

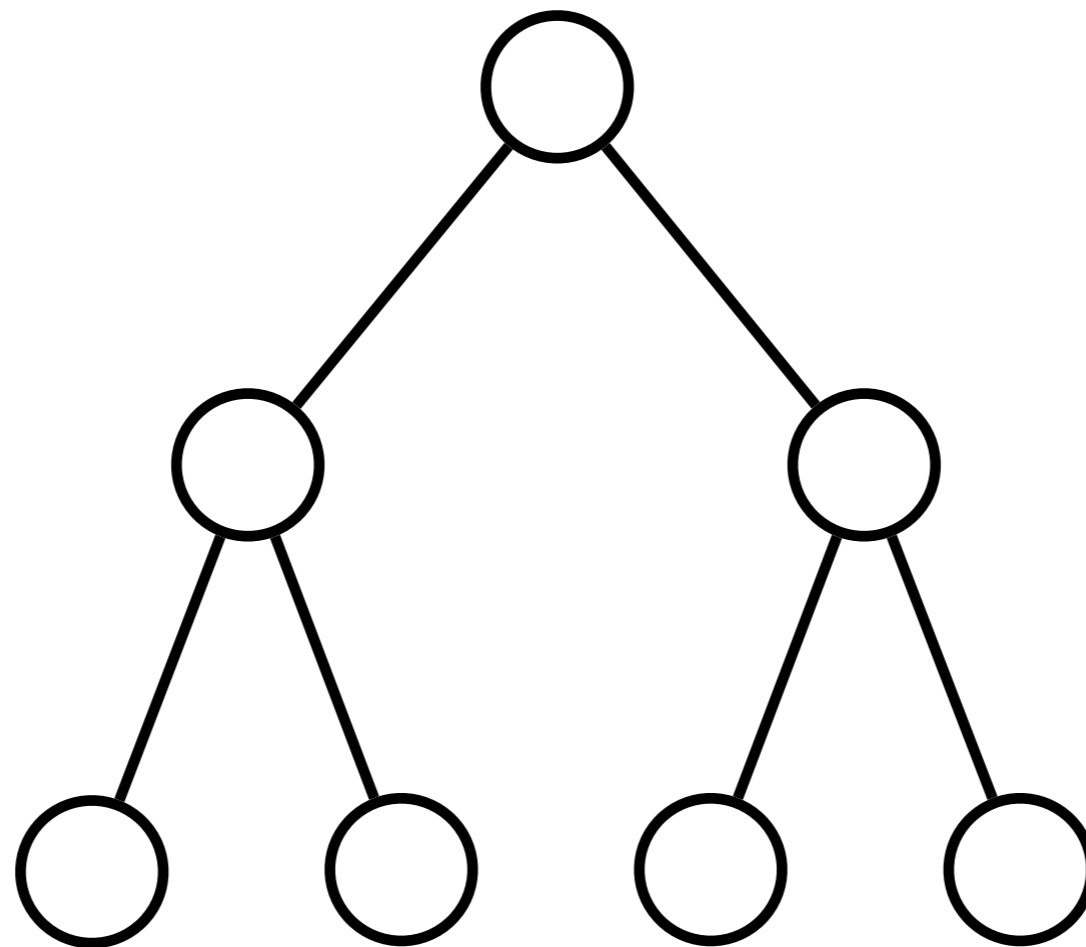
# Introduction à l'informatique CM10

Antonio E. Porreca  
[aeporreca.org/introinfo](http://aeporreca.org/introinfo)

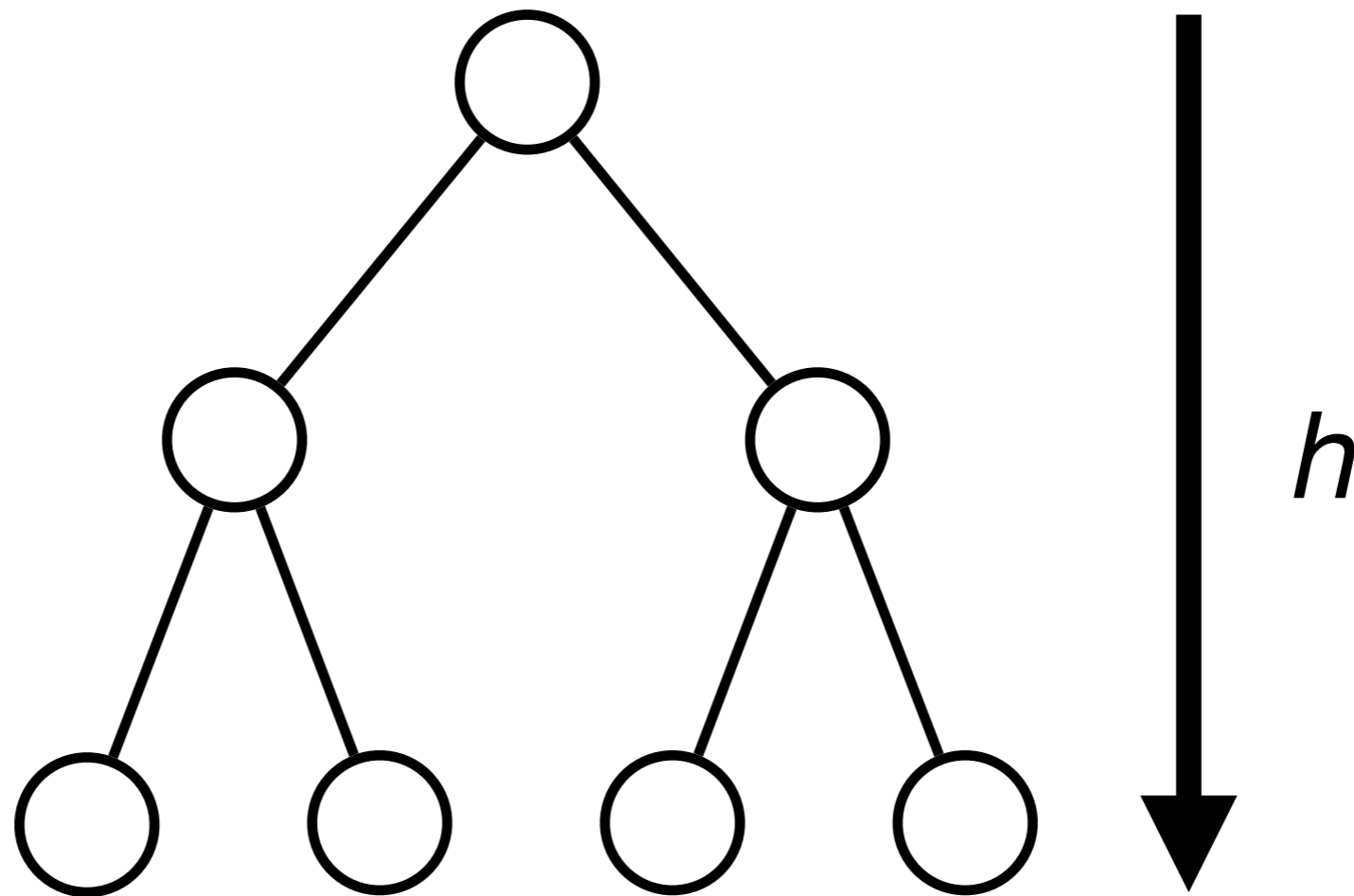
# Recherche dans un ABR

```
def rechercher_abr(arbre, x):  
    if est_vide(arbre):  
        return False  
    elif x == valeur(arbre):  
        return True  
    elif x < valeur(arbre):  
        return rechercher_abr(gauche(arbre), x)  
    else:  
        return rechercher_abr(droite(arbre), x)
```

