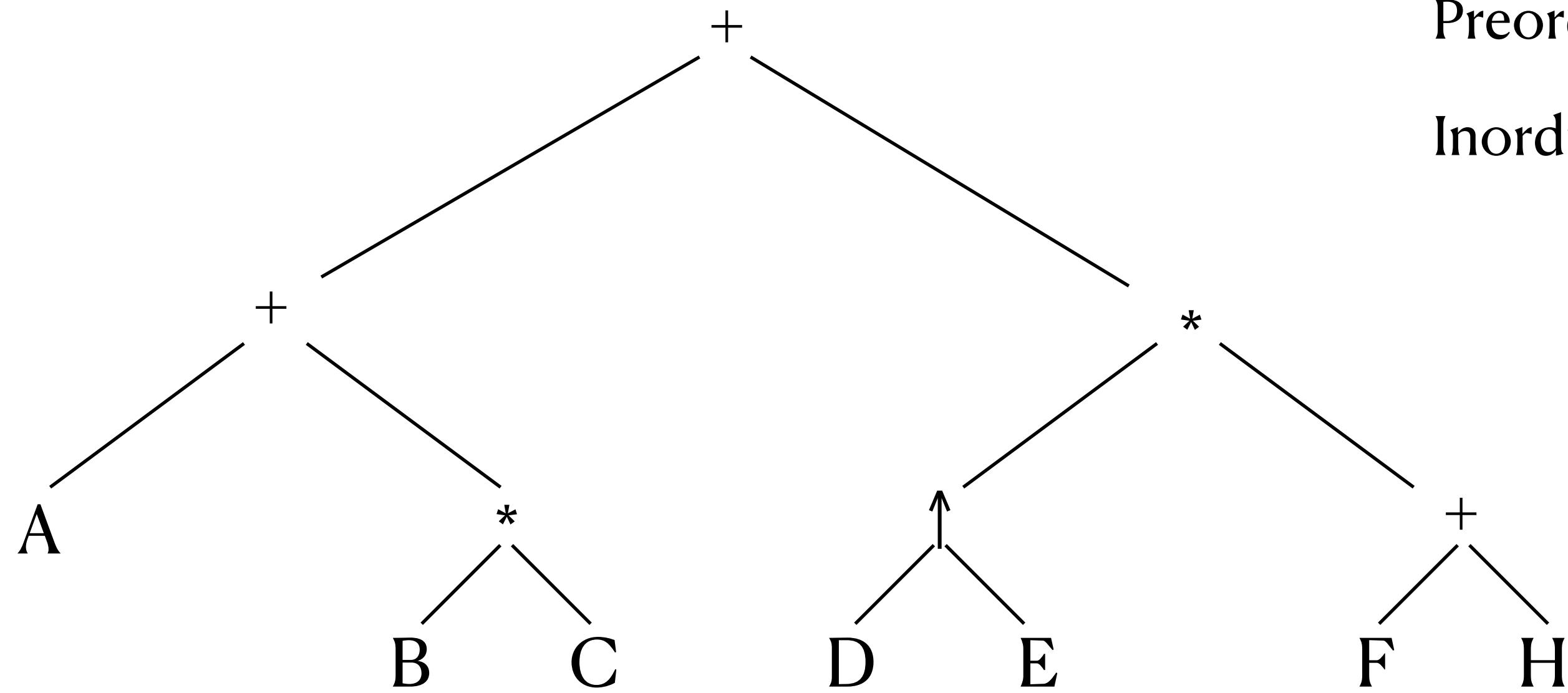


A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

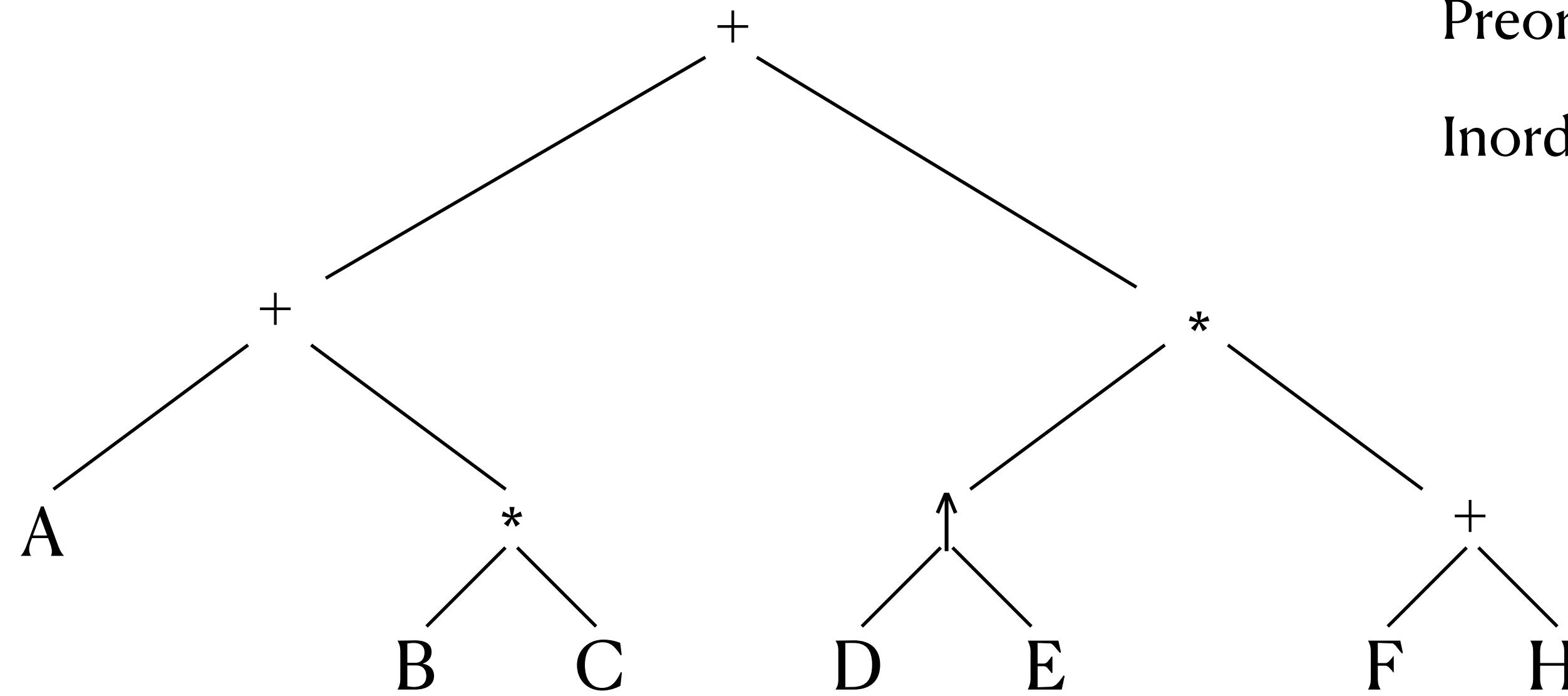
A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C))

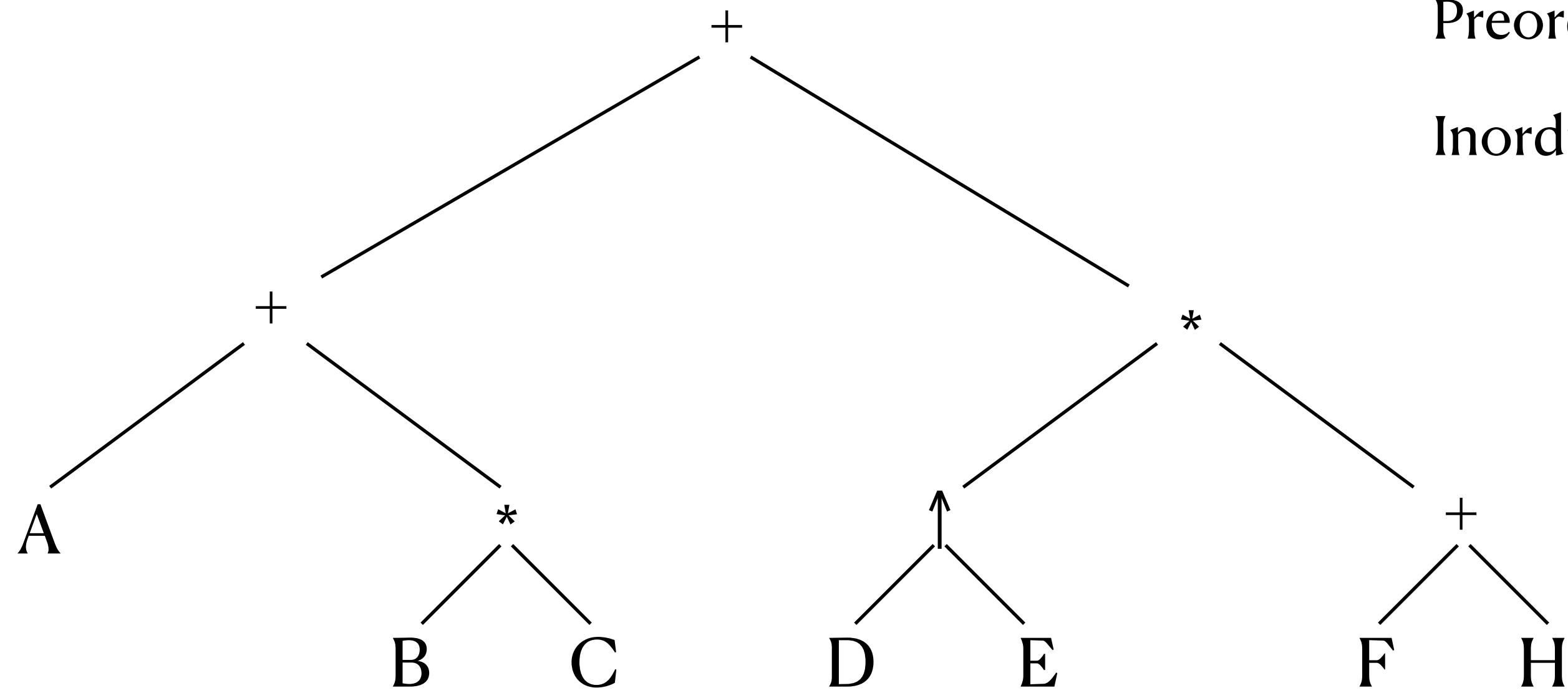
A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C)) +

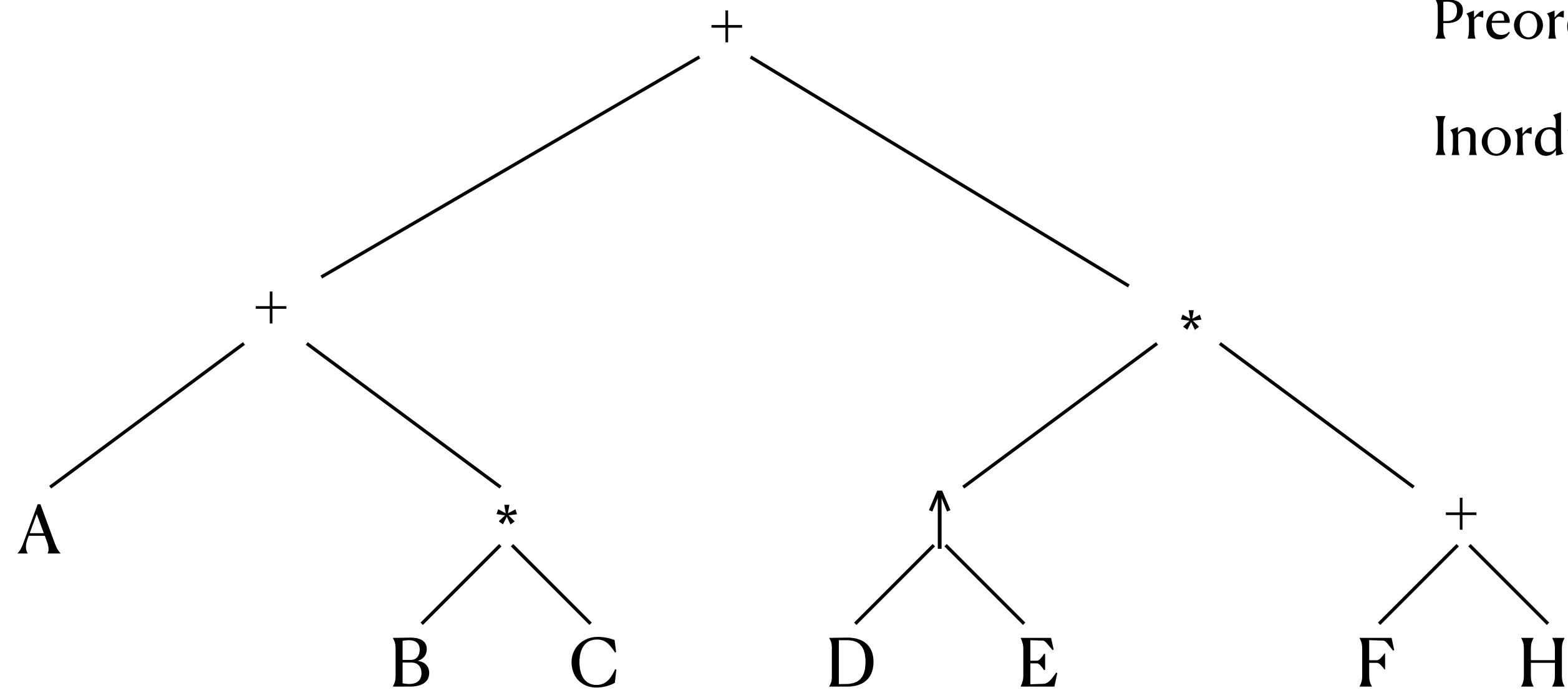
A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C)) + ((D ↑ E))

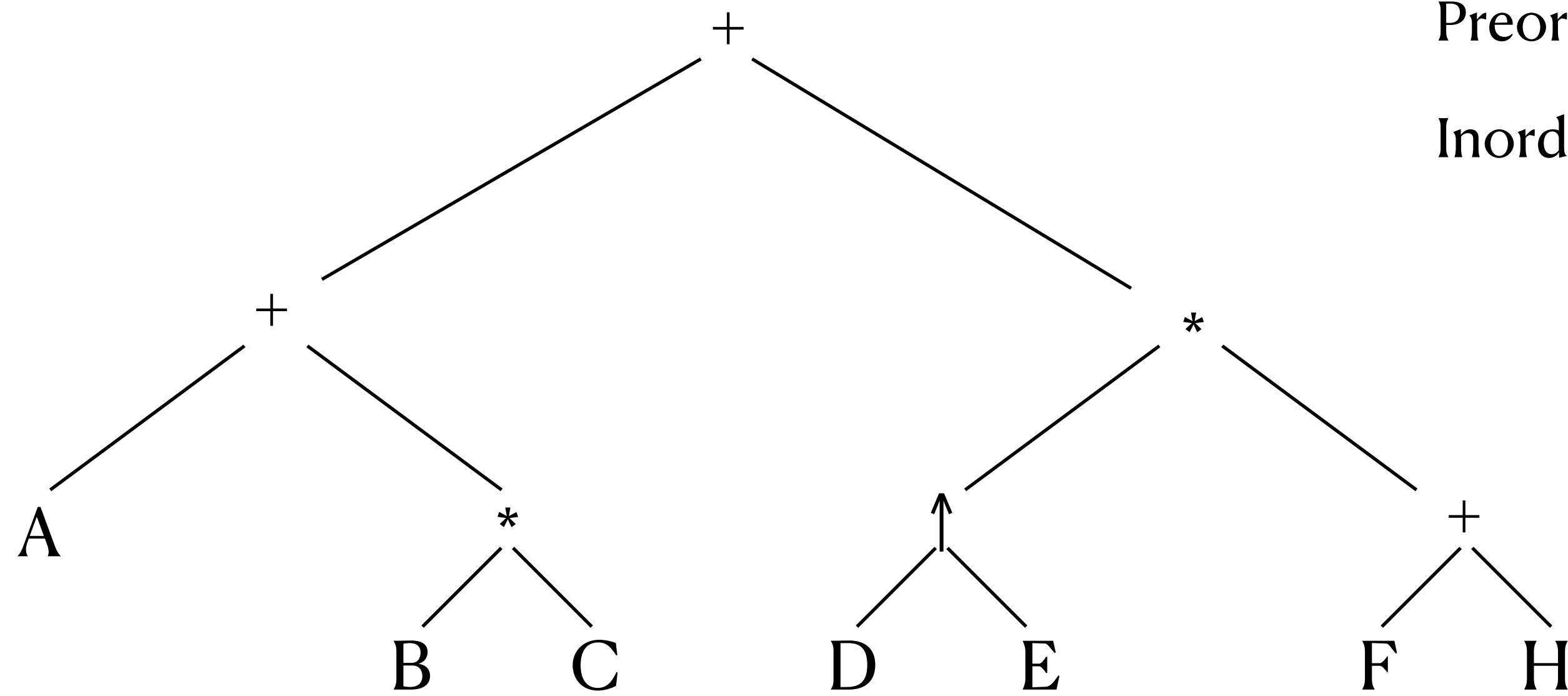
A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C)) + ((D ↑ E) *

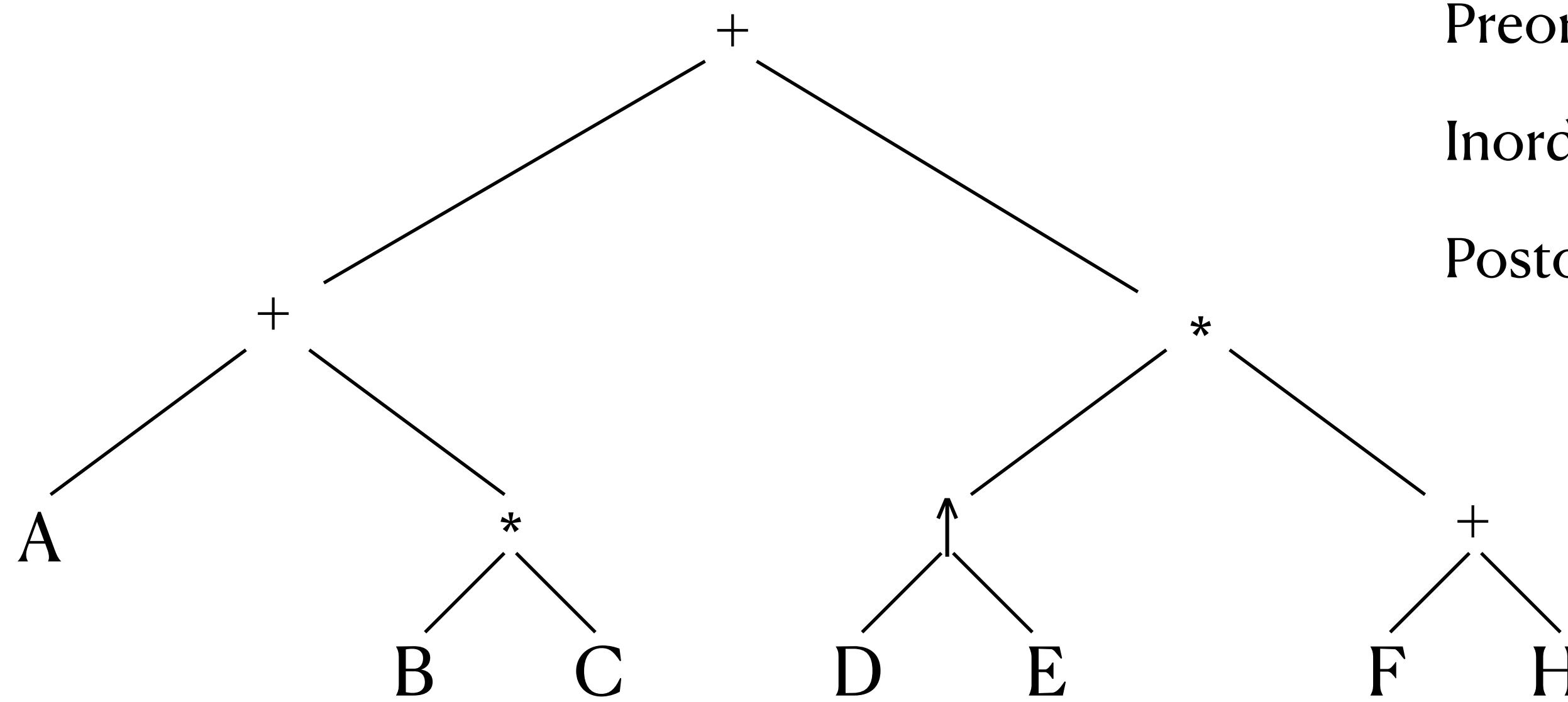
A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

A + B * C + D ↑ E * (F + H)



Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

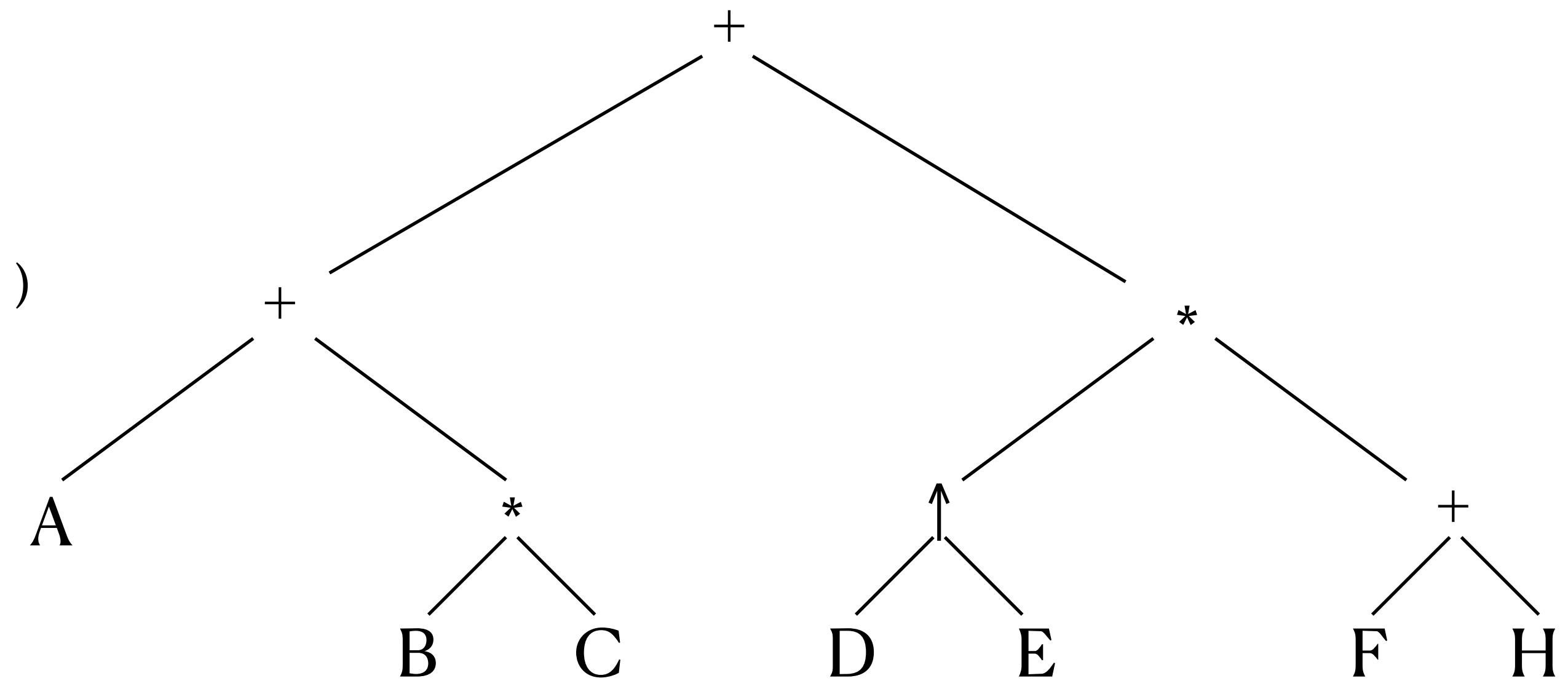
Postordine: A B C * + D E ↑ F H + * +

A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +



A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

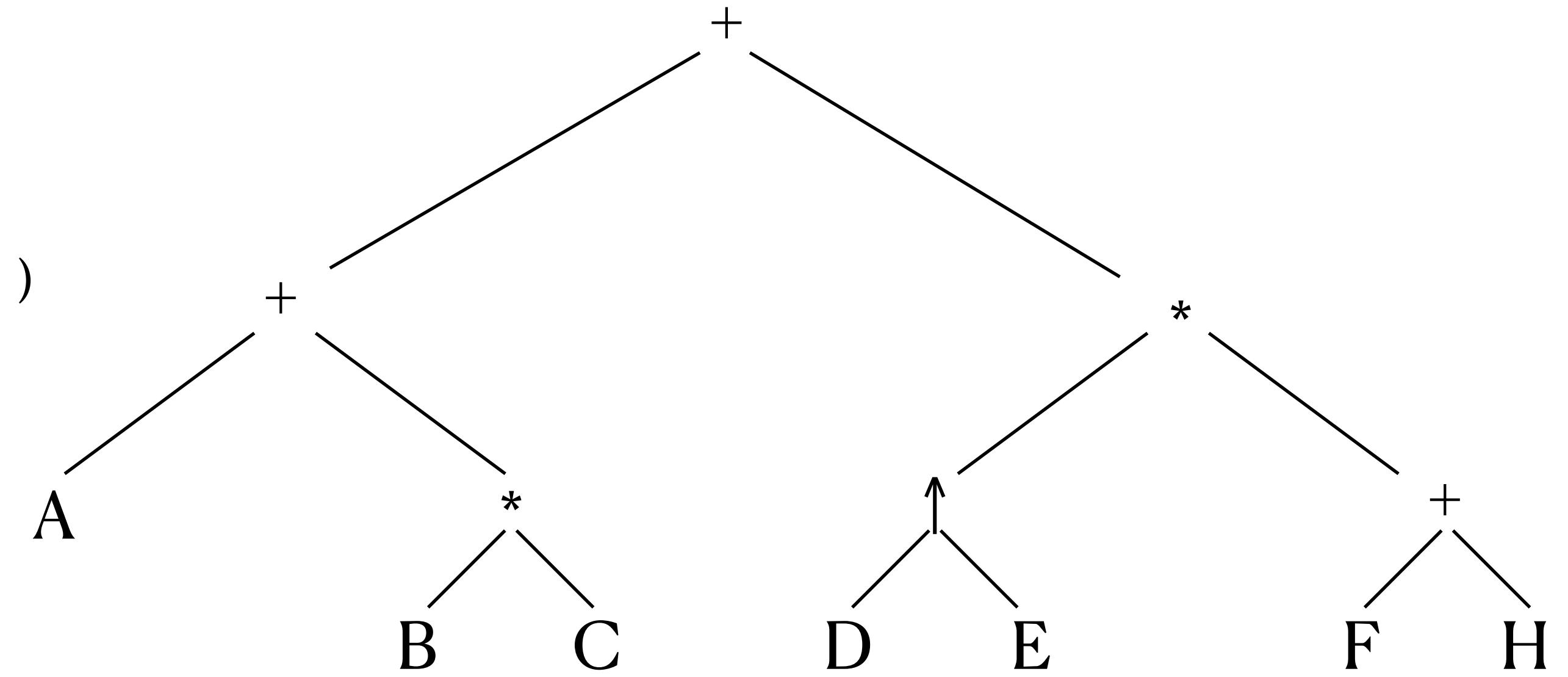
Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

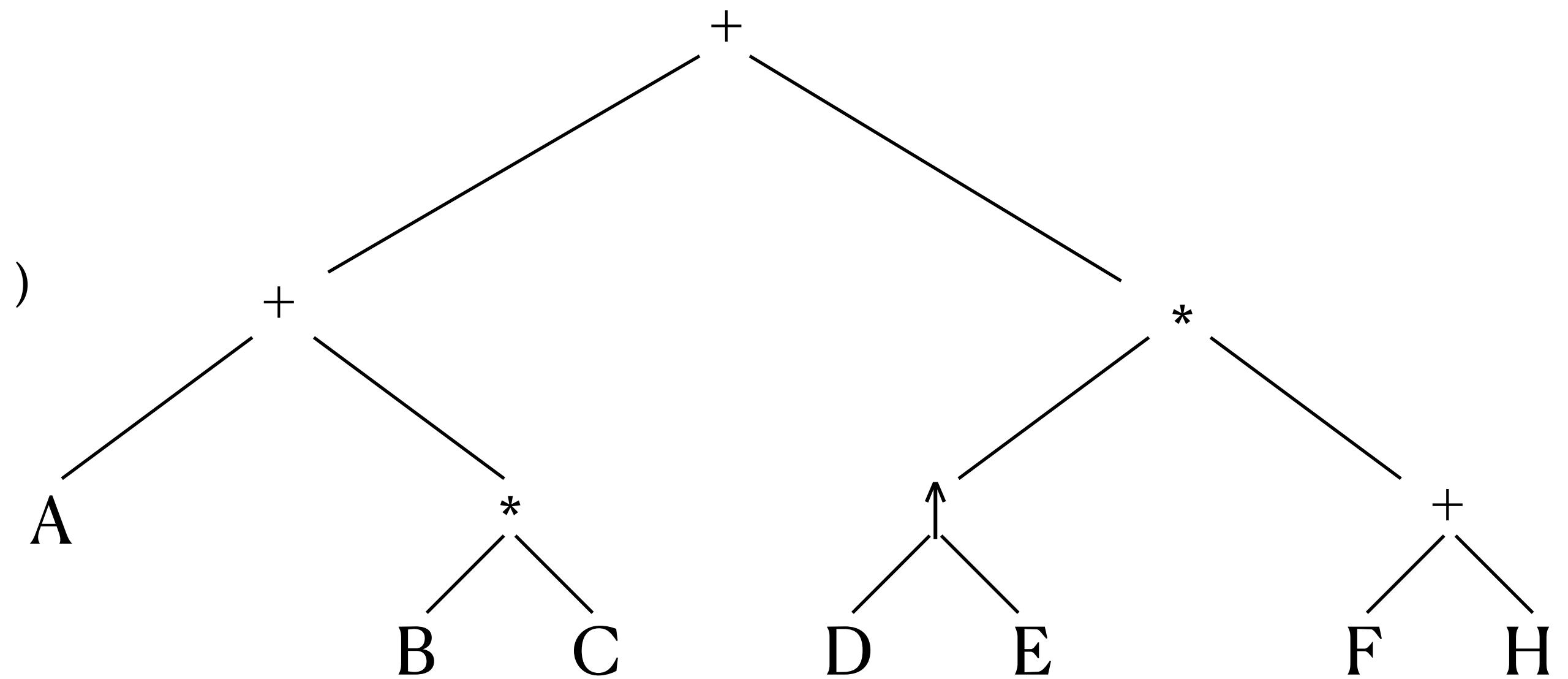
Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

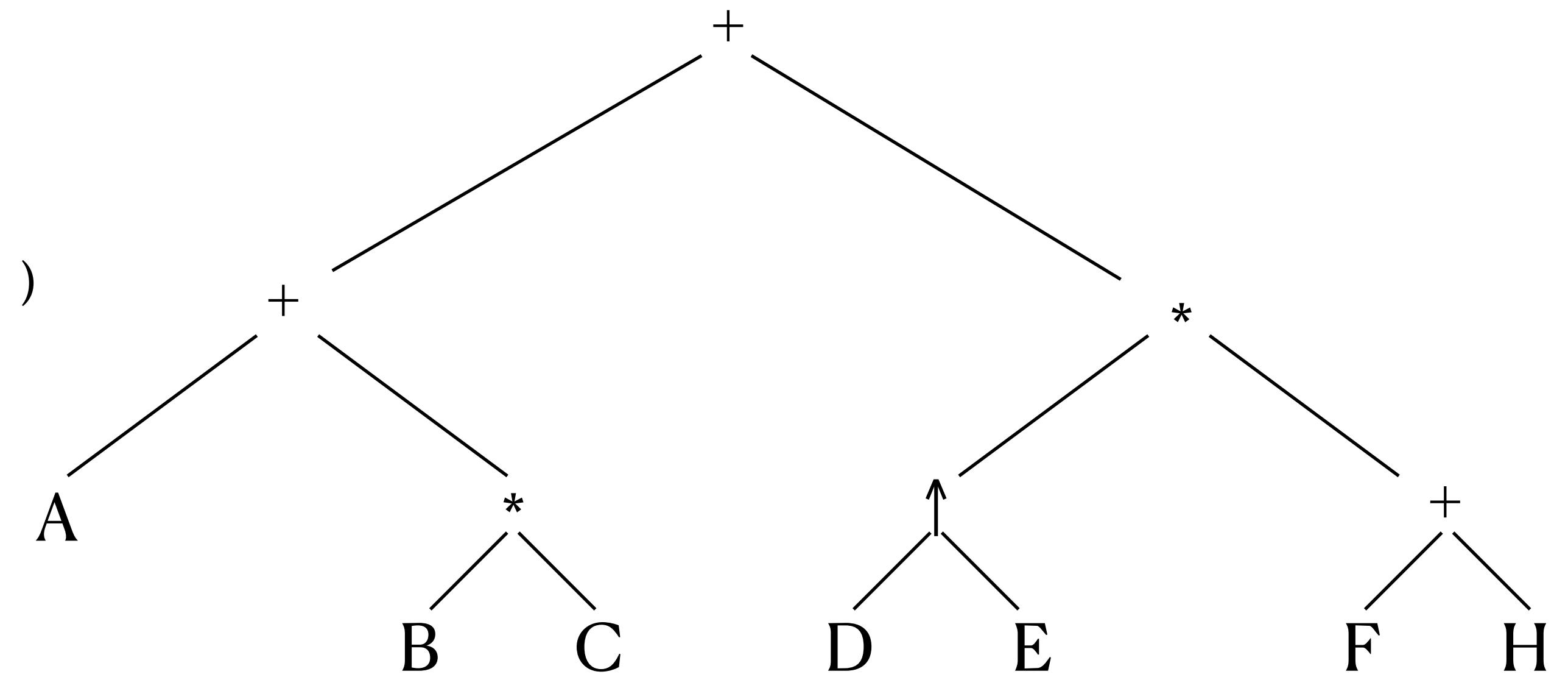
Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