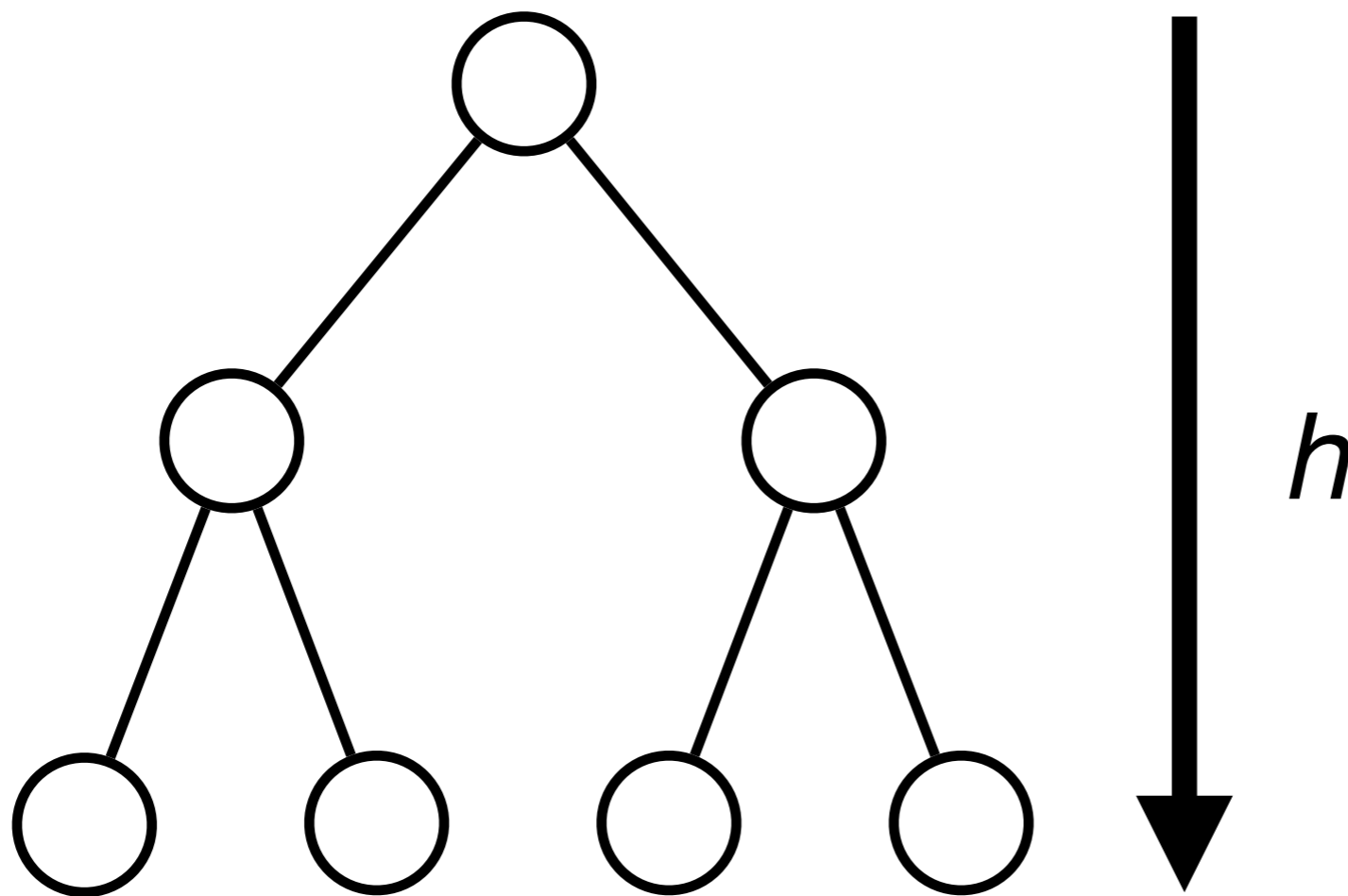
# Temps de recherche



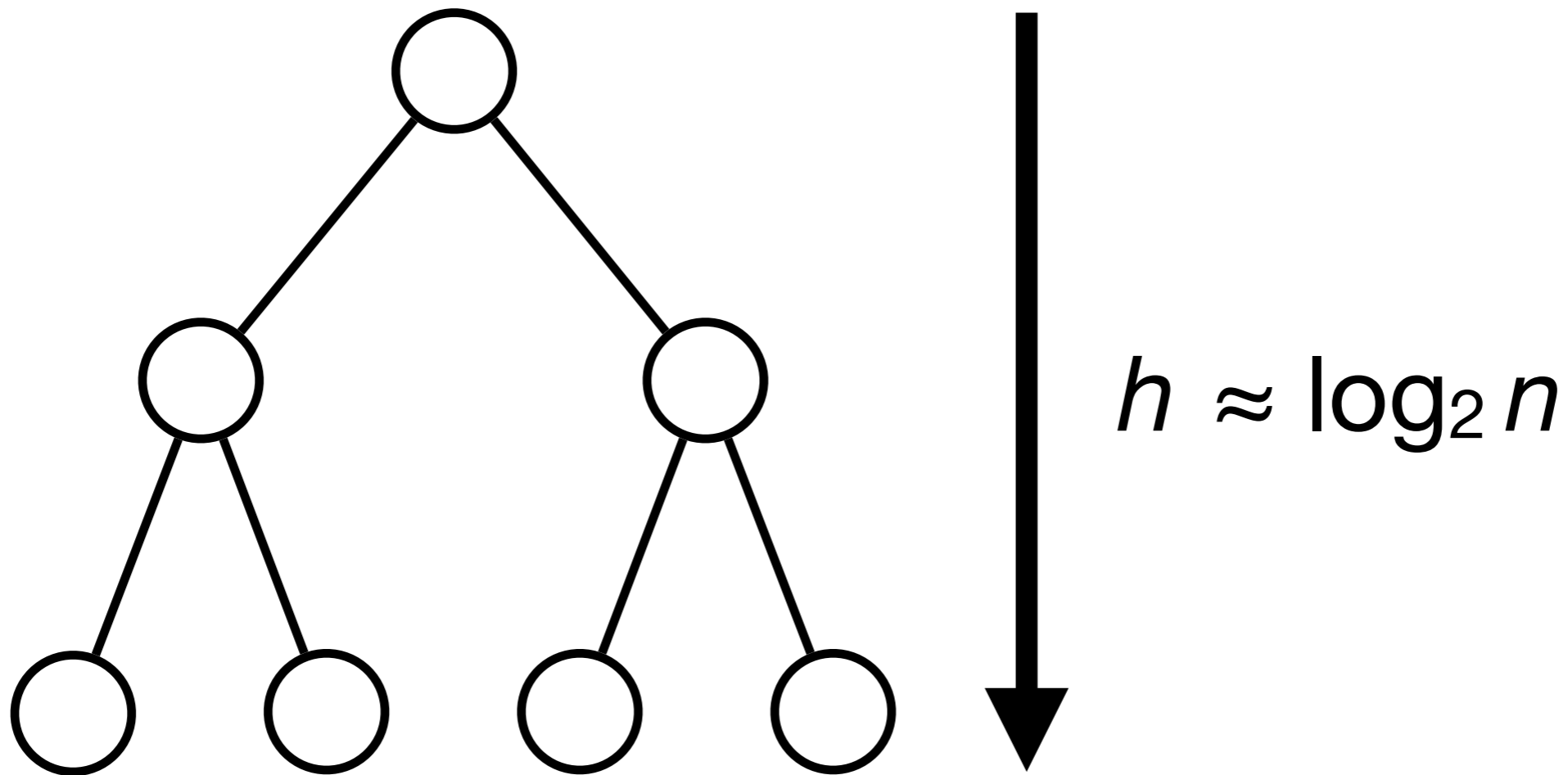
# Temps de recherche



# Pour un ABR équilibré



# Pour un ABR équilibré



# Insertion dans un ABR

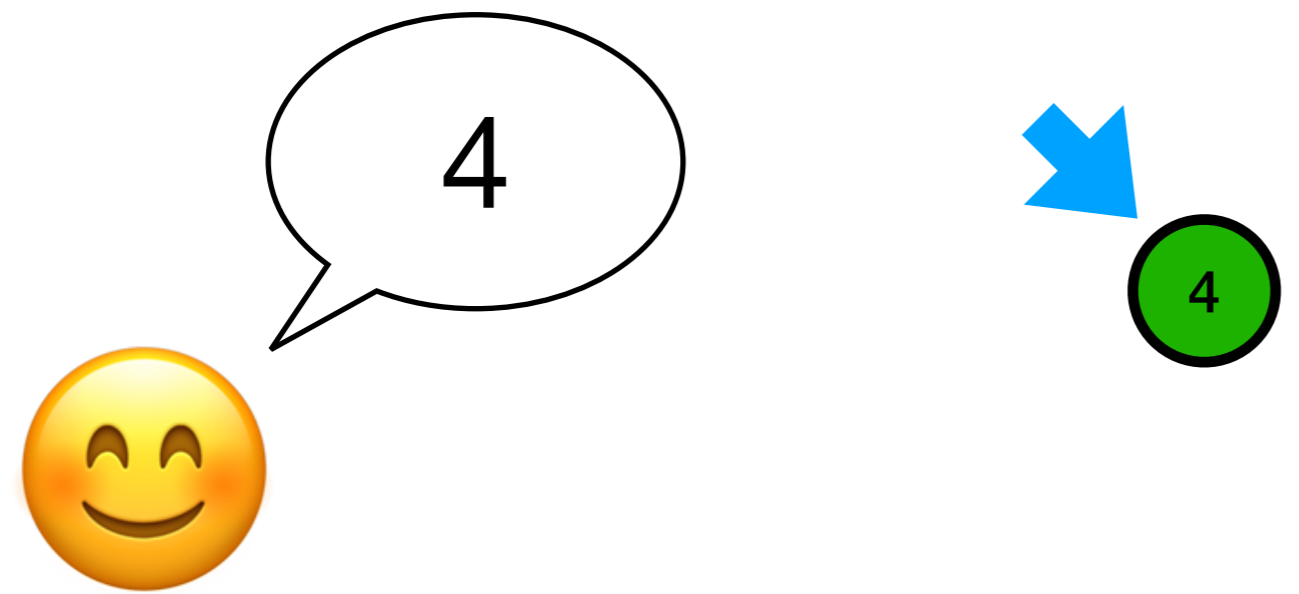
nil

# Insertion dans un ABR

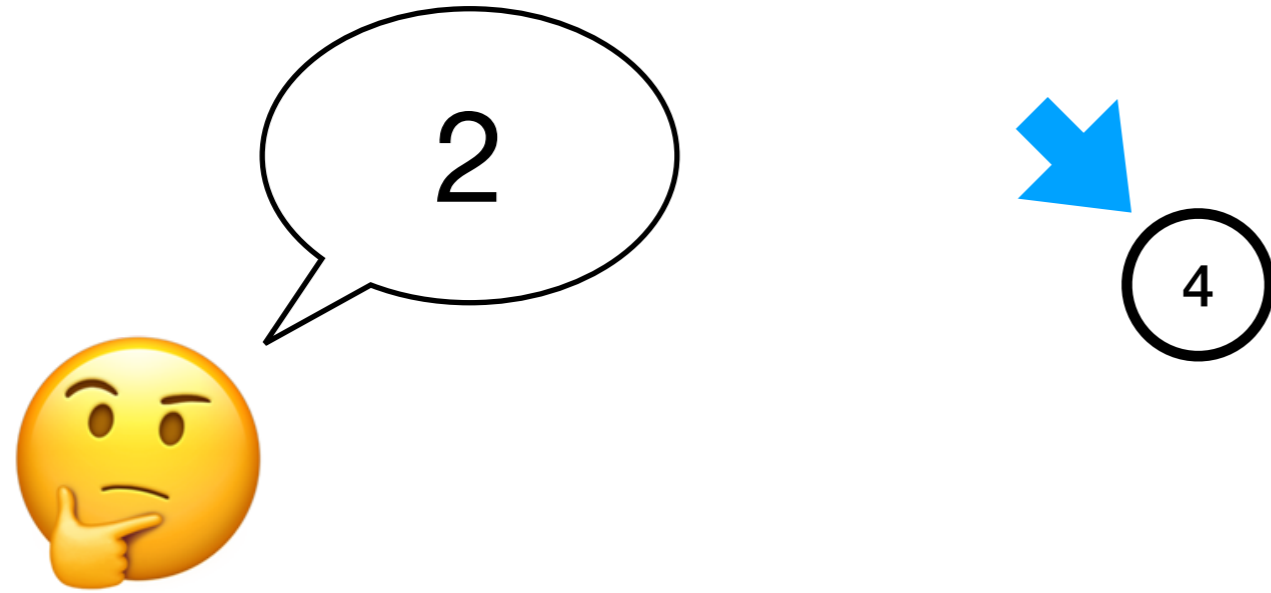




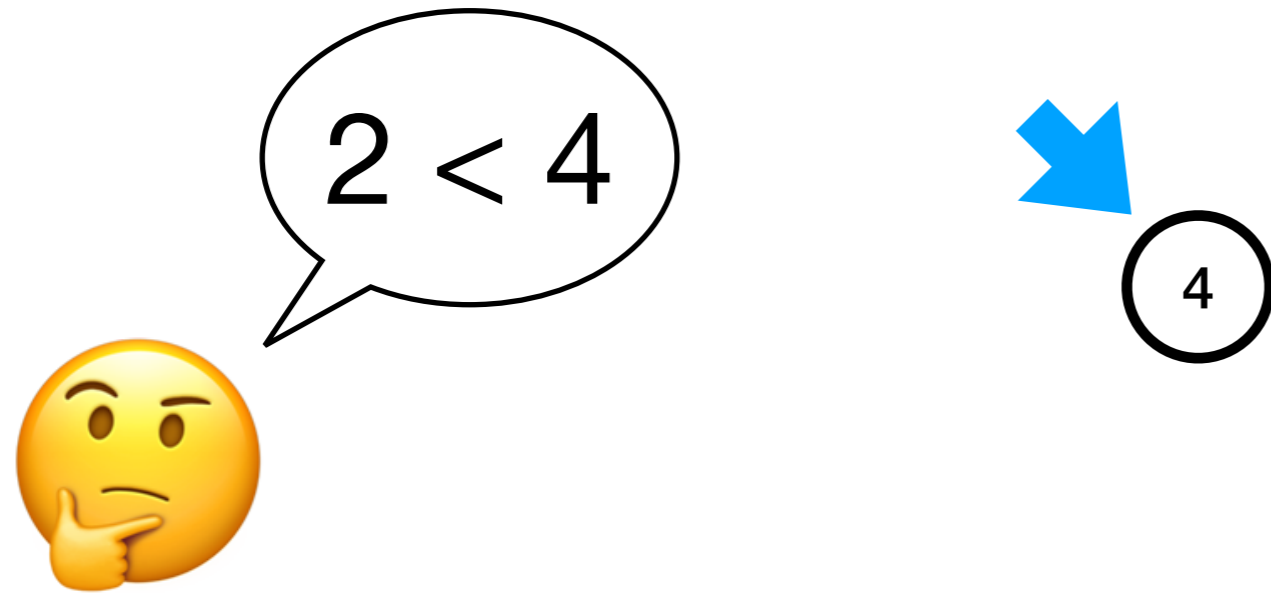
# Insertion dans un ABR



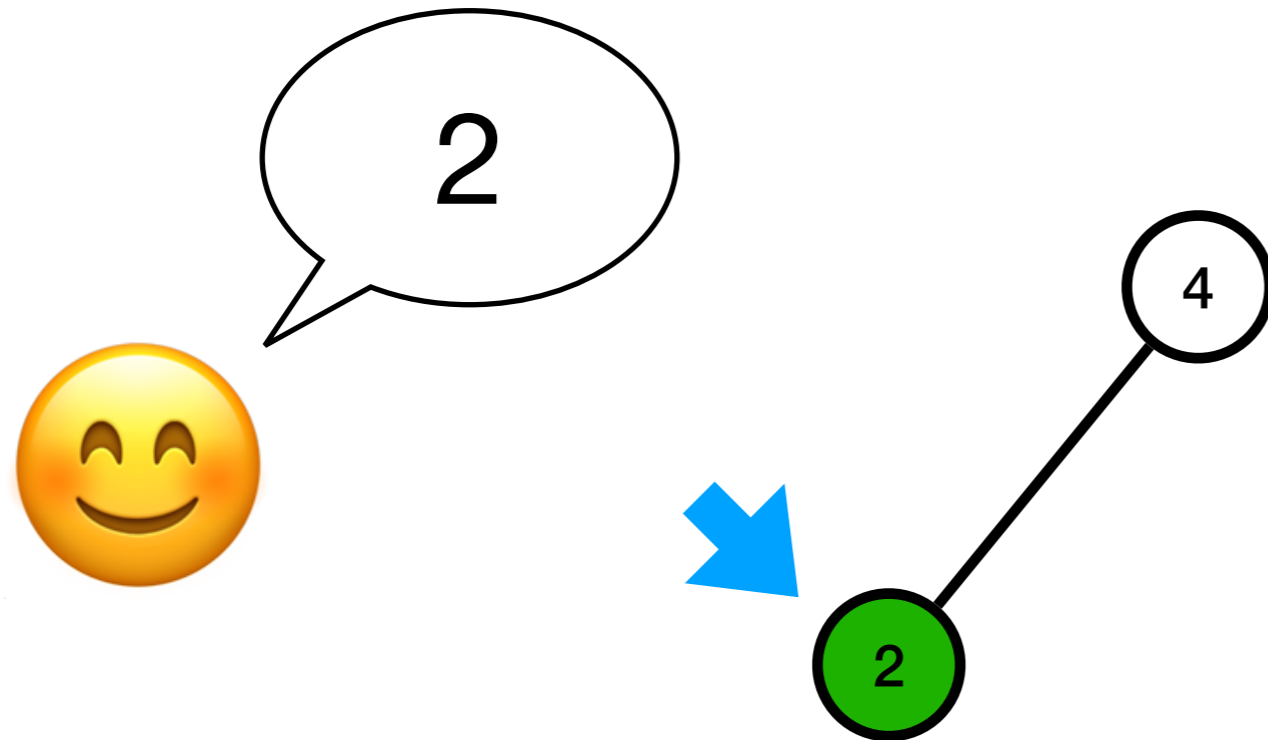