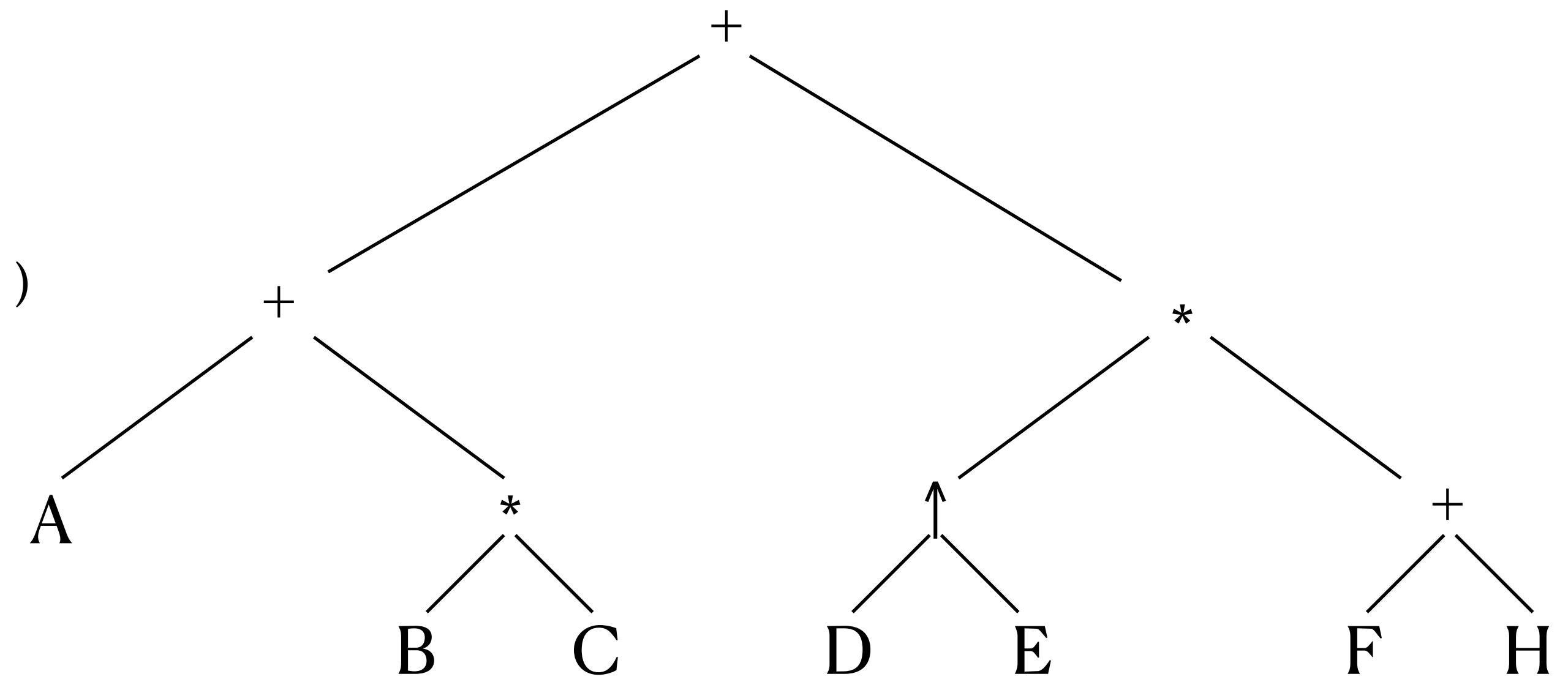
Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

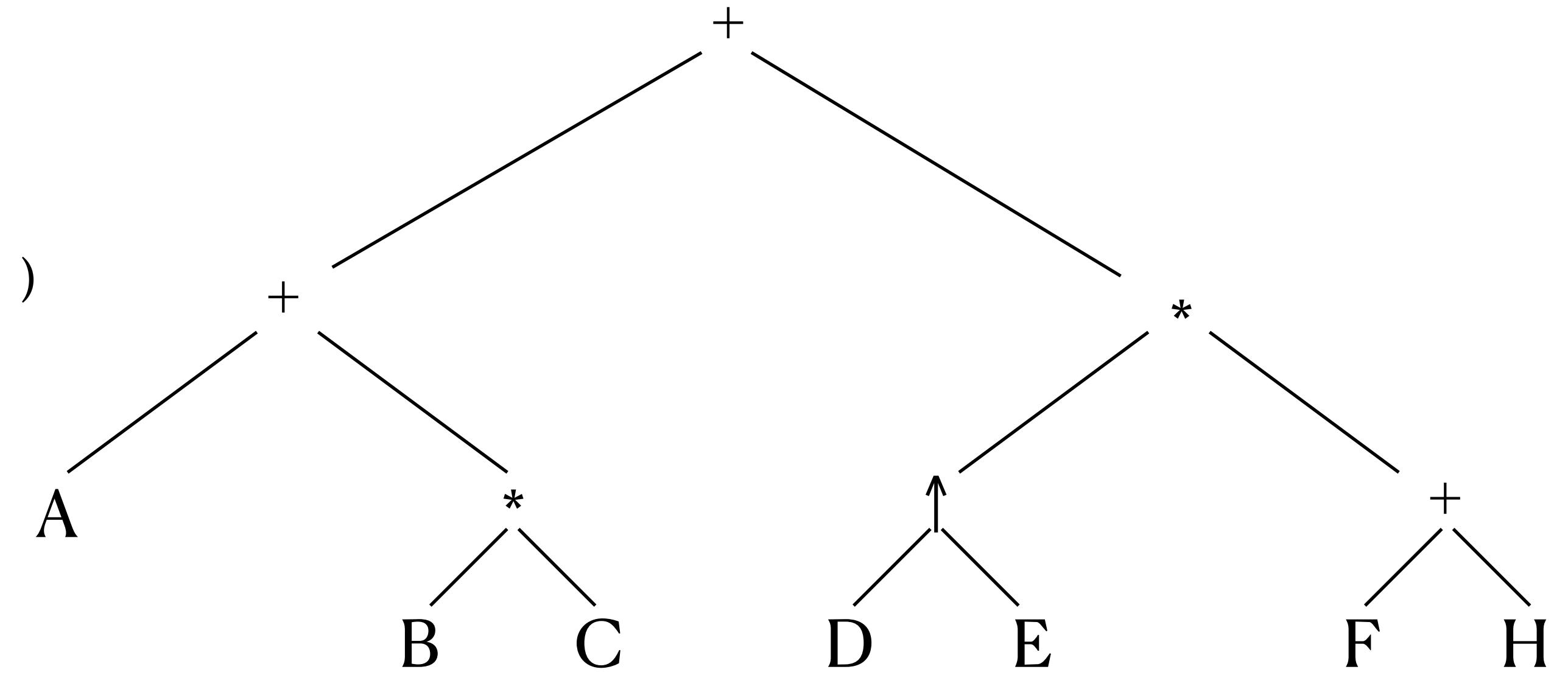
Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



A + B * C + D ↑ E * (F + H)

Preordine: + + A * B C * ↑ D E + F H

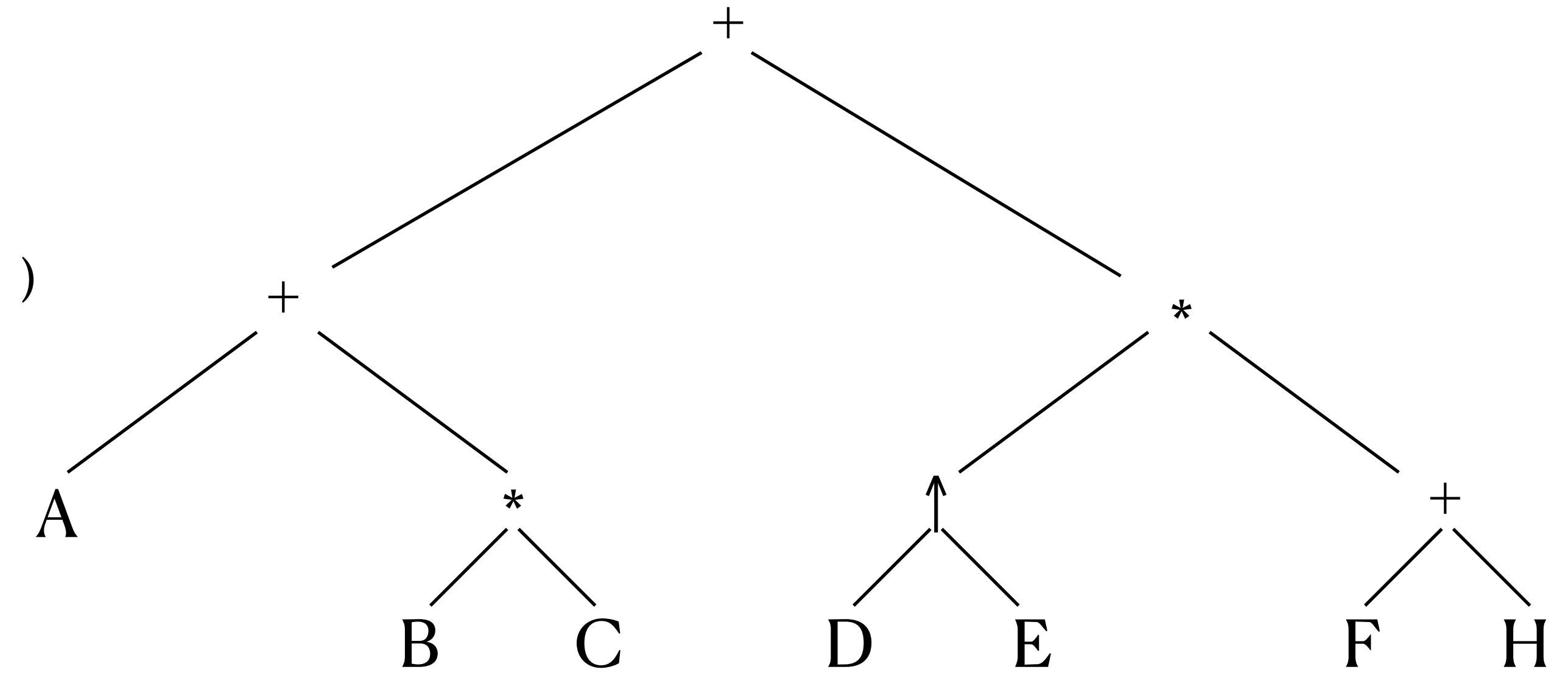
Inordine: (A + (B * C)) + ((D ↑ E) * (F + H))

Postordine: A B C * + D E ↑ F H + * +

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

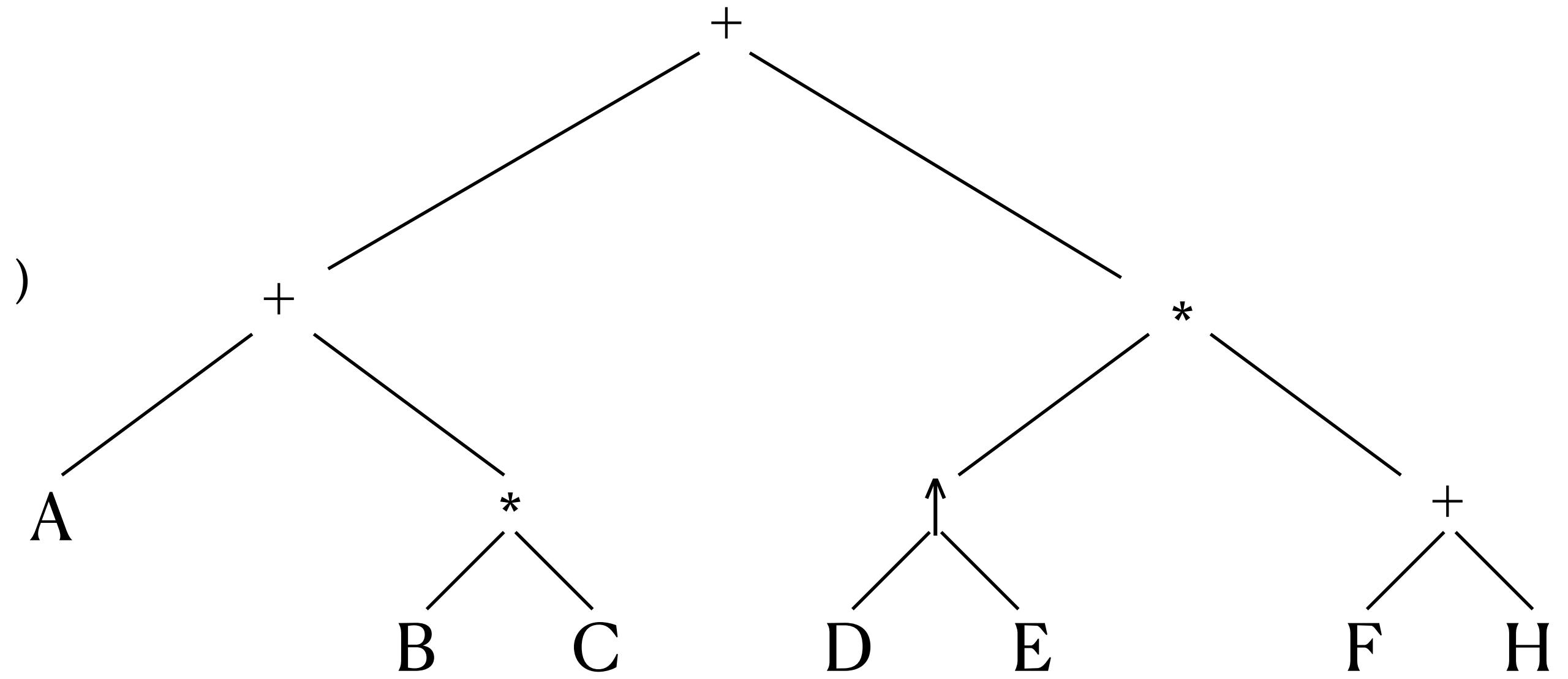
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

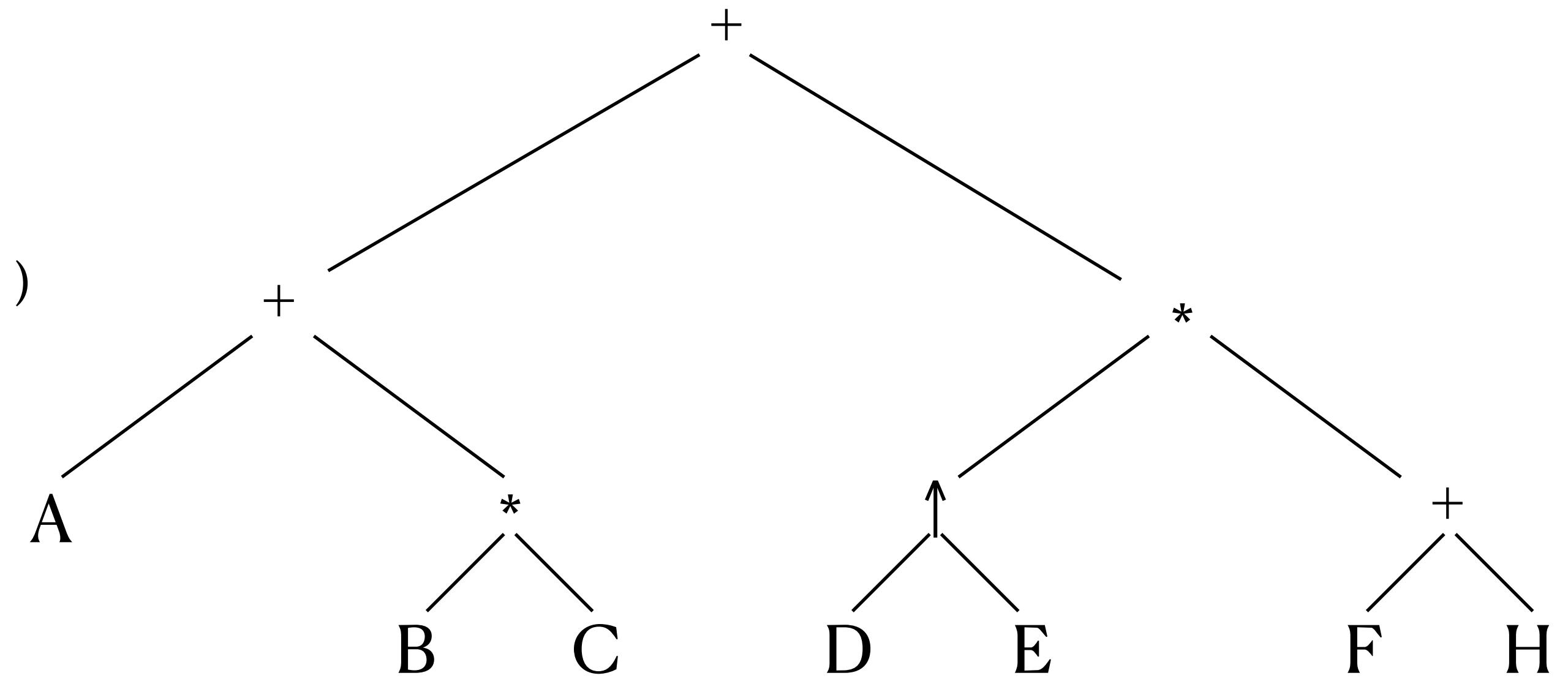
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

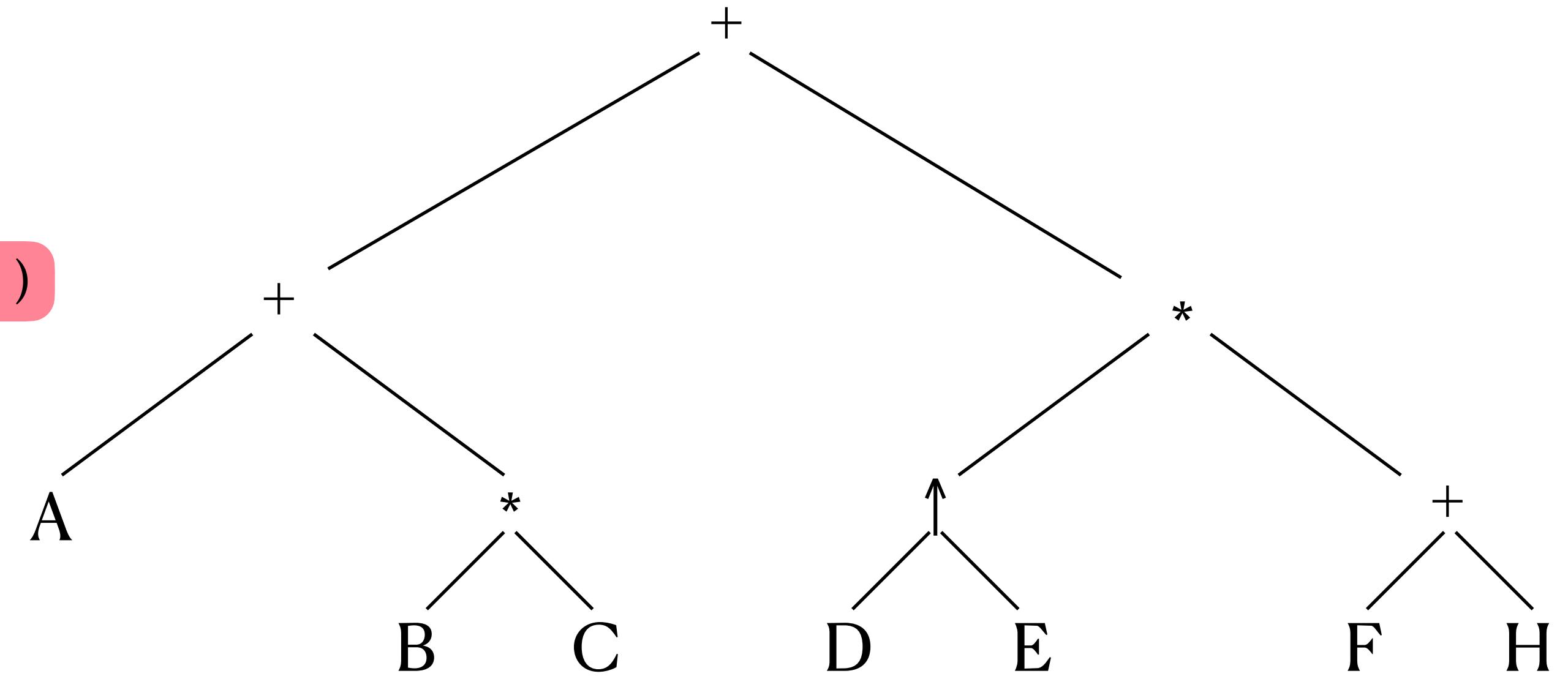
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

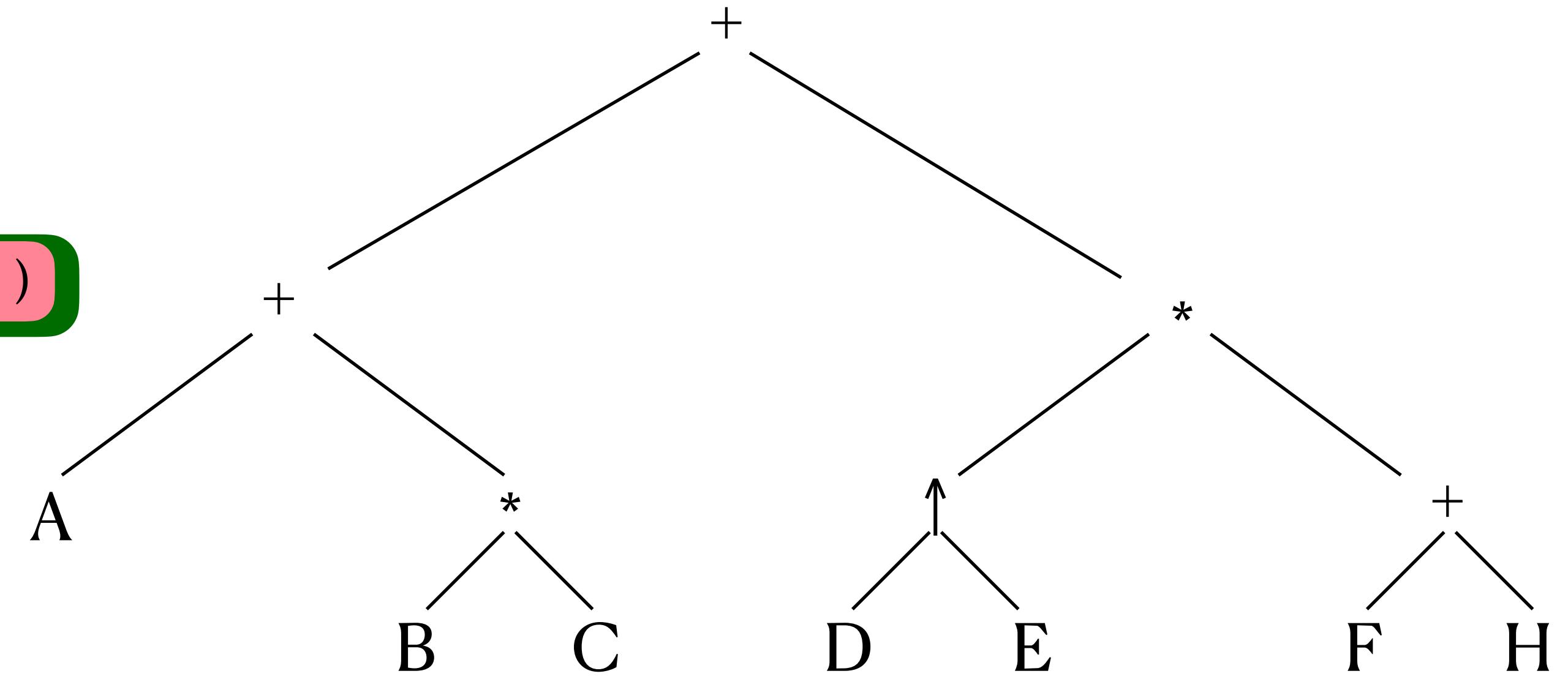
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