# Insertion dans un ABR



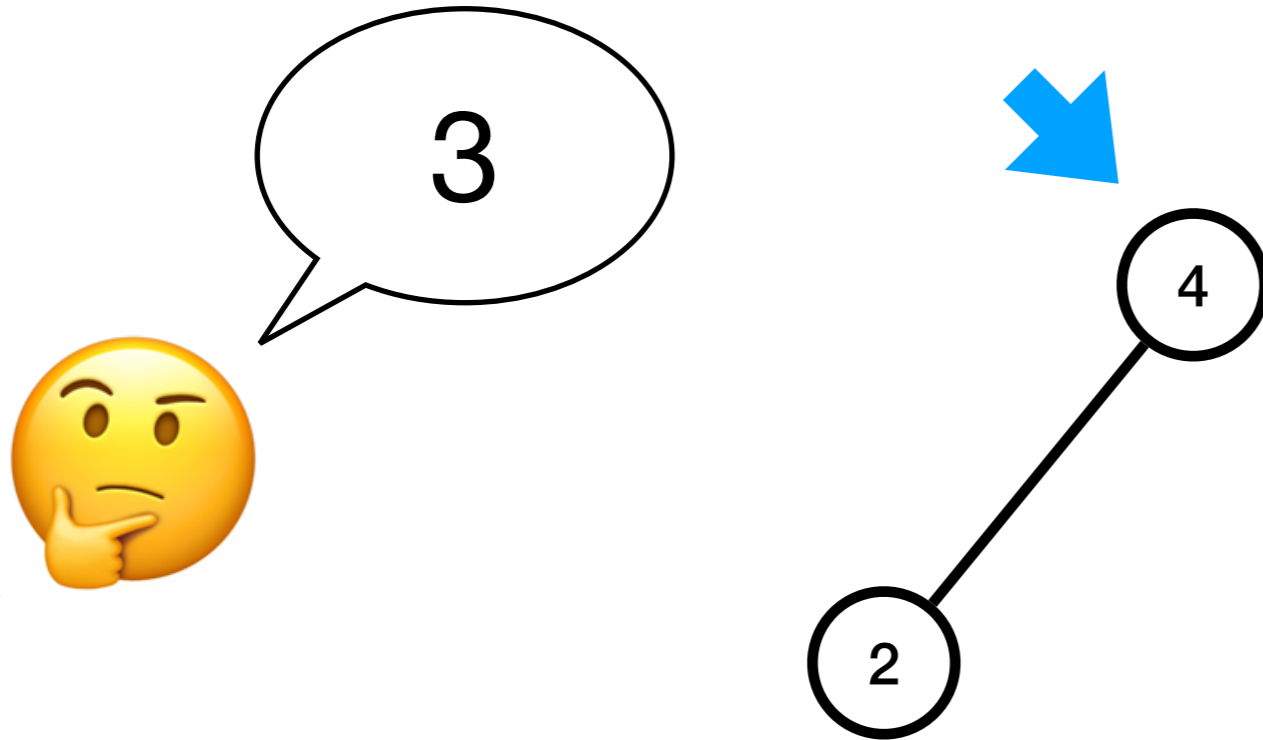
# Insertion dans un ABR



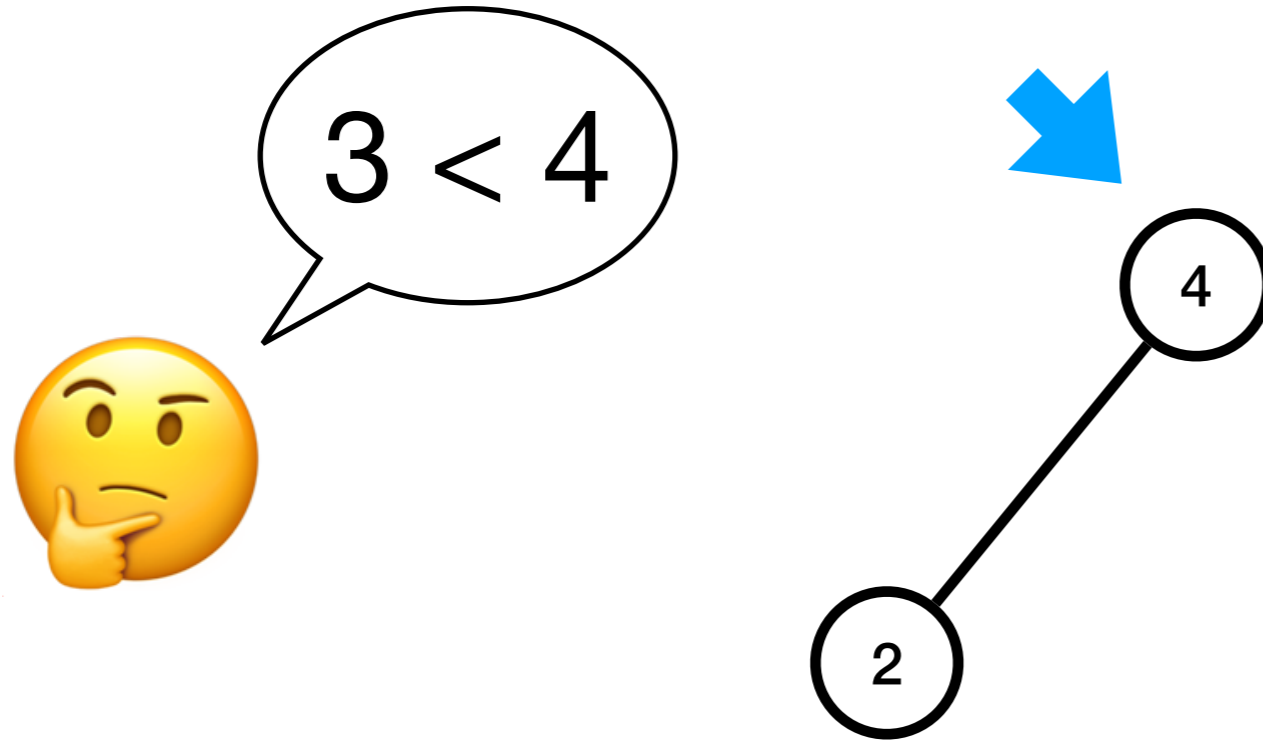
# Insertion dans un ABR



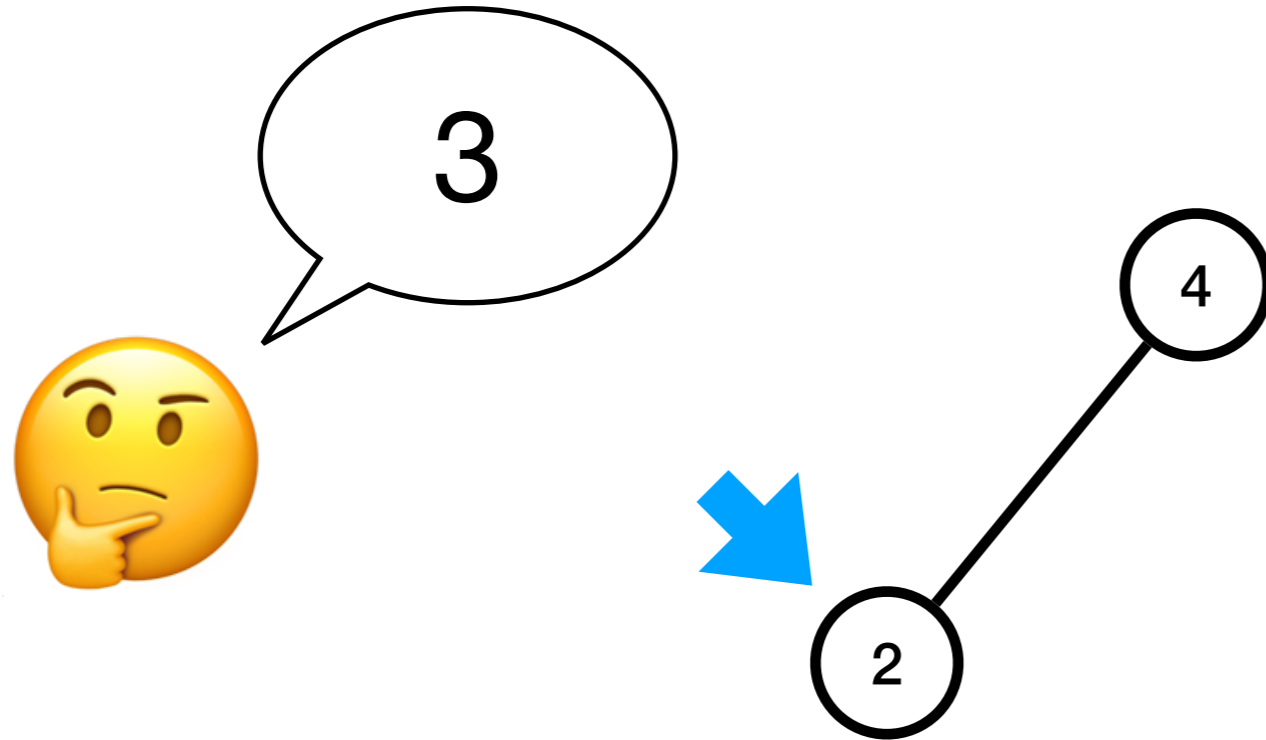
# Insertion dans un ABR



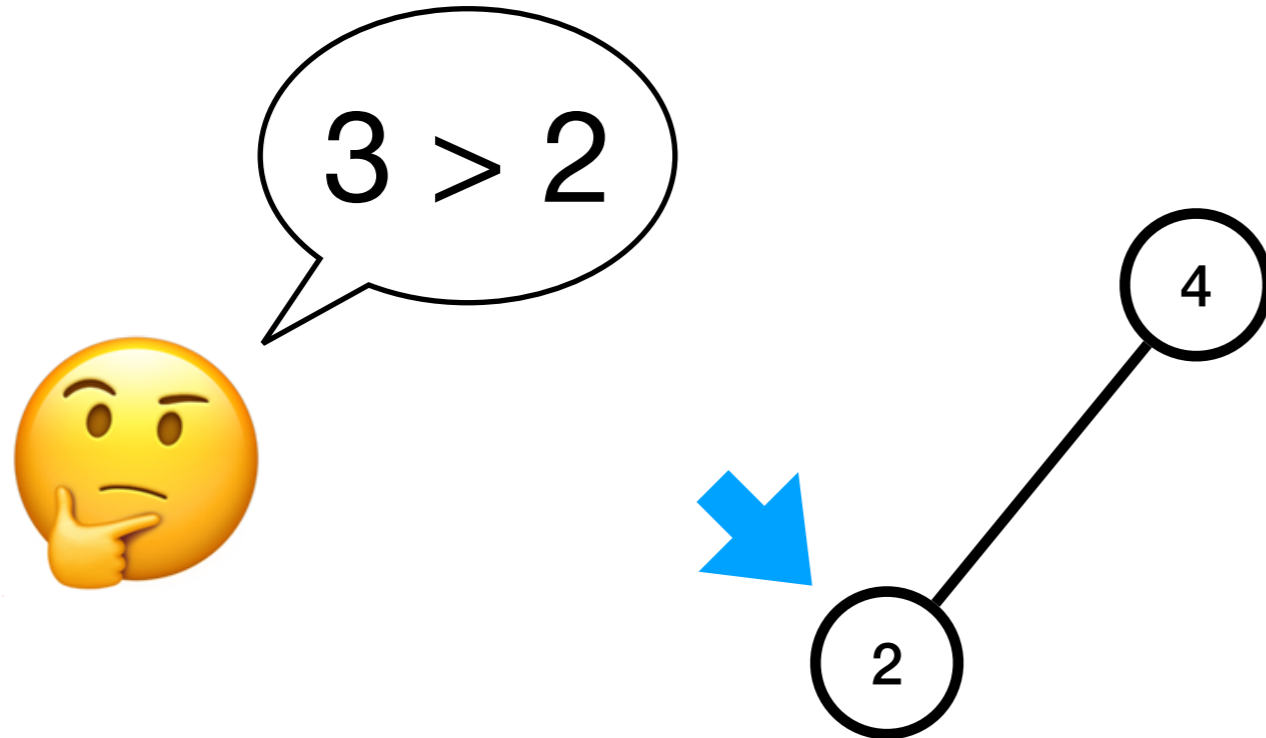
# Insertion dans un ABR



# Insertion dans un ABR

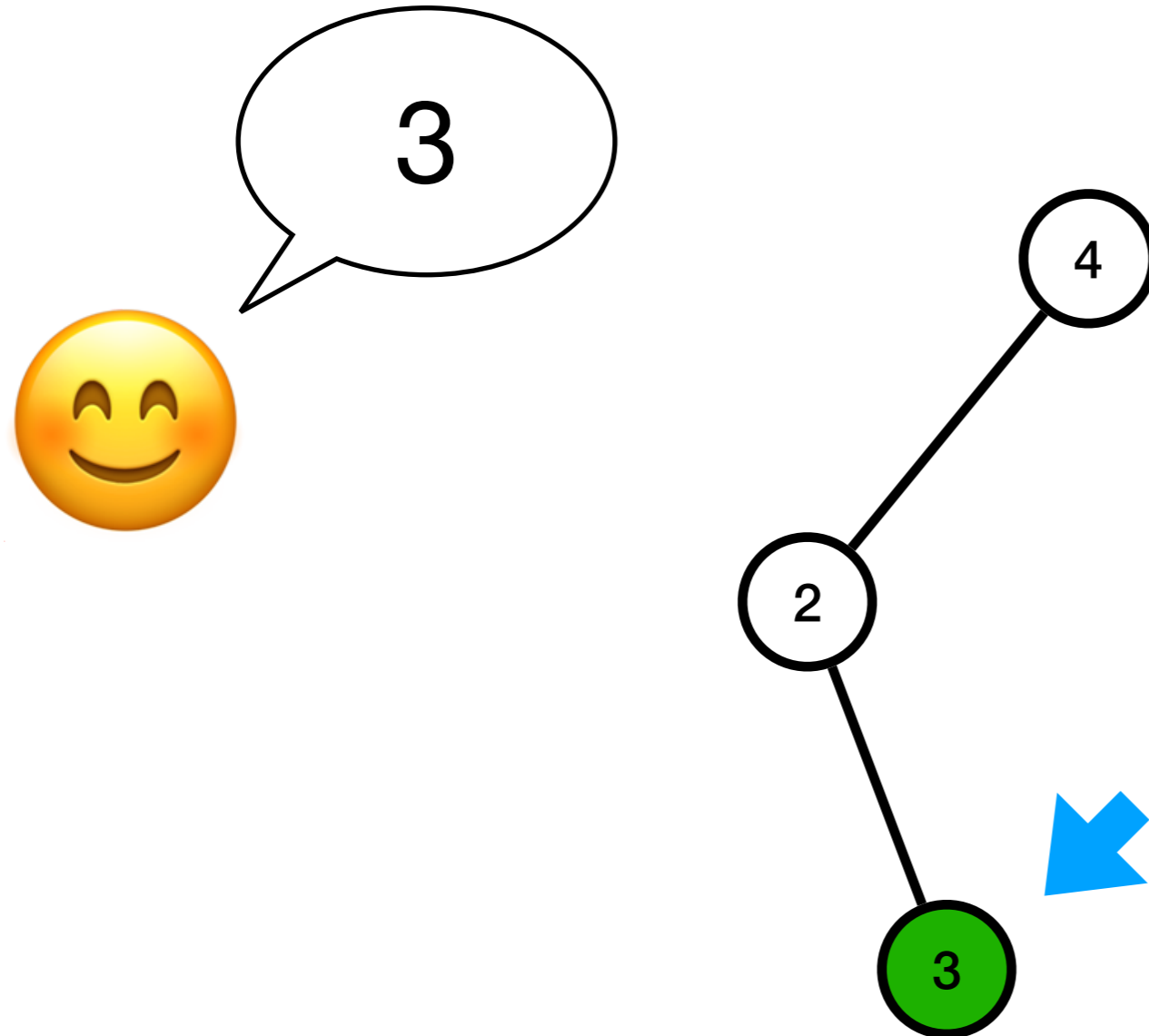