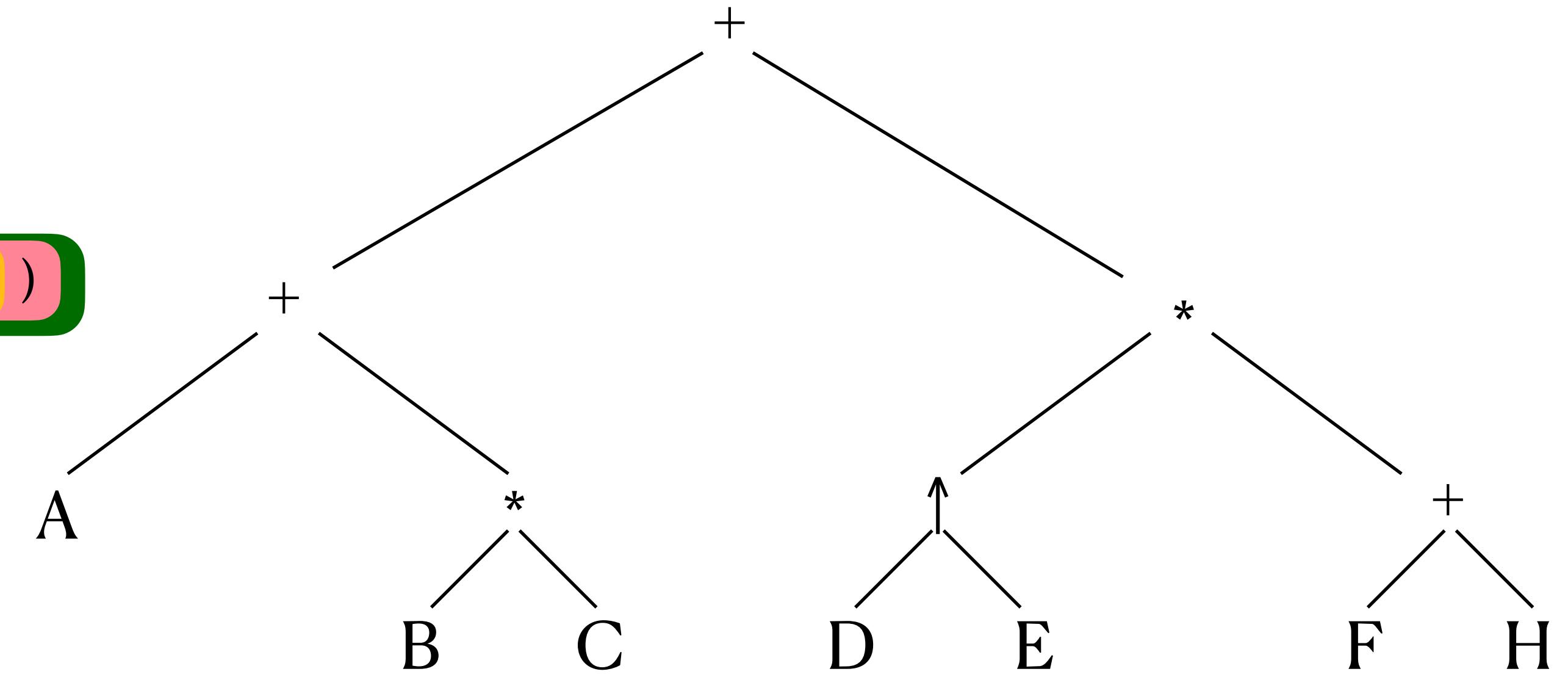
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

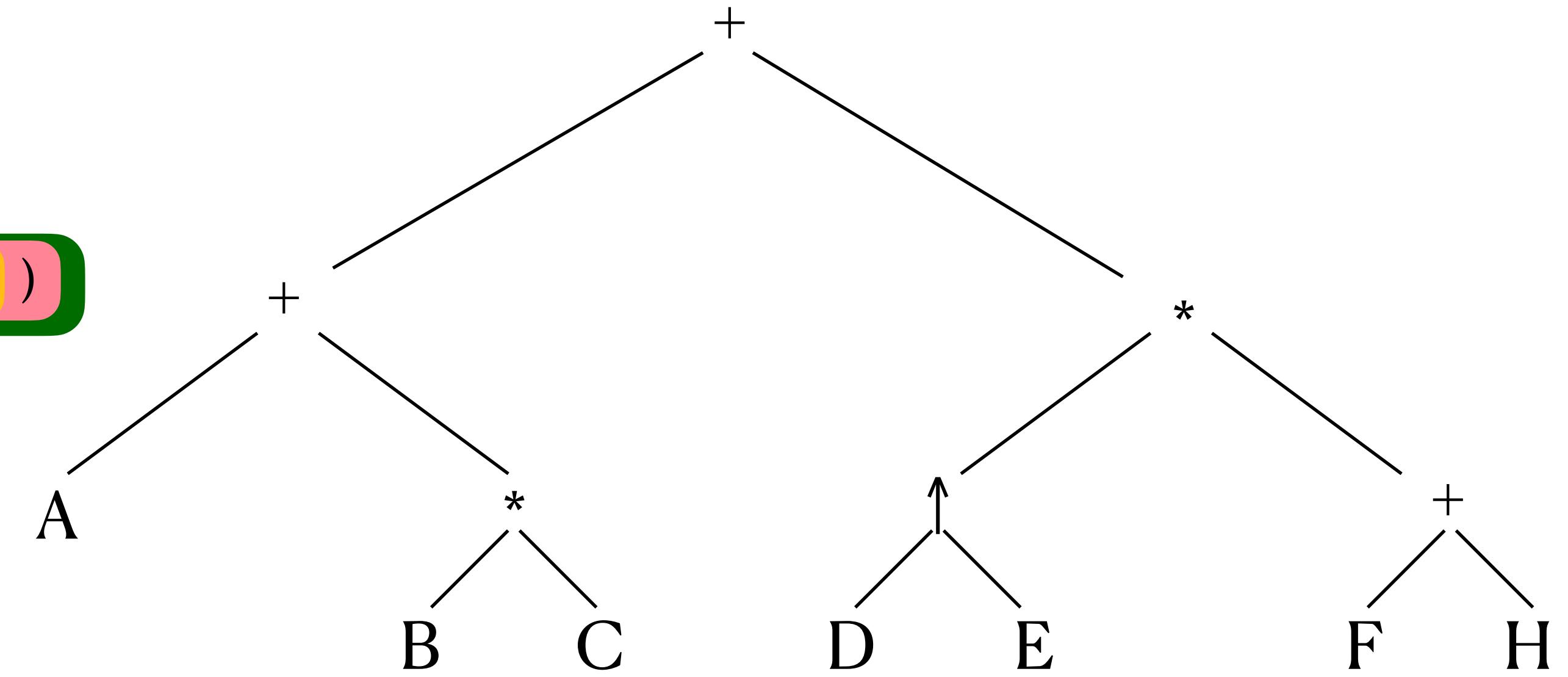
Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă

(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

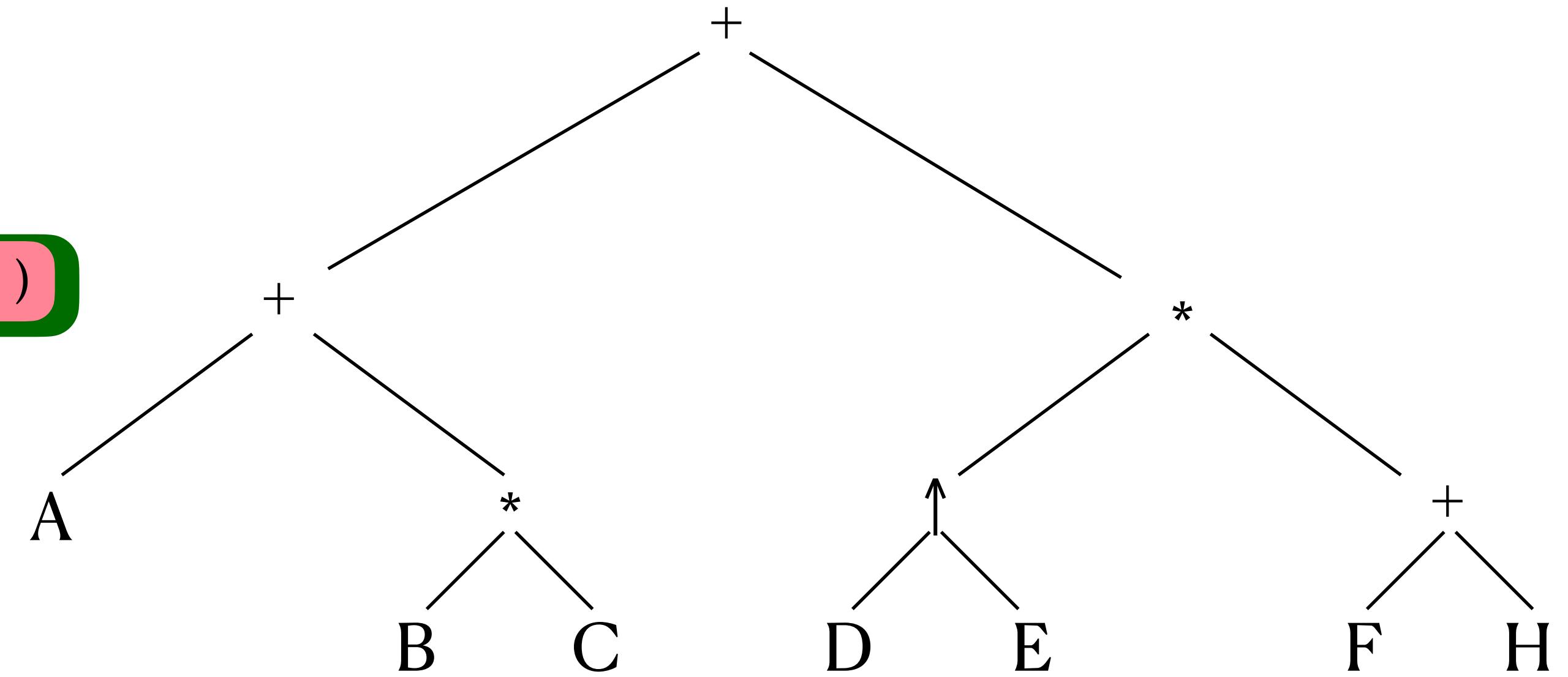
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