# Insertion dans un ABR

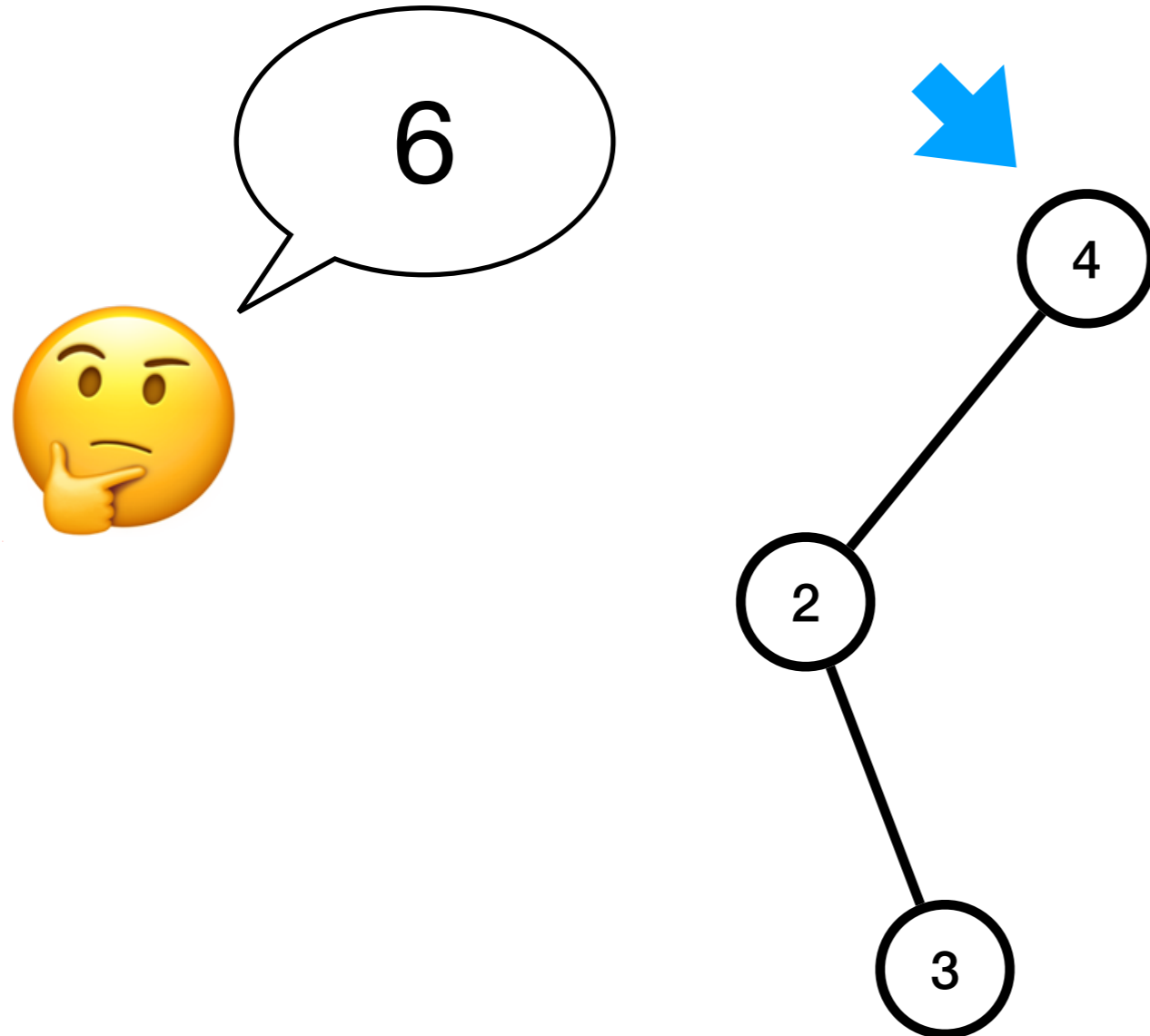




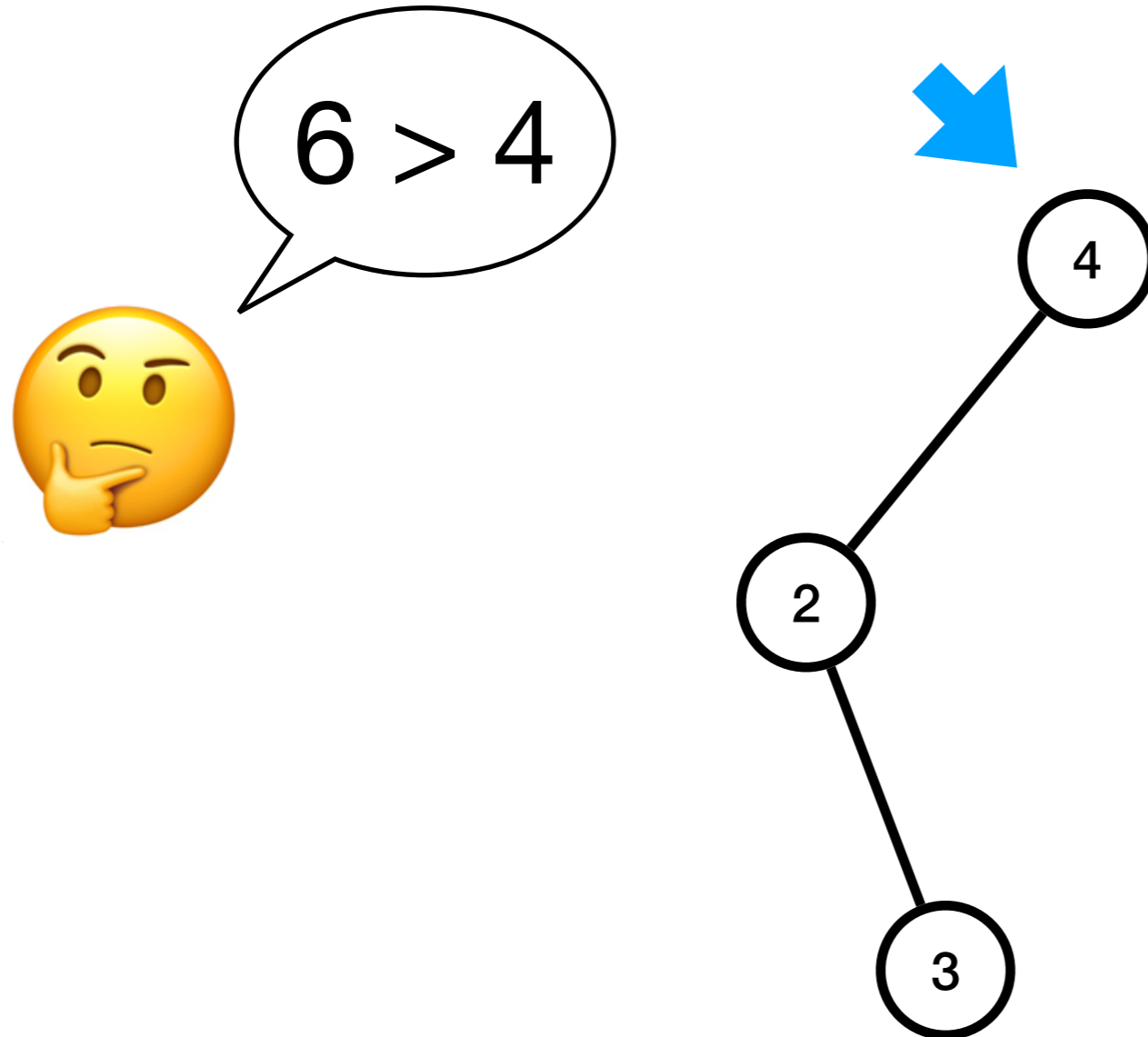
# Insertion dans un ABR



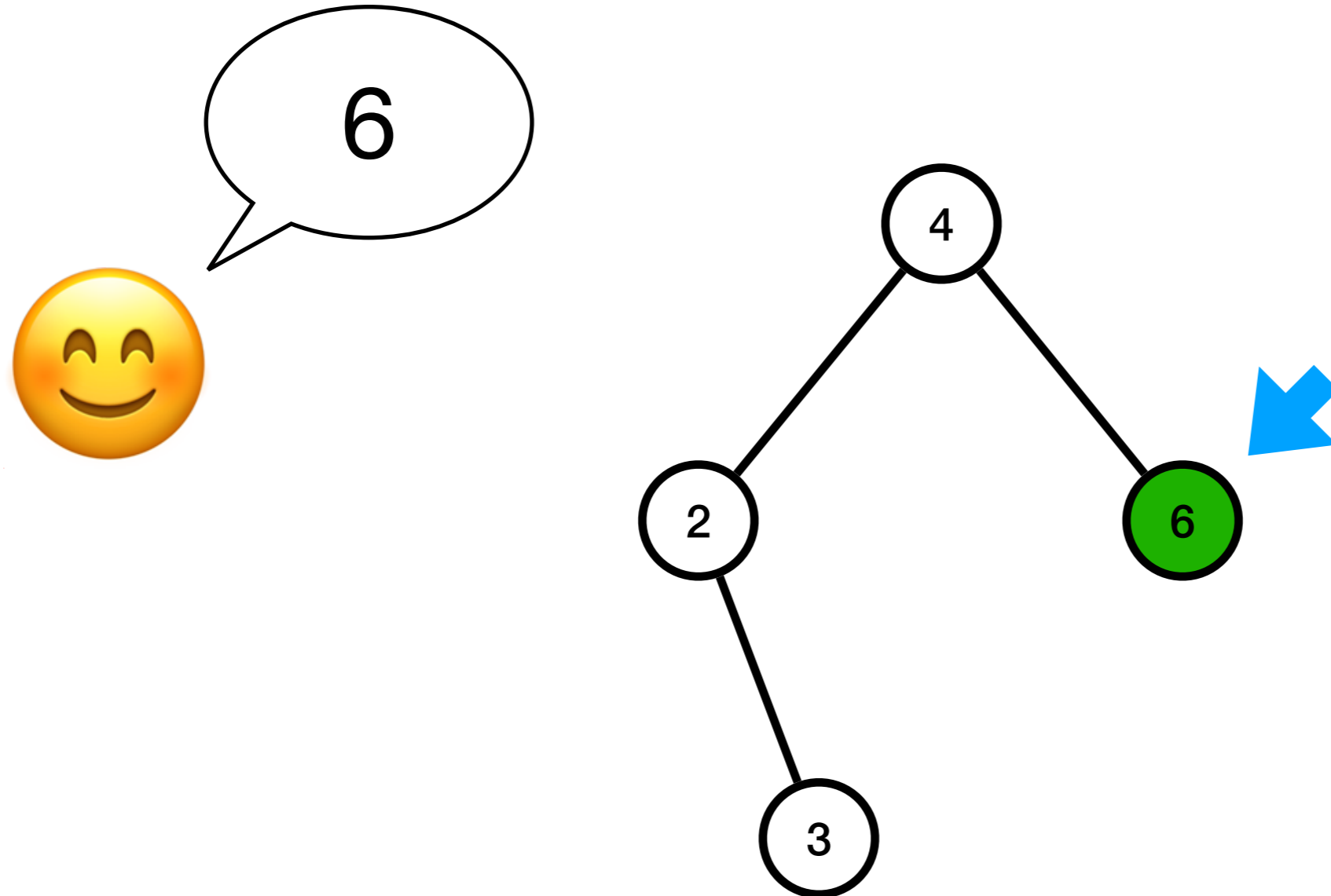
# Insertion dans un ABR



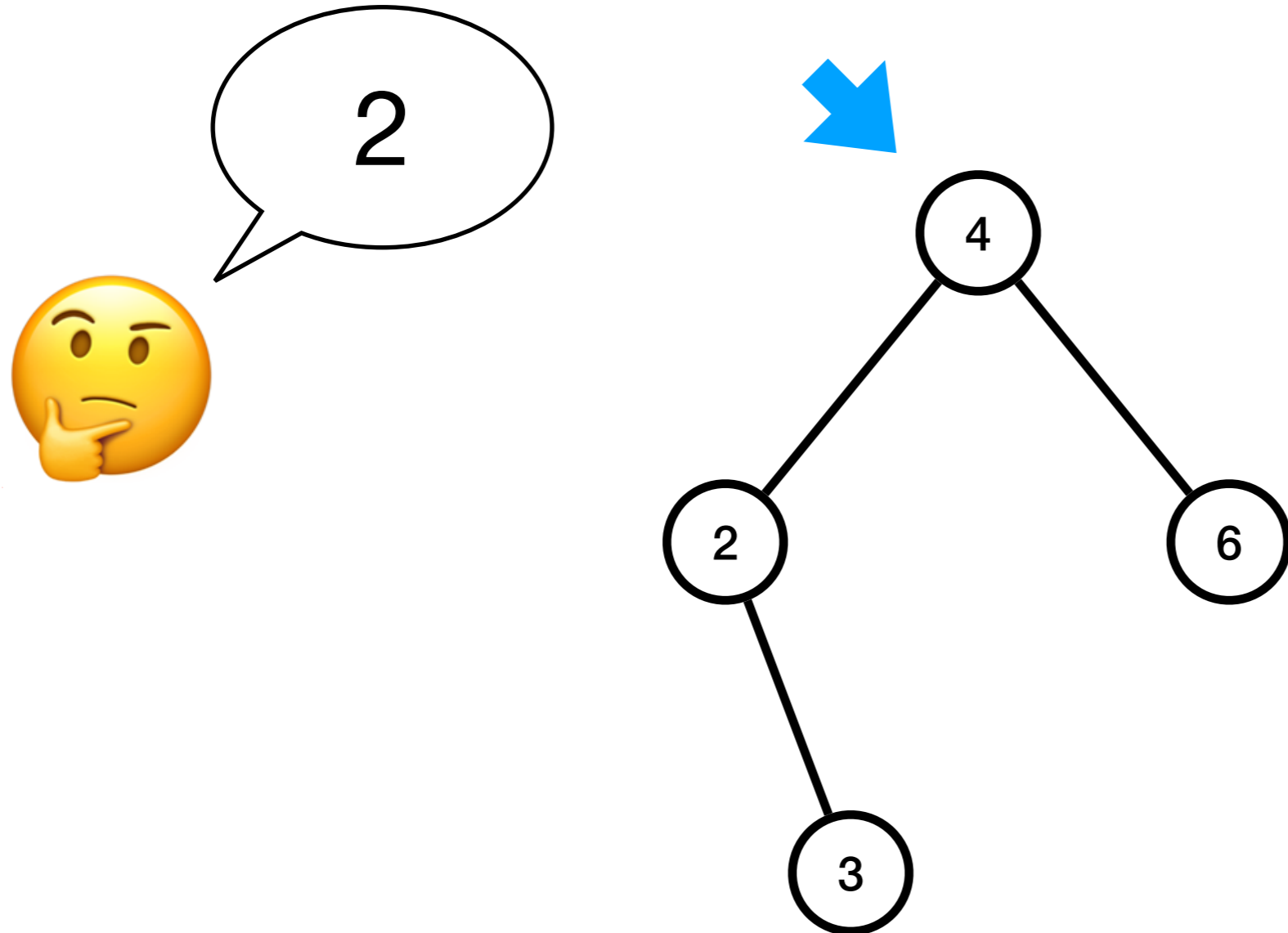
# Insertion dans un ABR



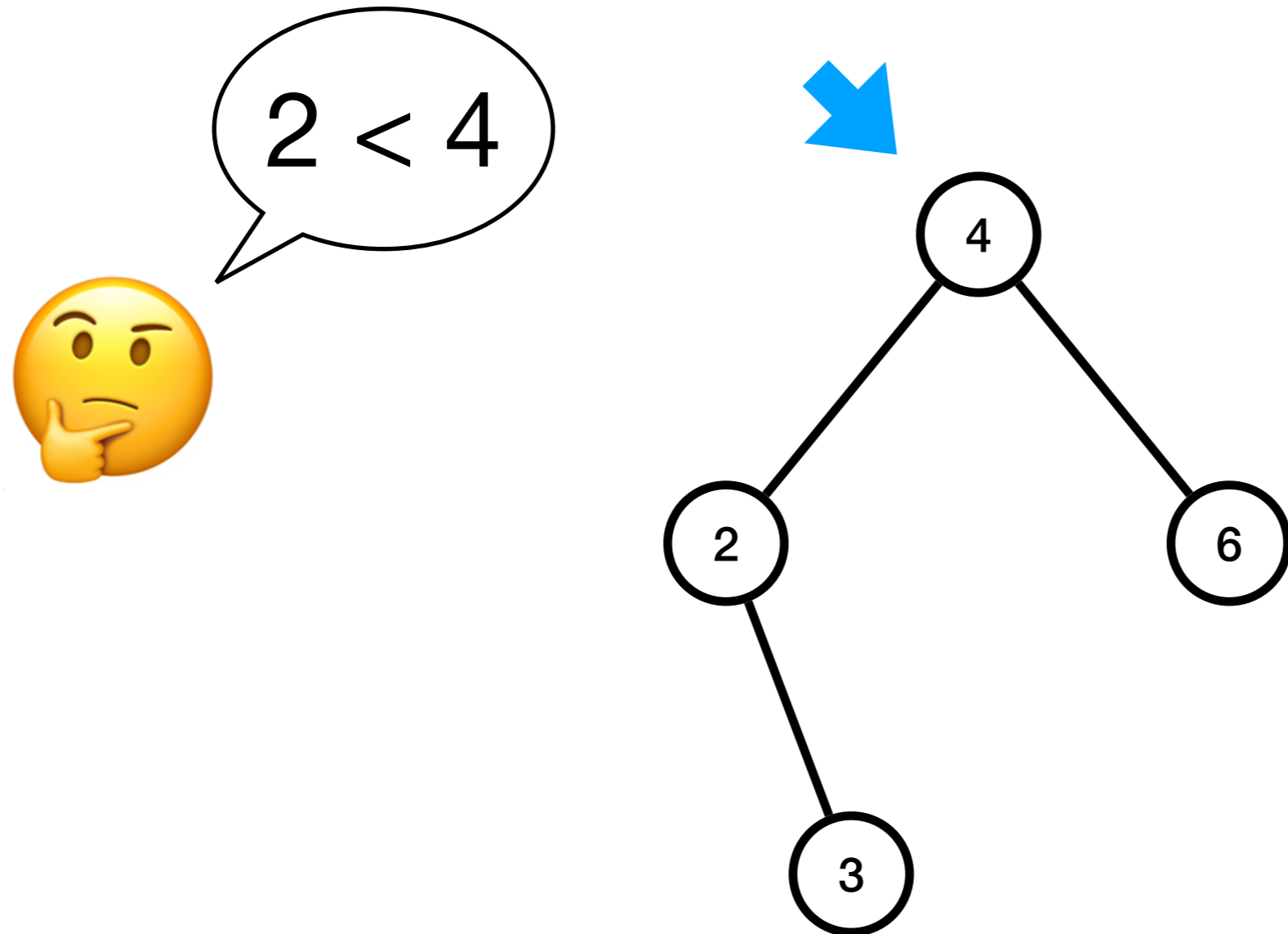
# Insertion dans un ABR



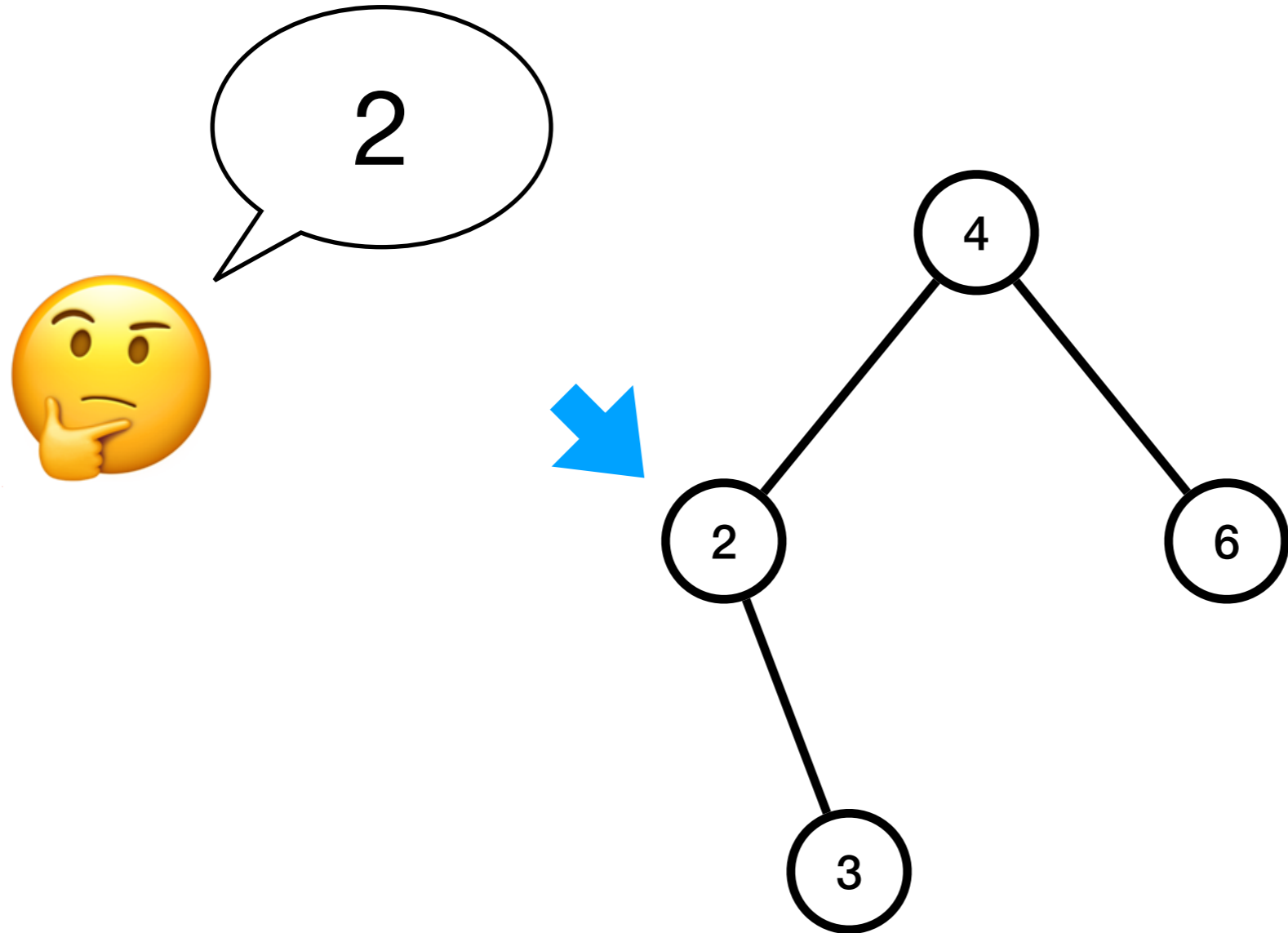
# Insertion dans un ABR



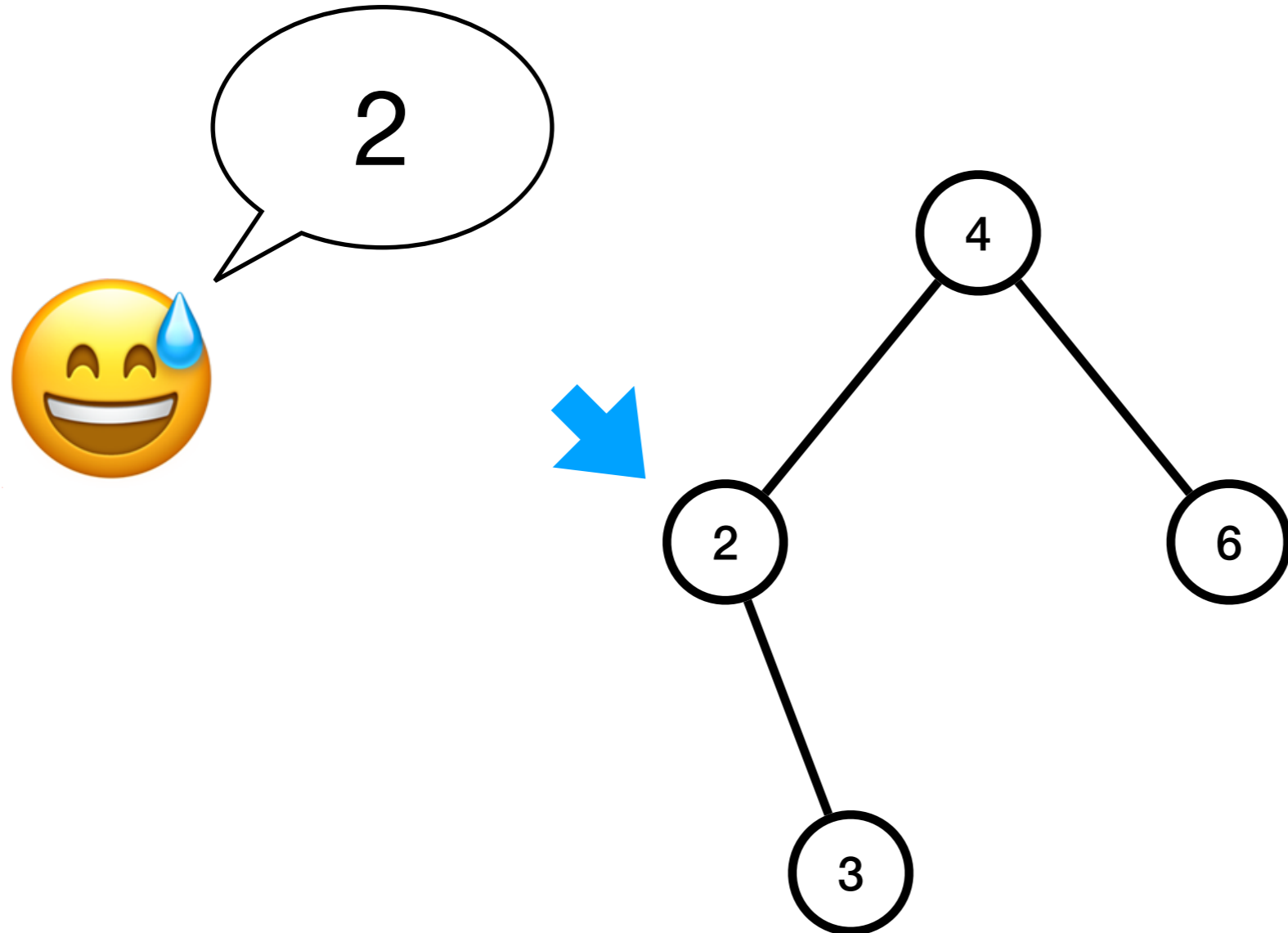
# Insertion dans un ABR



# Insertion dans un ABR

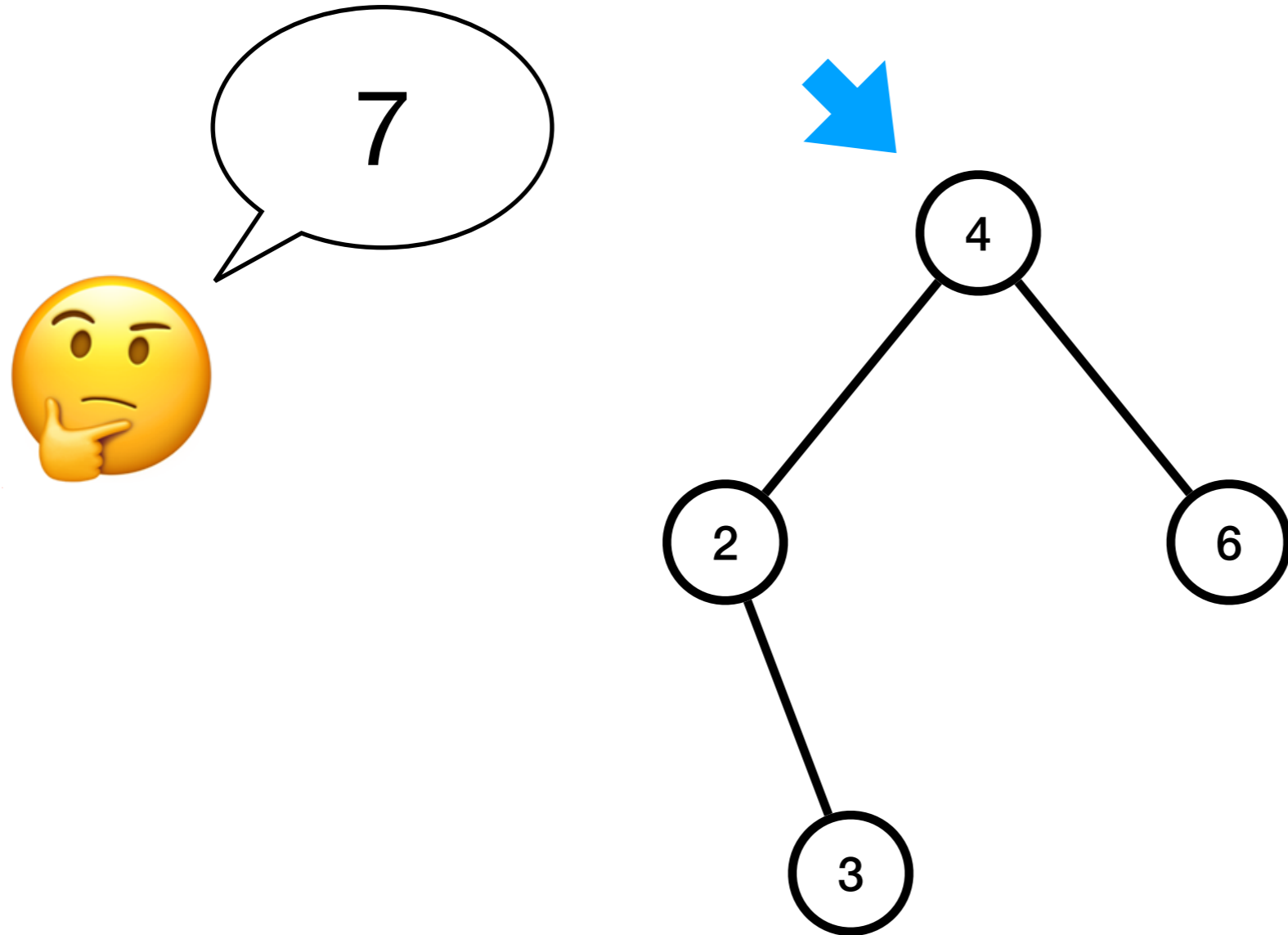


# Insertion dans un ABR

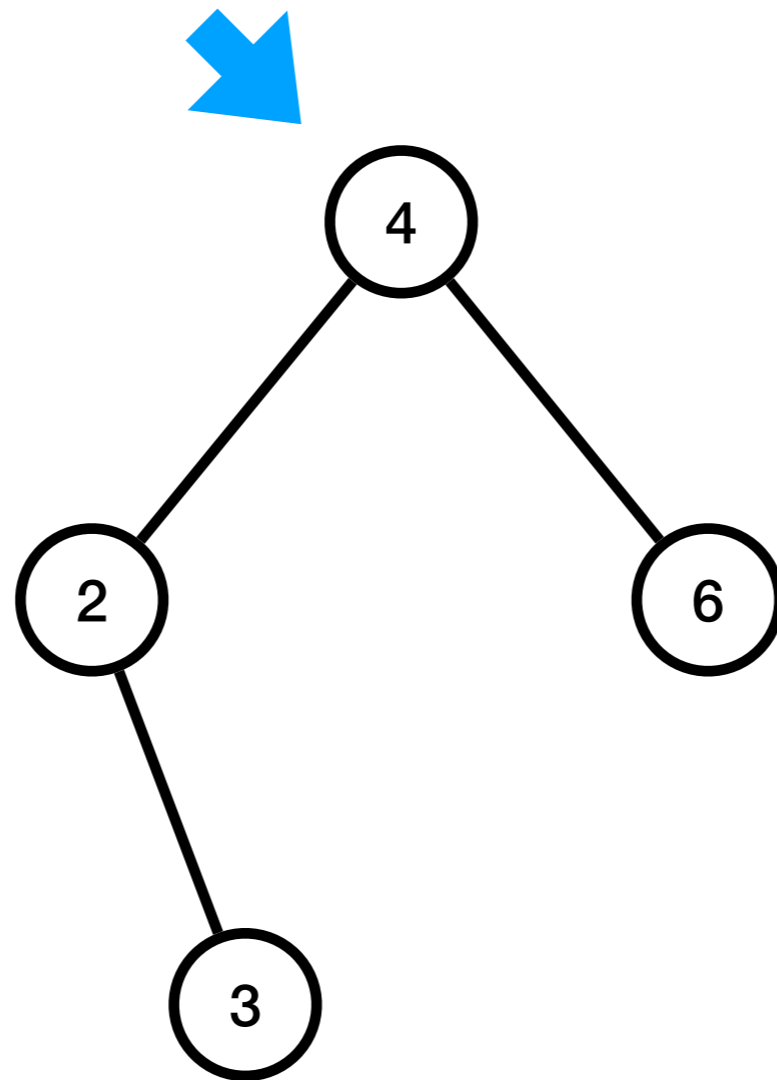
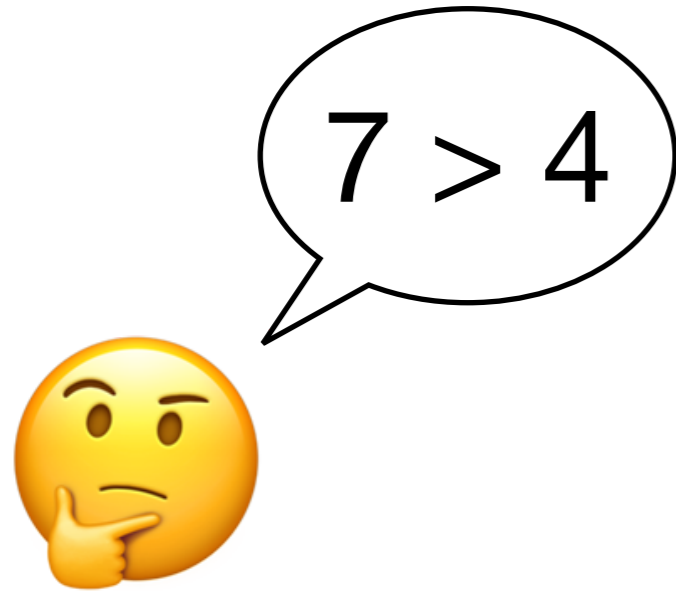




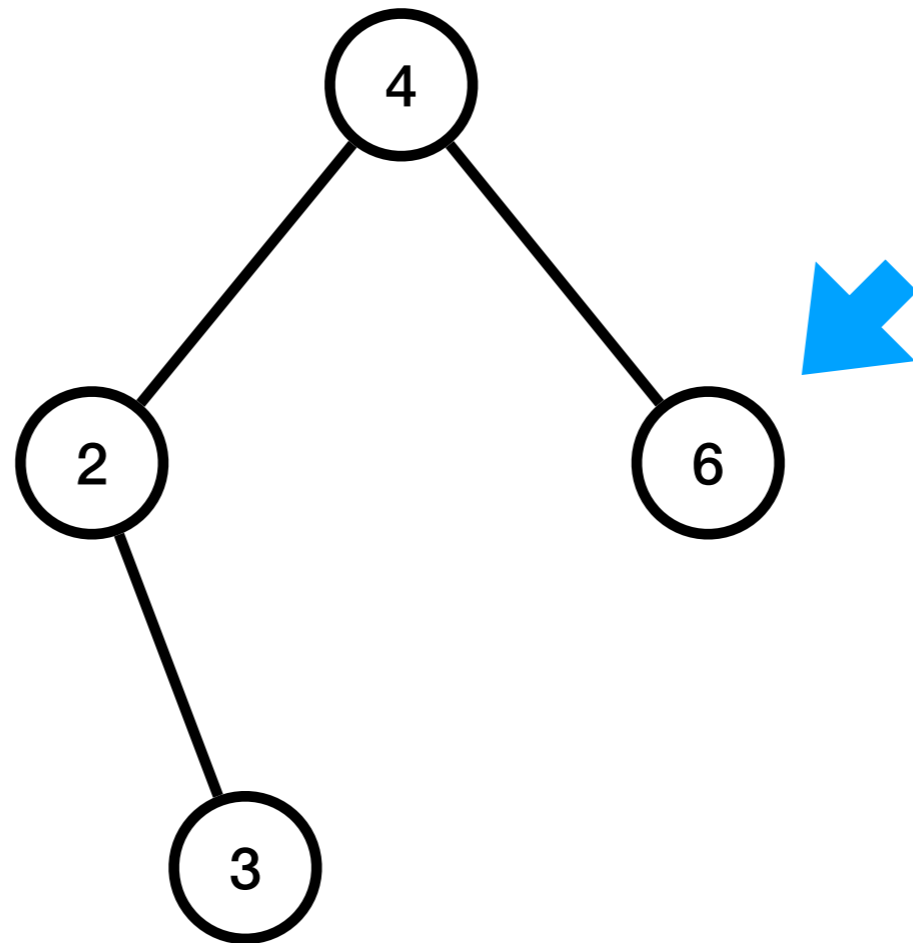
# Insertion dans un ABR



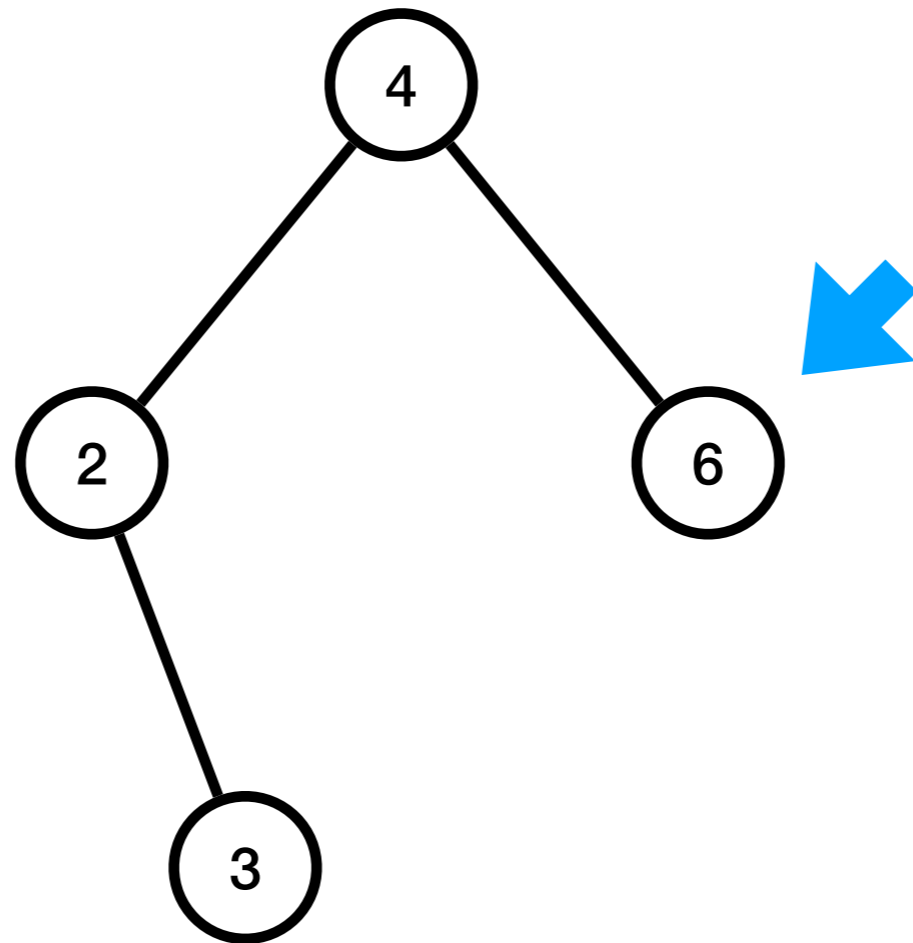
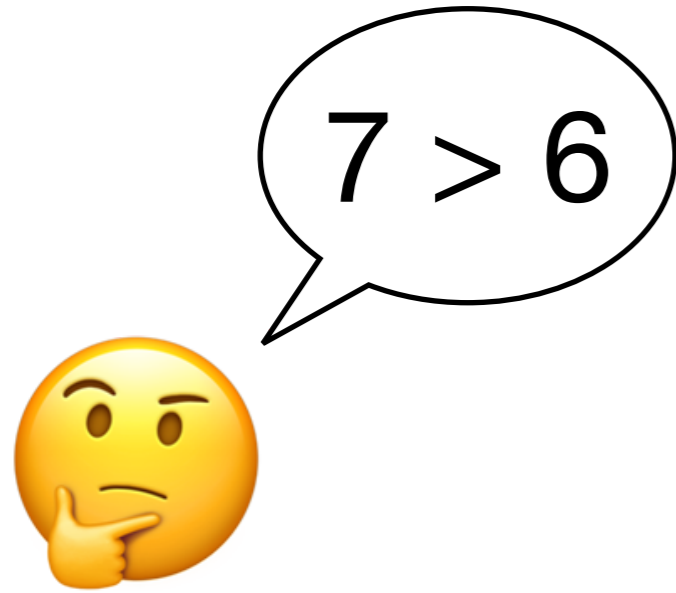
# Insertion dans un ABR