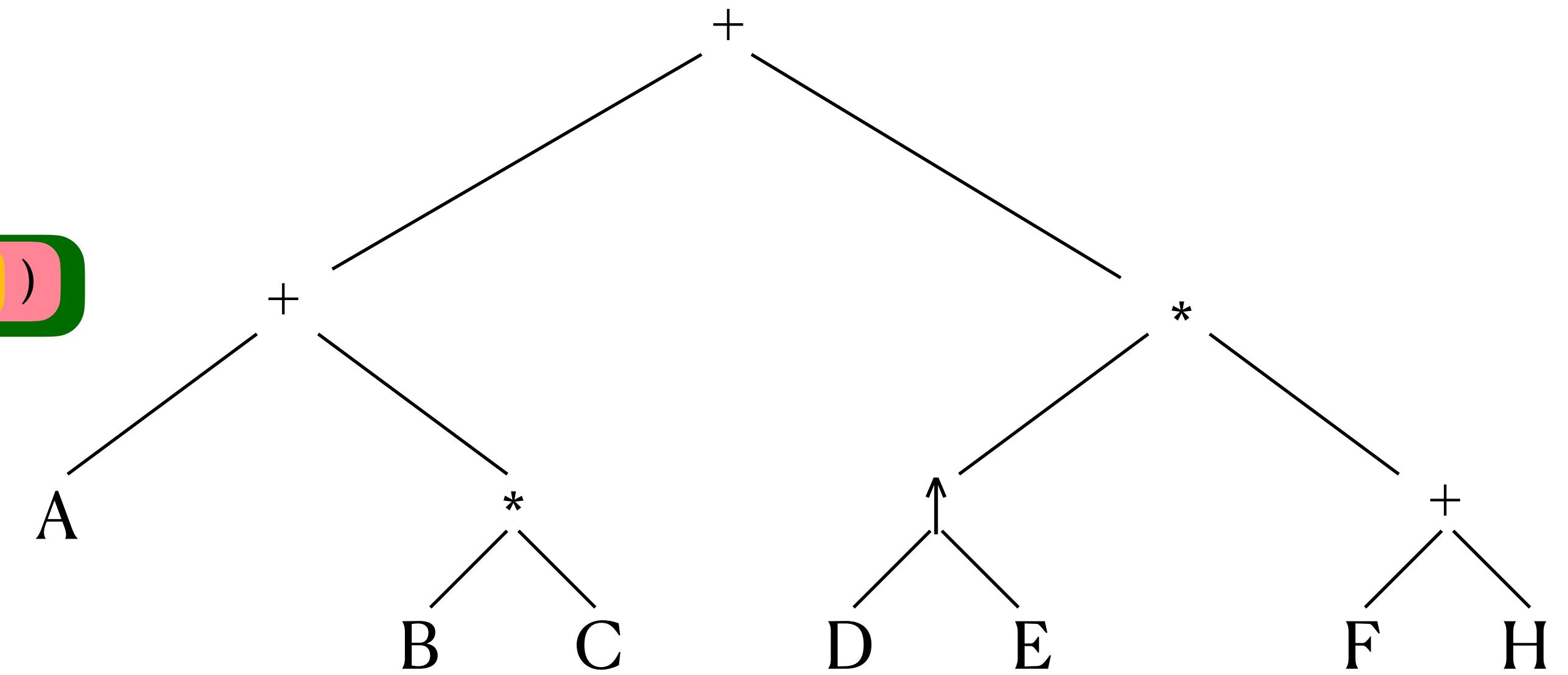
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

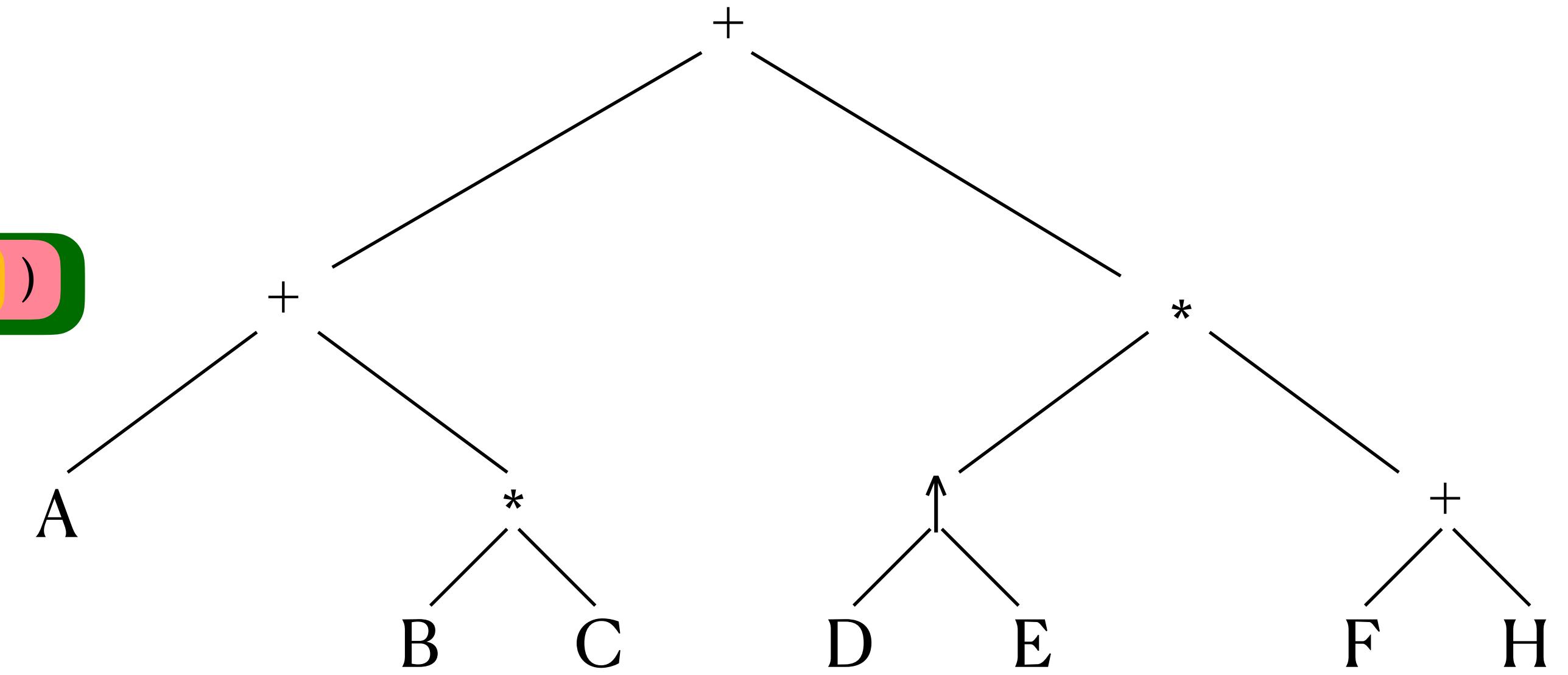
Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

Postordine: Forma Poloneză Inversă

(Reverse Polish Notation)



$A + B * C + D \uparrow E * (F + H)$

Preordine: $+ + A * B C * \uparrow D E + F H$

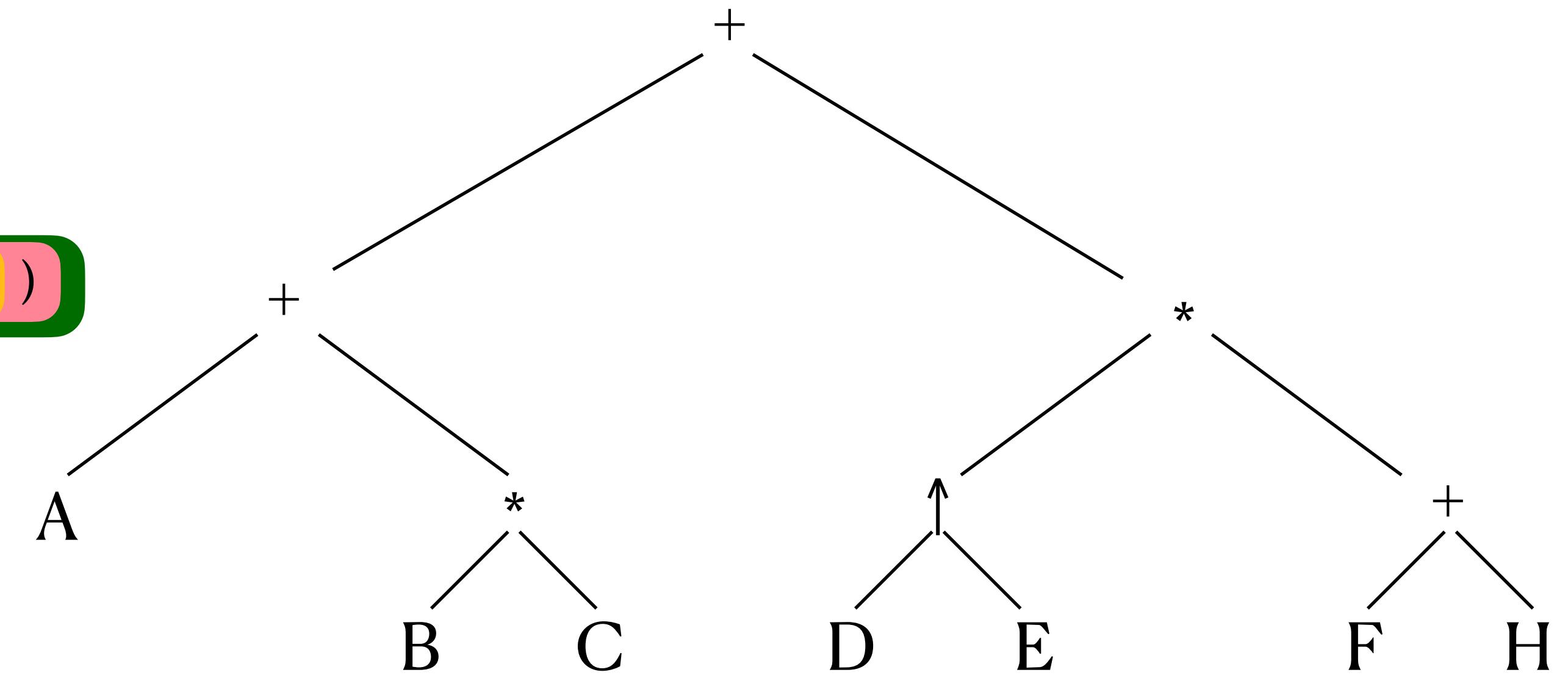
Inordine: $(A + (B * C)) + ((D \uparrow E) * (F + H))$