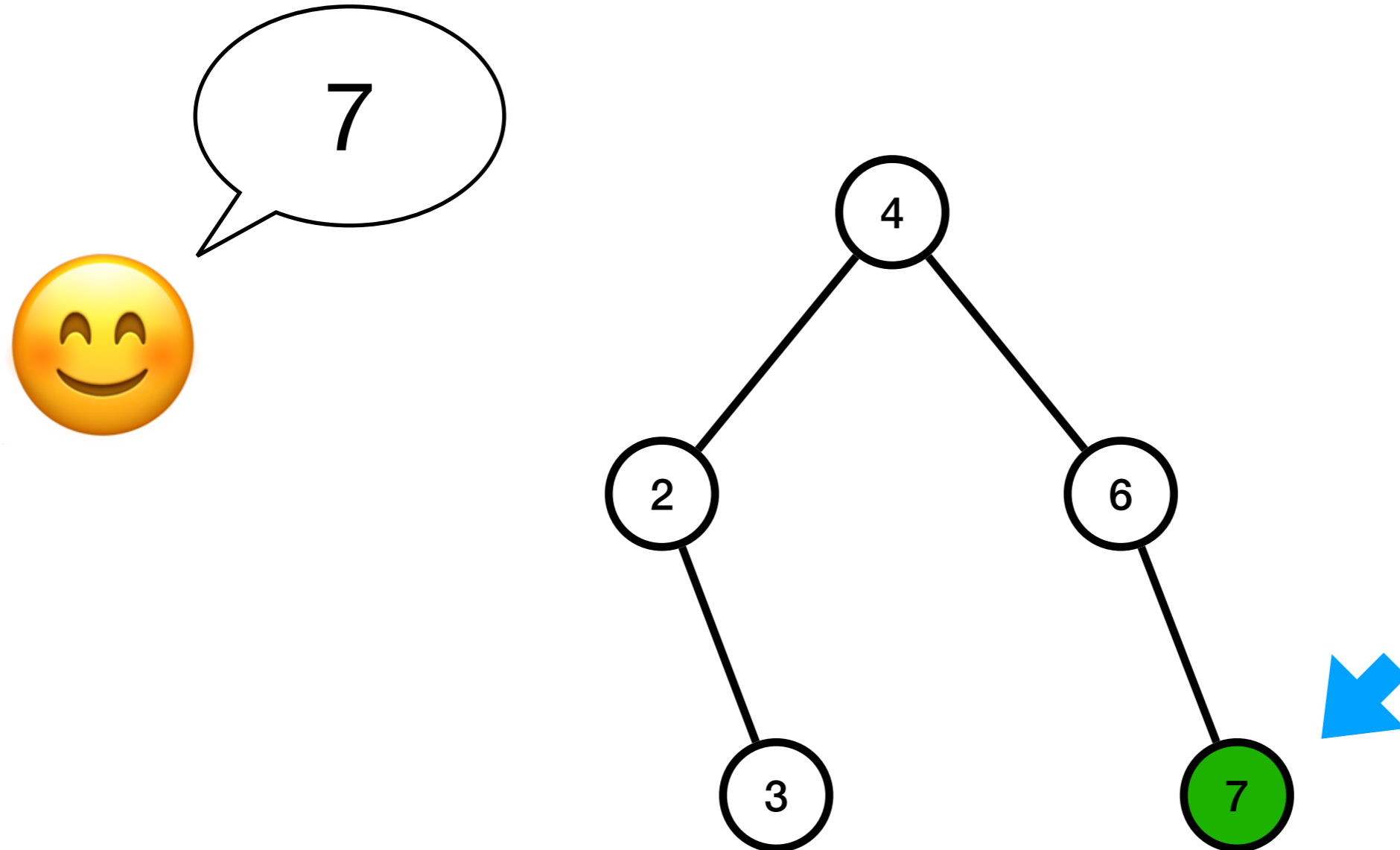
# Insertion dans un ABR



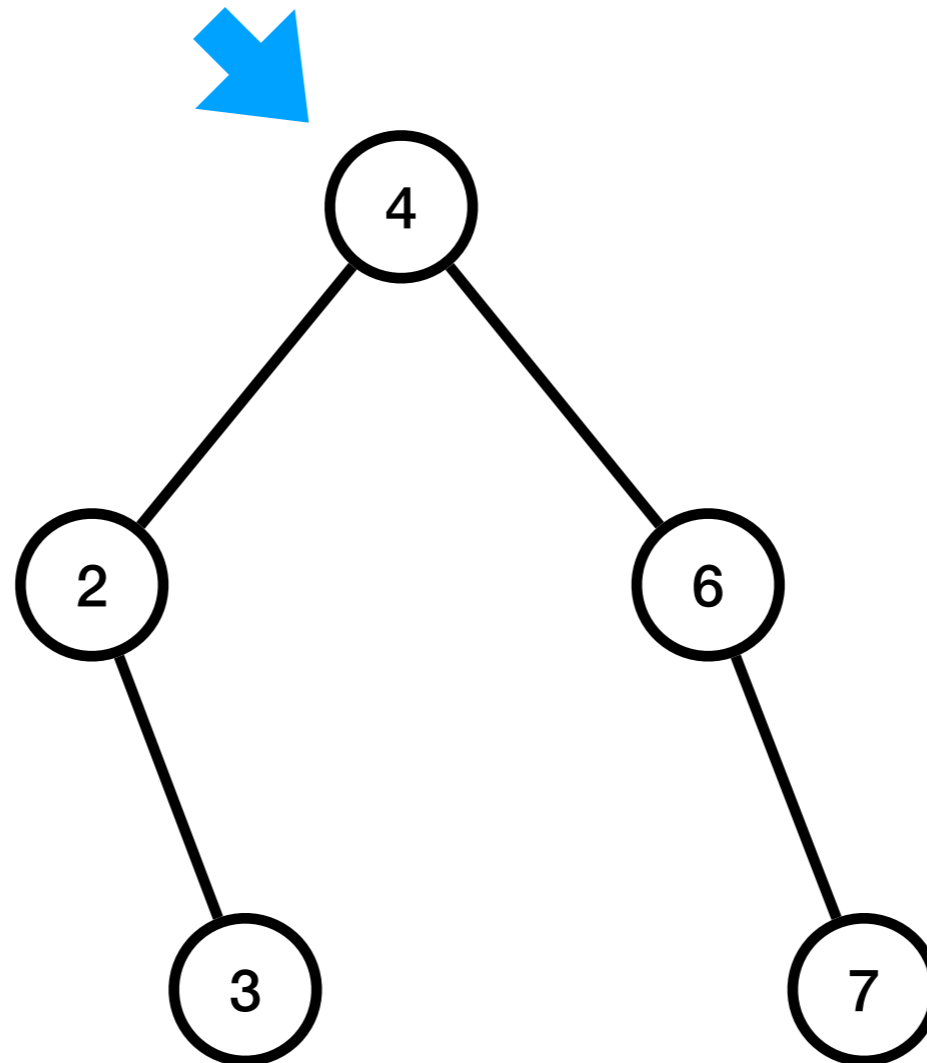
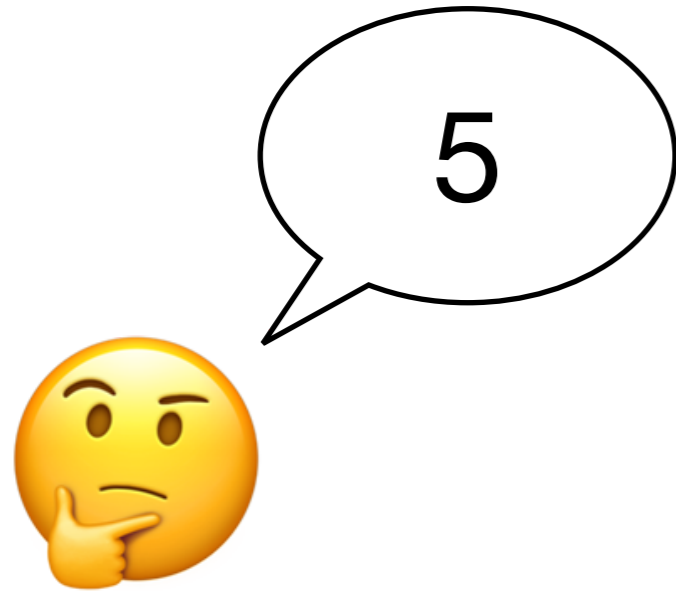
# Insertion dans un ABR



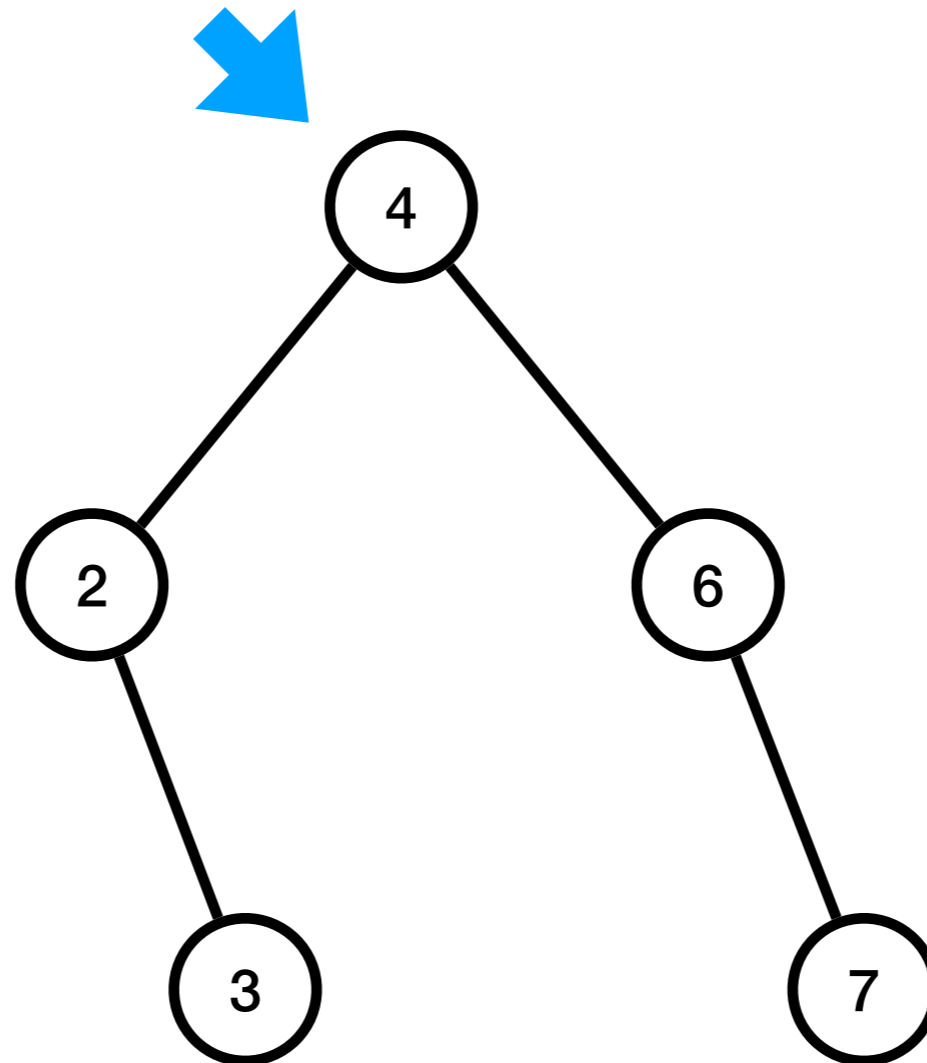
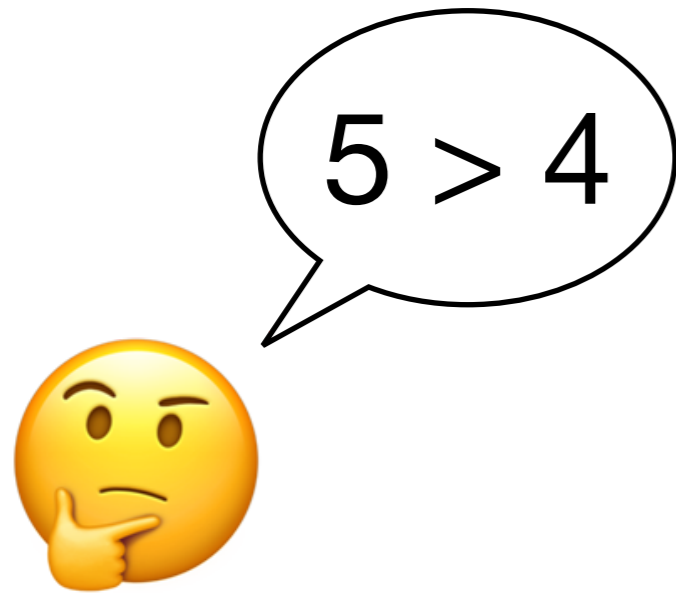
# Insertion dans un ABR



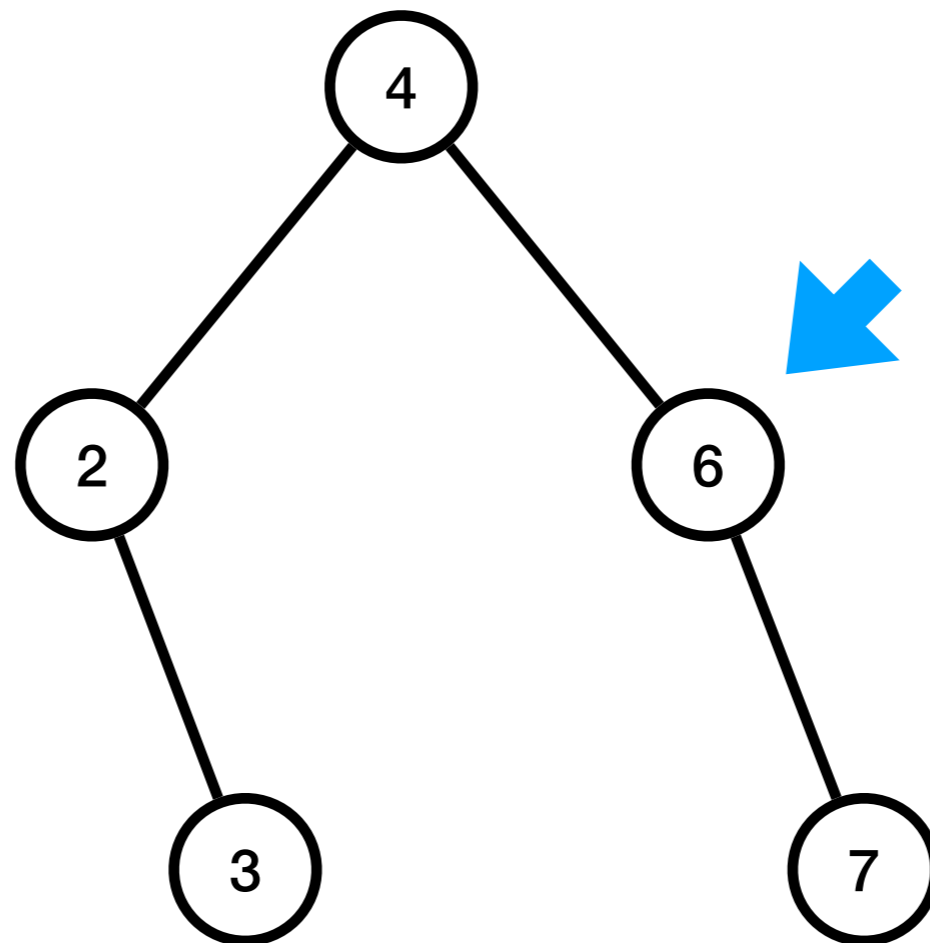
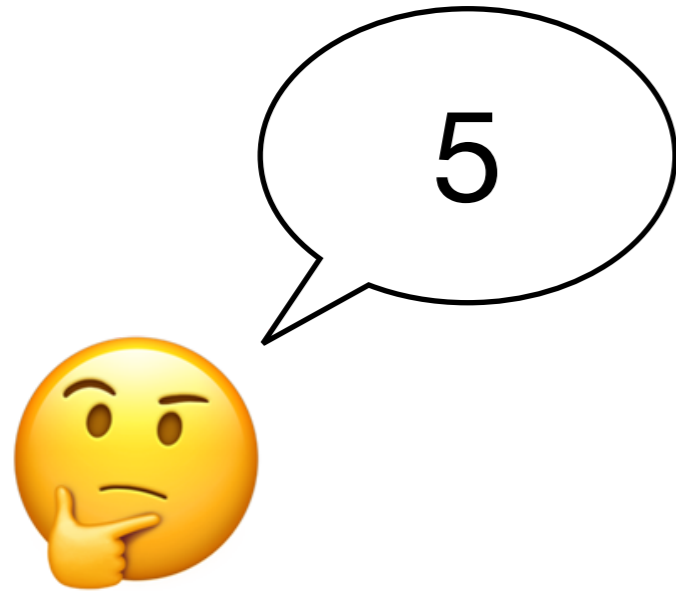
# Insertion dans un ABR



# Insertion dans un ABR

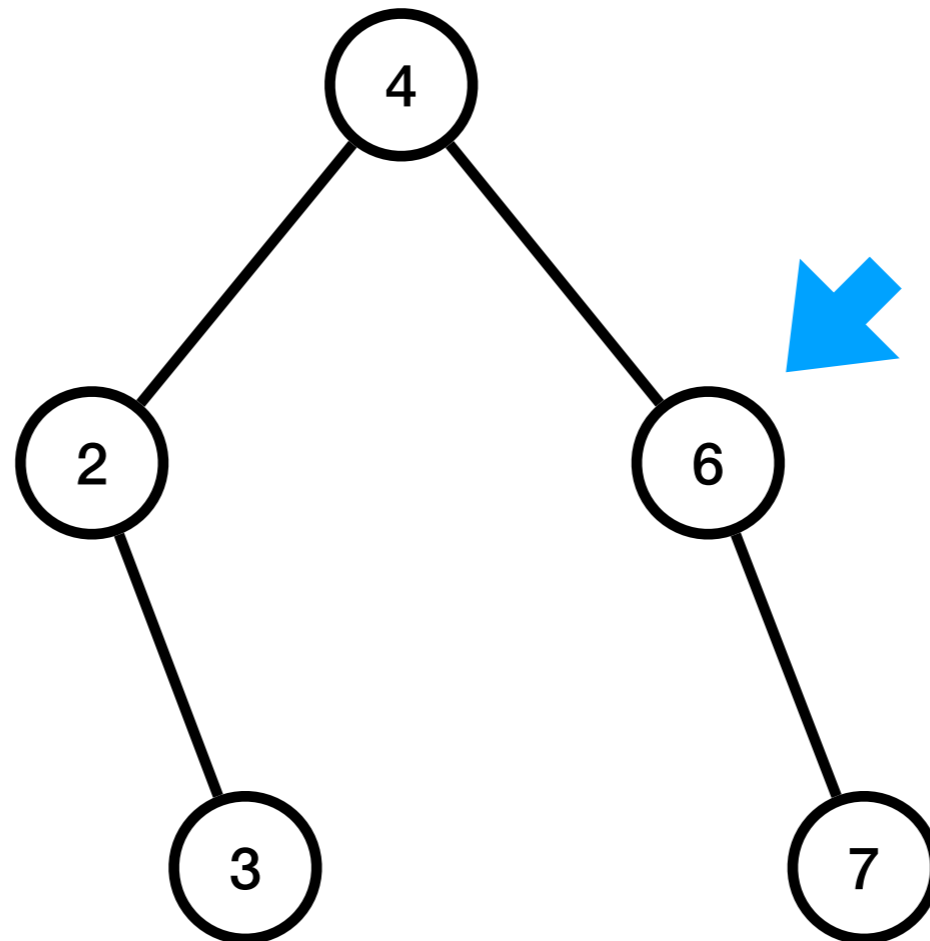
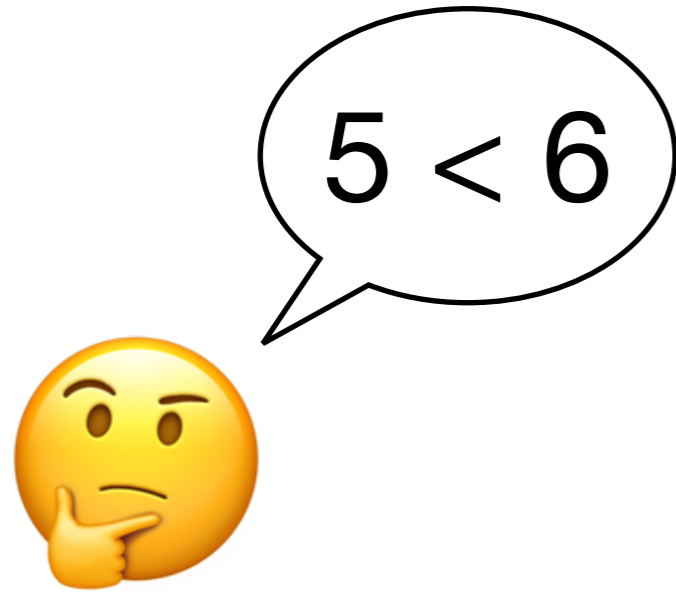


# Insertion dans un ABR

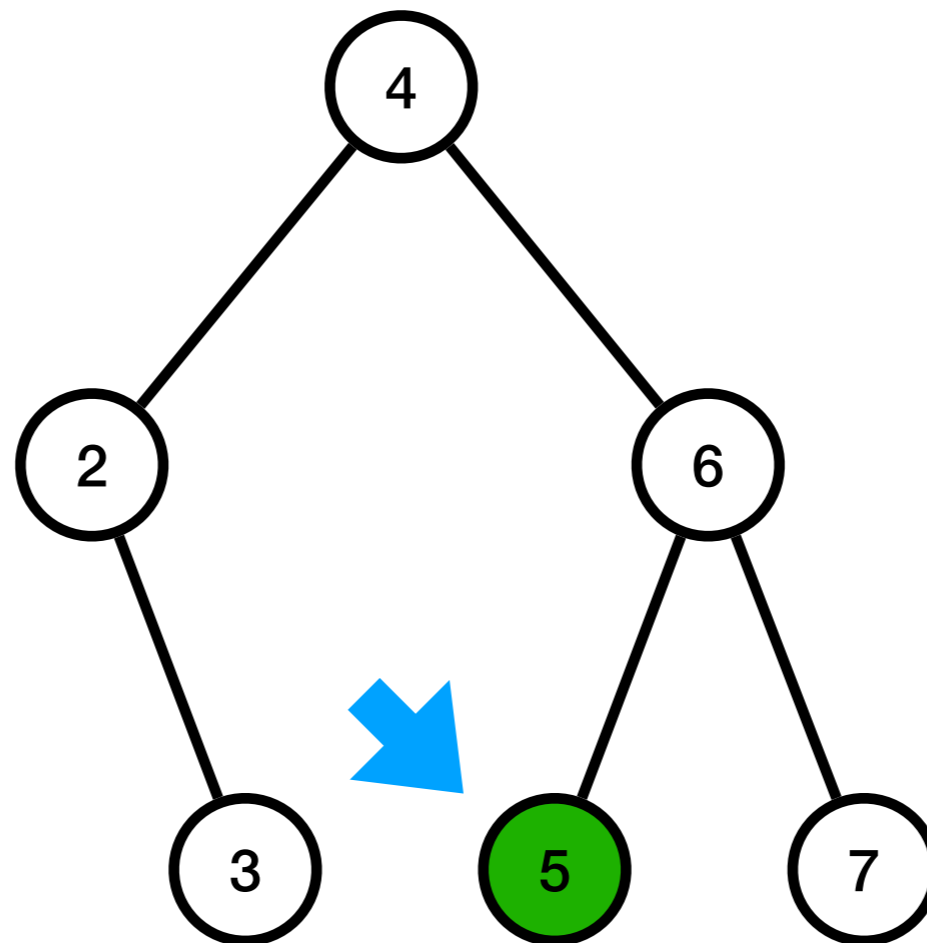
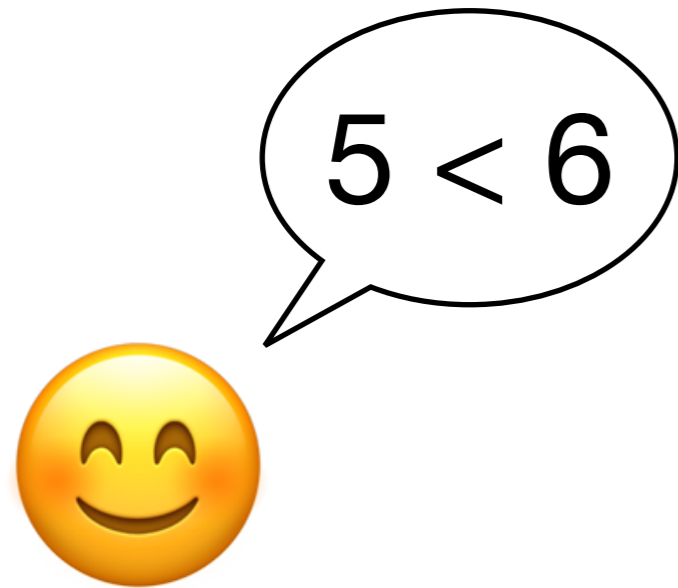




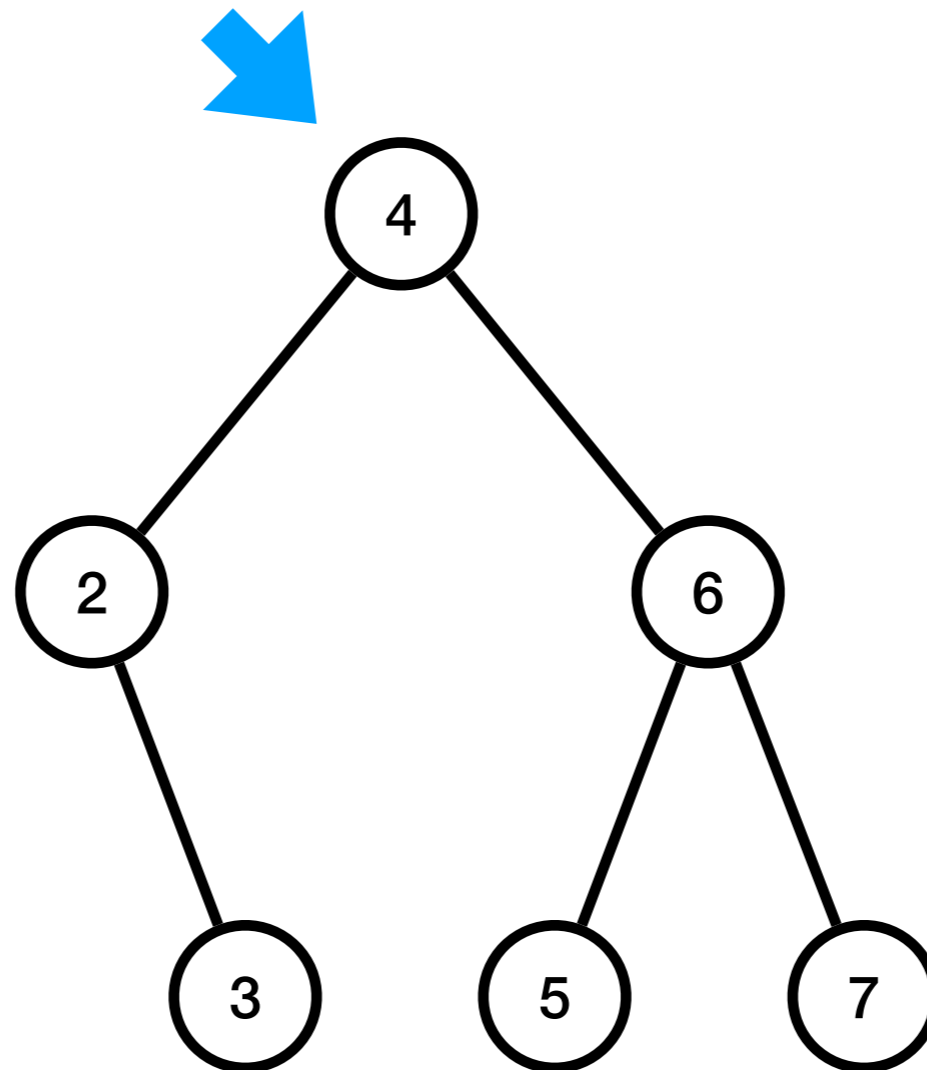
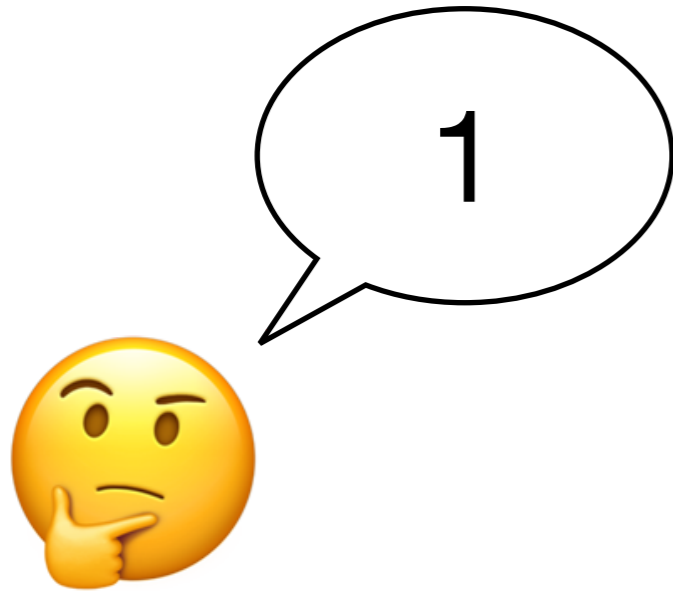
# Insertion dans un ABR



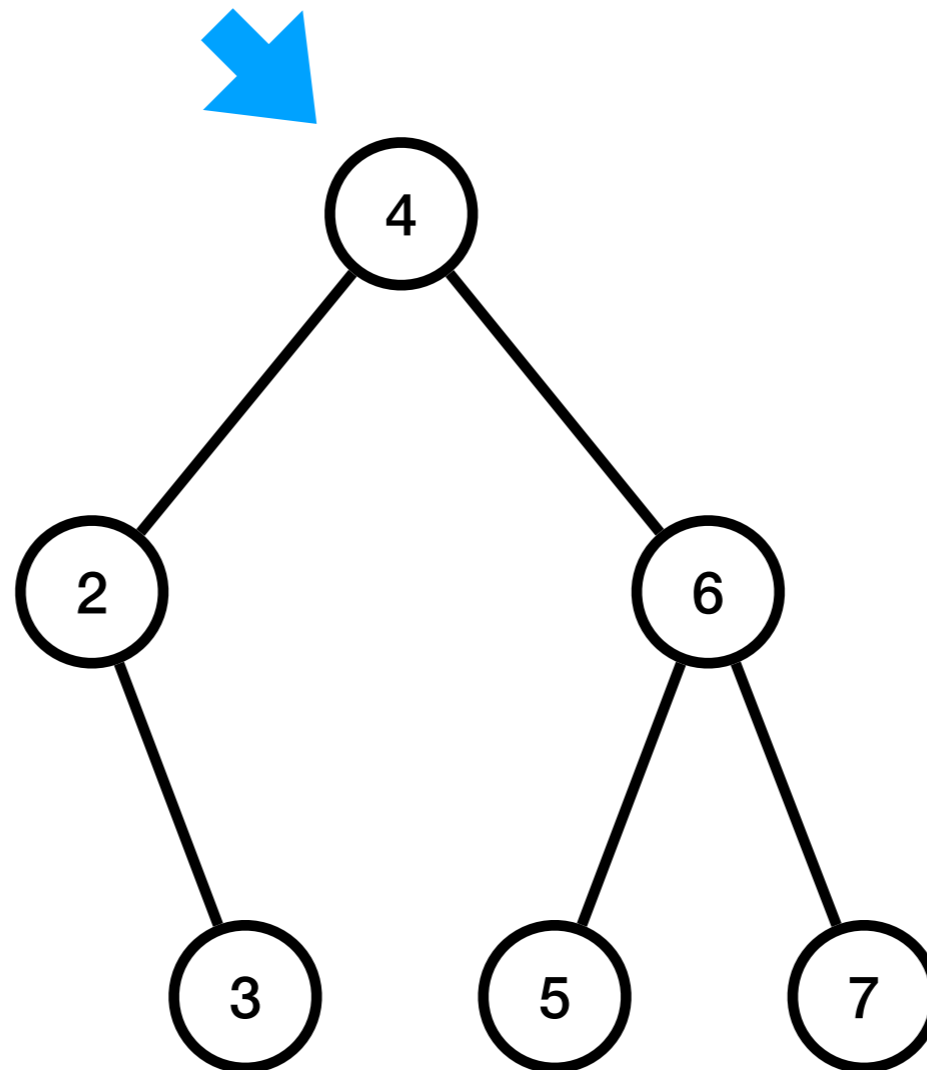
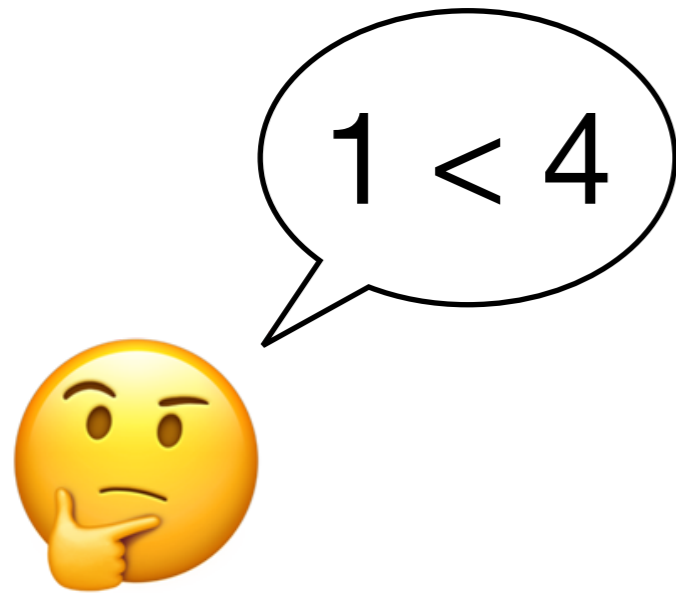
# Insertion dans un ABR