Postordine: $A B C * + D E \uparrow F H + * +$

Preordine: Forma Poloneză

Inordine: Forma Algebrică

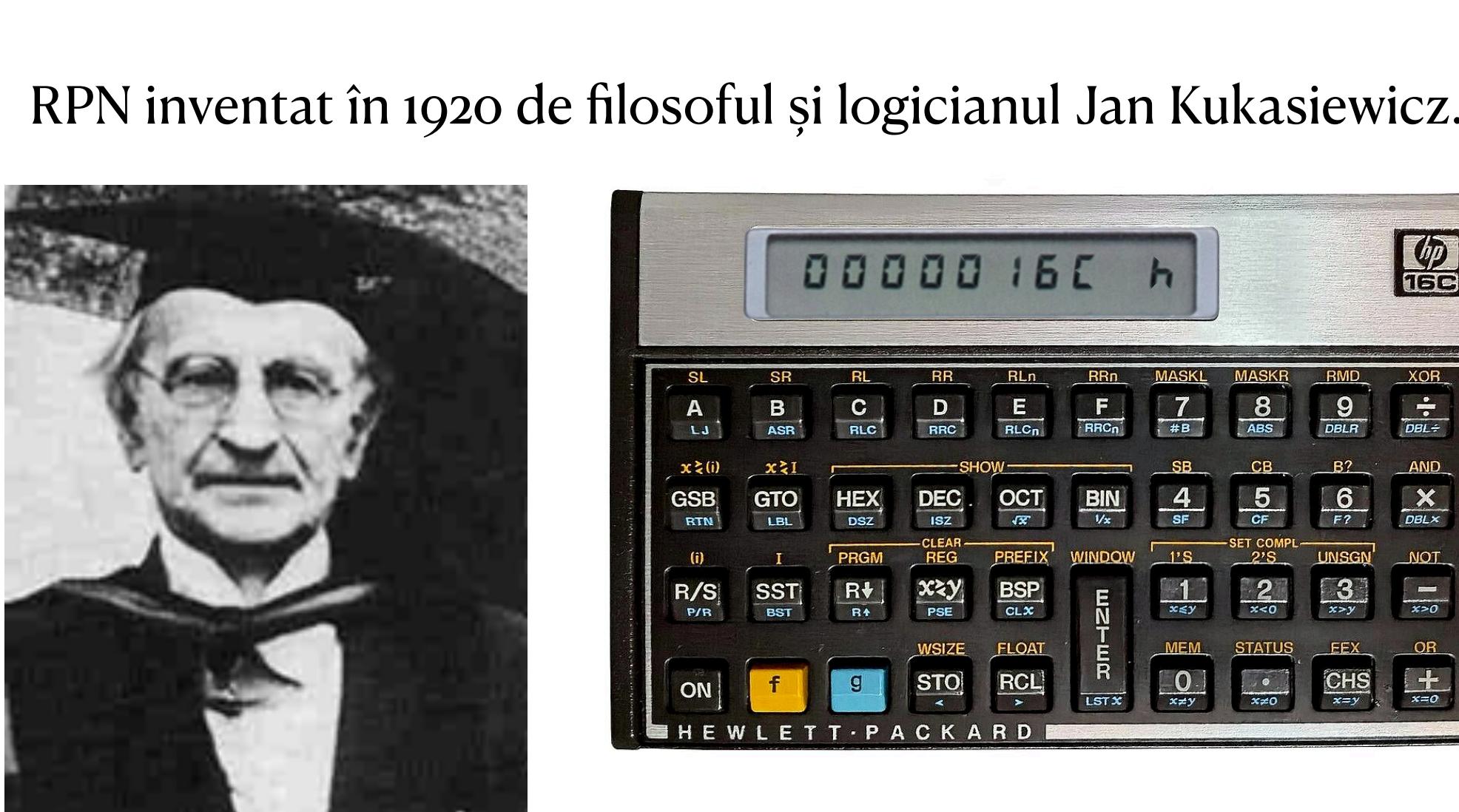
Postordine: Forma Poloneză Inversă
(Reverse Polish Notation)



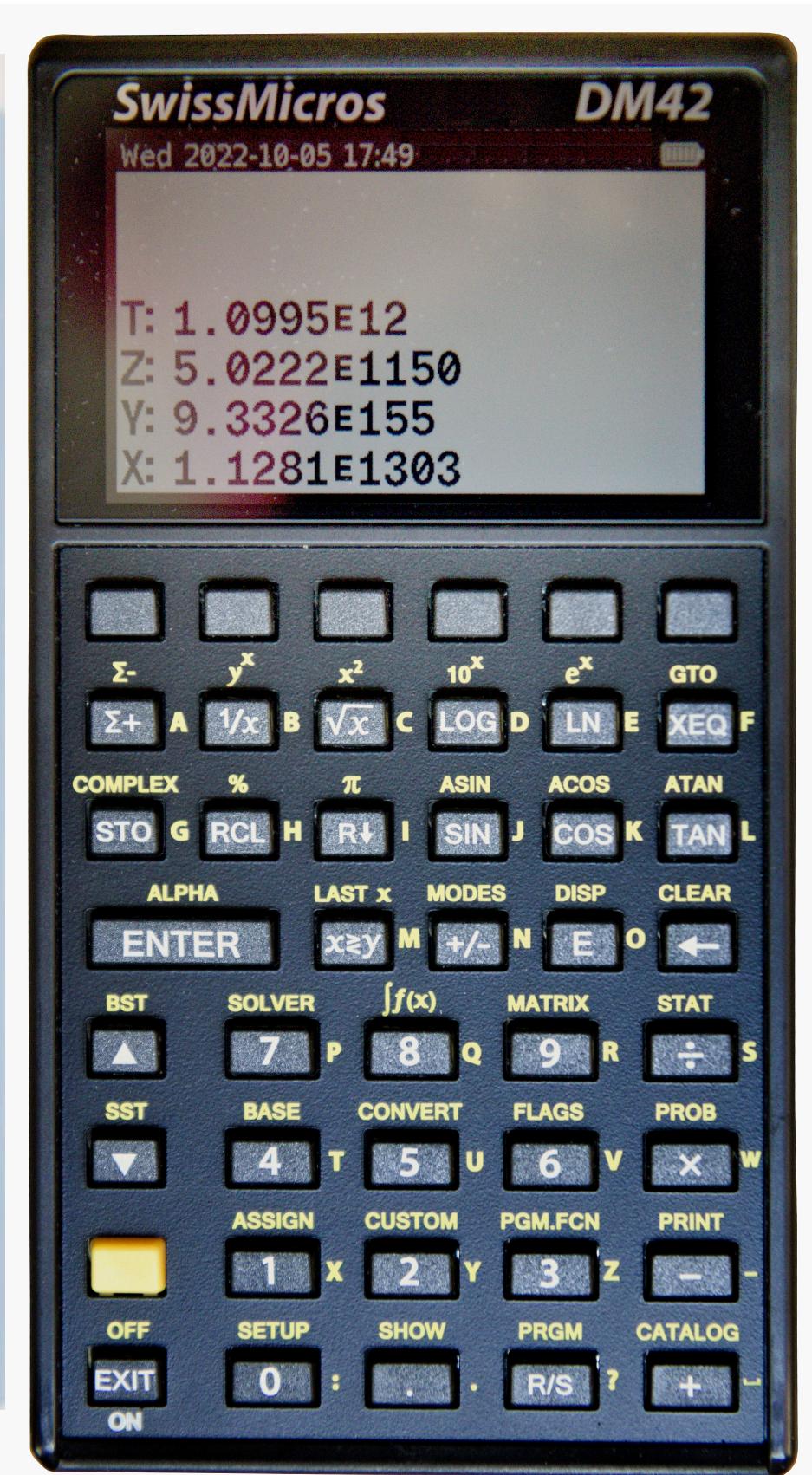
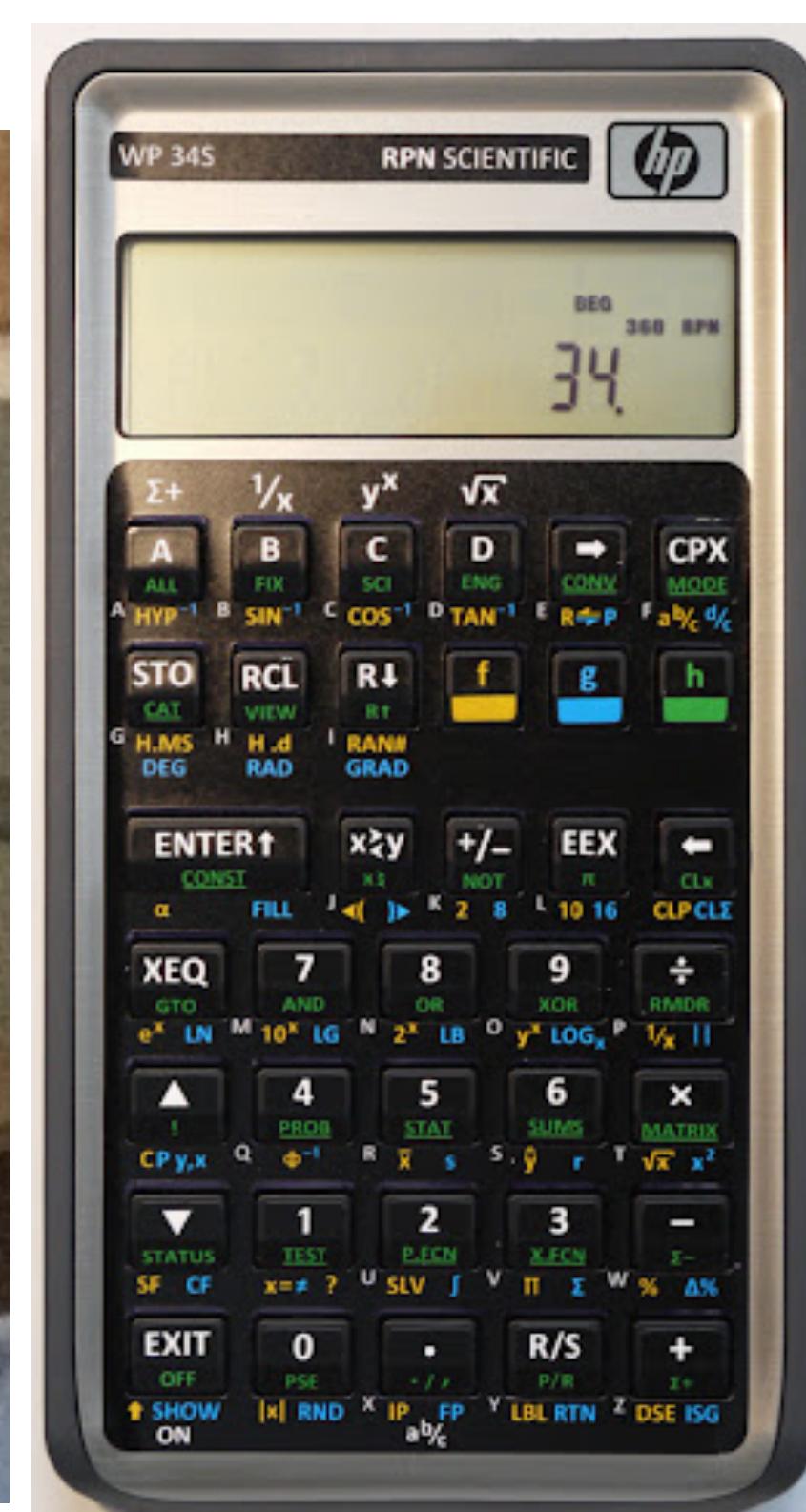
Forma Poloneză Inversă (RPN) este foarte utilă deoarece nu există ambiguitate privind ordinea operațiilor

În anii 70 - 80 multe calculatoare cu înaltă precizie (ingineri, economisti, arhitecti..) au fost create cu introducerea datelor RPN, sunt recunoscute prin absența tastei egal (=)

Chiar și în zilele noastre se produc asemenea calculatoare (SwissMicros)



RPN inventat în 1920 de filosoful și logicianul Jan Kukasiewicz.



HTML

```
<body>  
  <div>  
    <h1> heading here </h1>  
    <p> Lorem ipsum dolor sit amet</p>  
    <p> Lorem ipsum dolor </em> sit </em> amet</p>  
  </div>  
  <div>  
    <ul>  
      <li> item 1 </li>  
      <li> item 2 </li>  
      <li> item 3 </li>  
    </ul>  
  </div>  
</body>
```

HTML

```
<body>
  <div>
    <h1> heading here </h1>
    <p> Lorem ipsum dolor sit amet</p>
    <p> Lorem ipsum dolor </em> sit </em> amet</p>
  </div>
  <div>
    <ul>
      <li> item 1 </li>
      <li> item 2 </li>
      <li> item 3 </li>
    </ul>
  </div>
</body>
```

HTML

```
<body>
  <div>
    <h1> heading here </h1>
    <p> Lorem ipsum dolor sit amet</p>
    <p> Lorem ipsum dolor </em> sit </em> amet</p>
  </div>
  <div>
    <ul>
      <li> item 1 </li>
      <li> item 2 </li>
      <li> item 3 </li>
    </ul>
  </div>
</body>
```