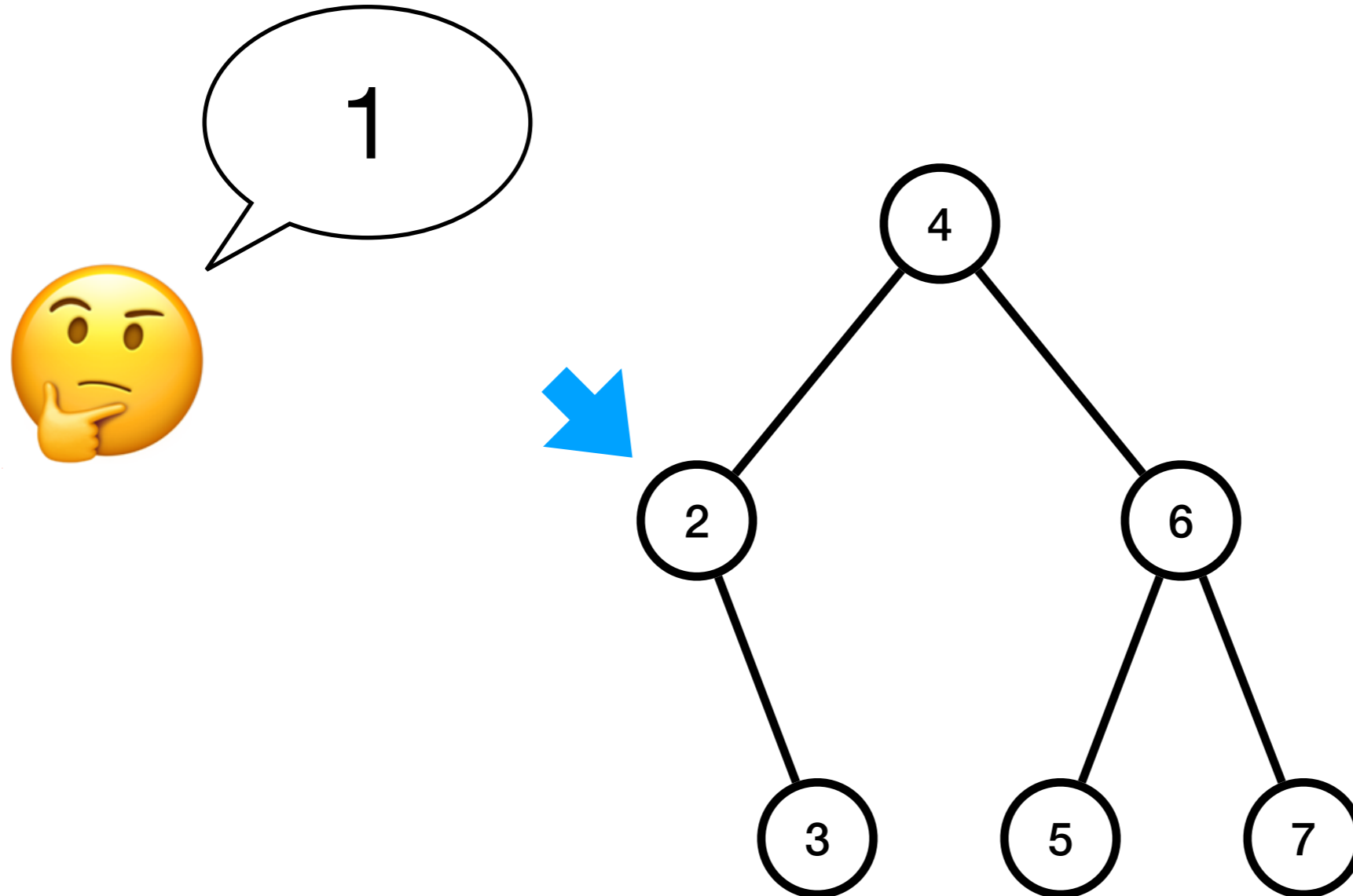
# Insertion dans un ABR



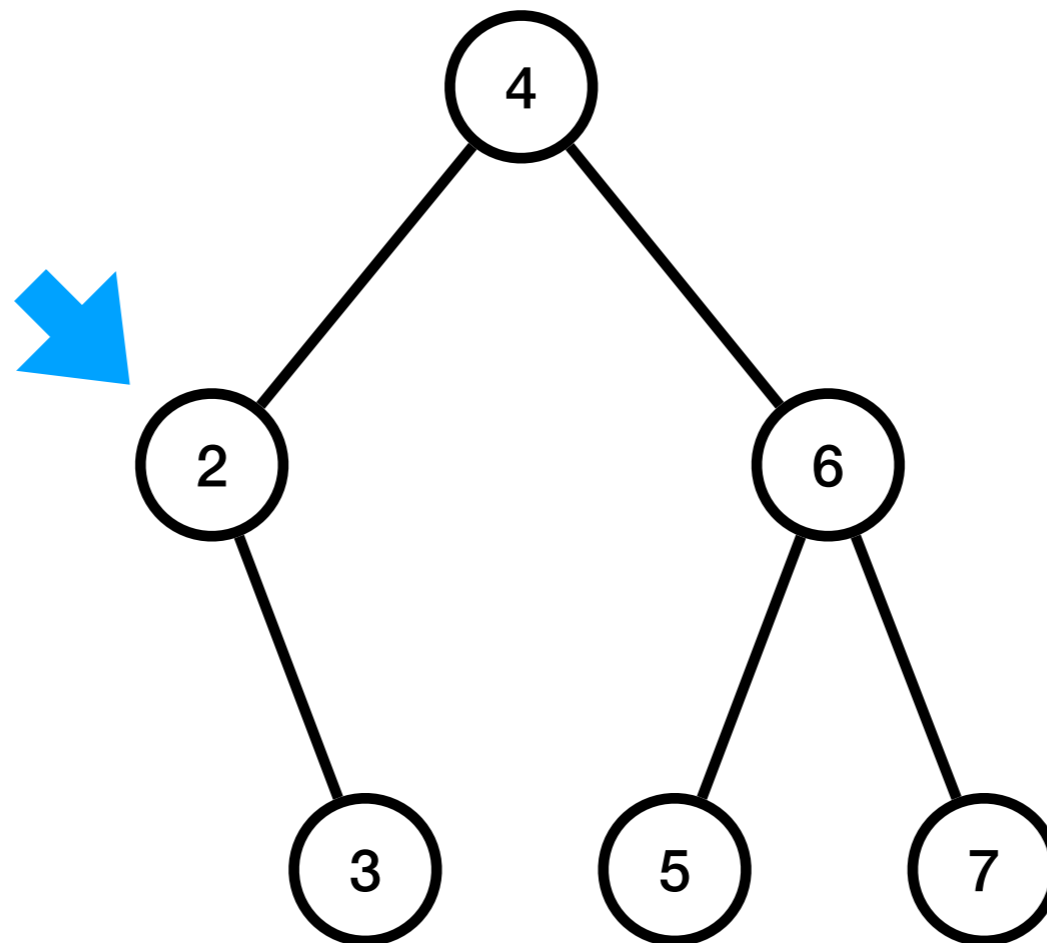
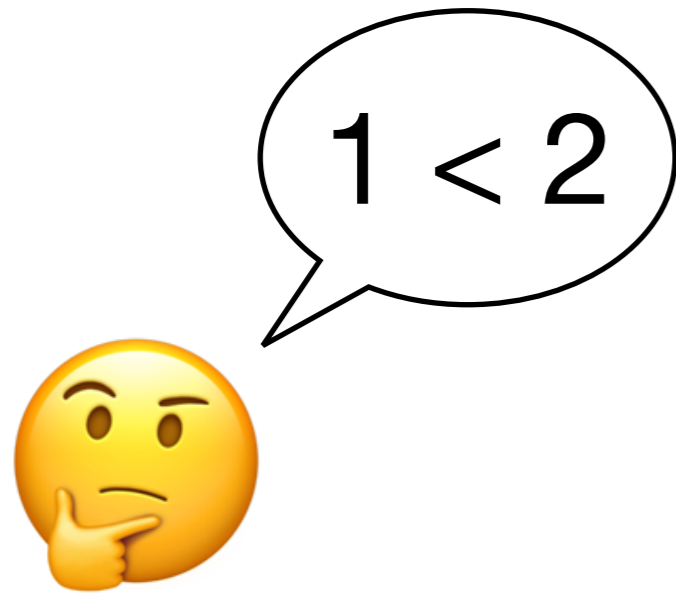
# Insertion dans un ABR



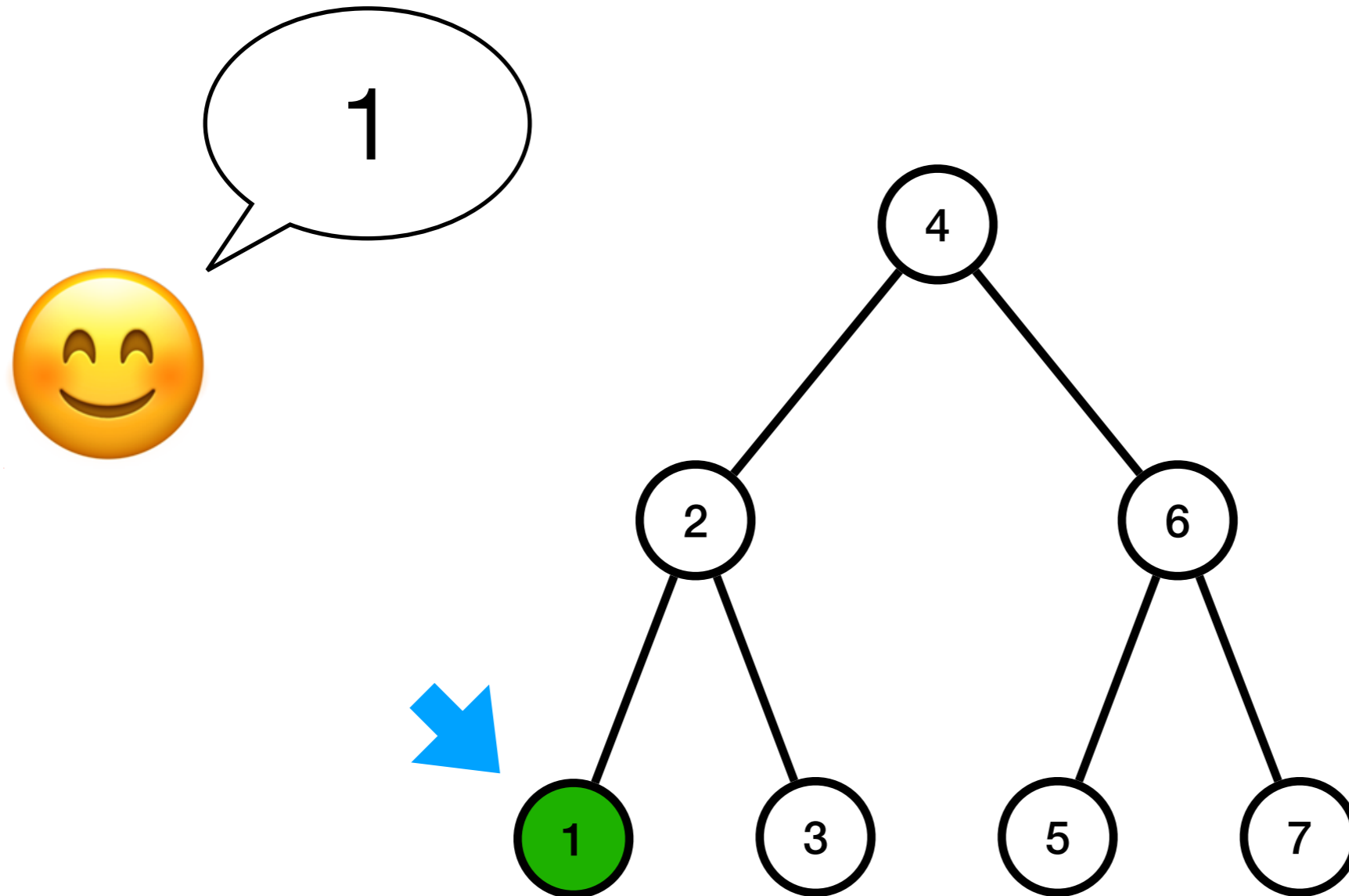
# Insertion dans un ABR



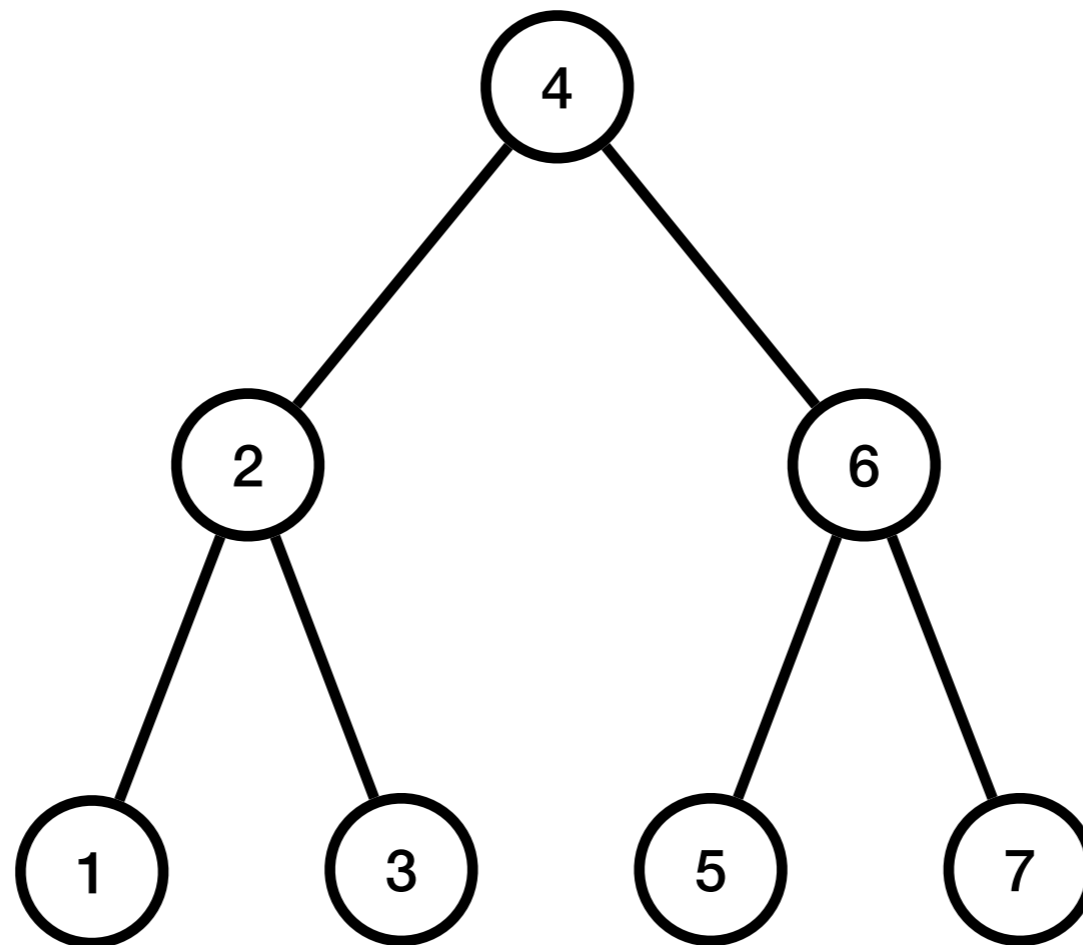
# Insertion dans un ABR



# Insertion dans un ABR



# Insertion dans un ABR





# Insertion dans un ABR

# Insertion dans un ABR

```
def insérer_abr(arbre, x):
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             enfant_gauche,  
                             droite(arbre))
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             enfant_gauche,  
                             droite(arbre))  
    elif x > valeur(arbre):
```



# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             enfant_gauche,  
                             droite(arbre))  
    elif x > valeur(arbre):  
        enfant_droite = insérer_abr(droite(arbre), x)
```

# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             enfant_gauche,  
                             droite(arbre))  
    elif x > valeur(arbre):  
        enfant_droite = insérer_abr(droite(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             gauche(arbre),  
                             enfant_droite)
```



# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             enfant_gauche,  
                             droite(arbre))  
    elif x > valeur(arbre):  
        enfant_droite = insérer_abr(droite(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             gauche(arbre),  
                             enfant_droite)  
    else:  
        # x == valeur(arbre)
```

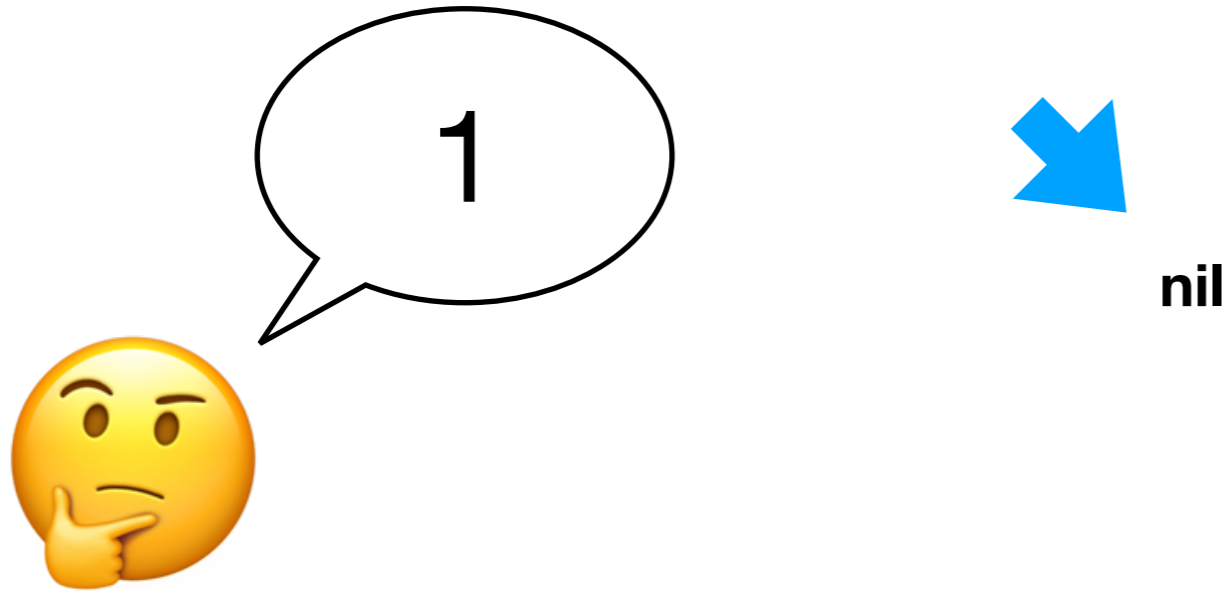
# Insertion dans un ABR

```
def insérer_abr(arbre, x):  
    if est_vide(arbre):  
        return nouveau_nœud(x, None, None)  
    elif x < valeur(arbre):  
        enfant_gauche = insérer_abr(gauche(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             enfant_gauche,  
                             droite(arbre))  
    elif x > valeur(arbre):  
        enfant_droite = insérer_abr(droite(arbre), x)  
        return nouveau_nœud(valeur(arbre),  
                             gauche(arbre),  
                             enfant_droite)  
    else:  
        # x == valeur(arbre)  
        return arbre
```

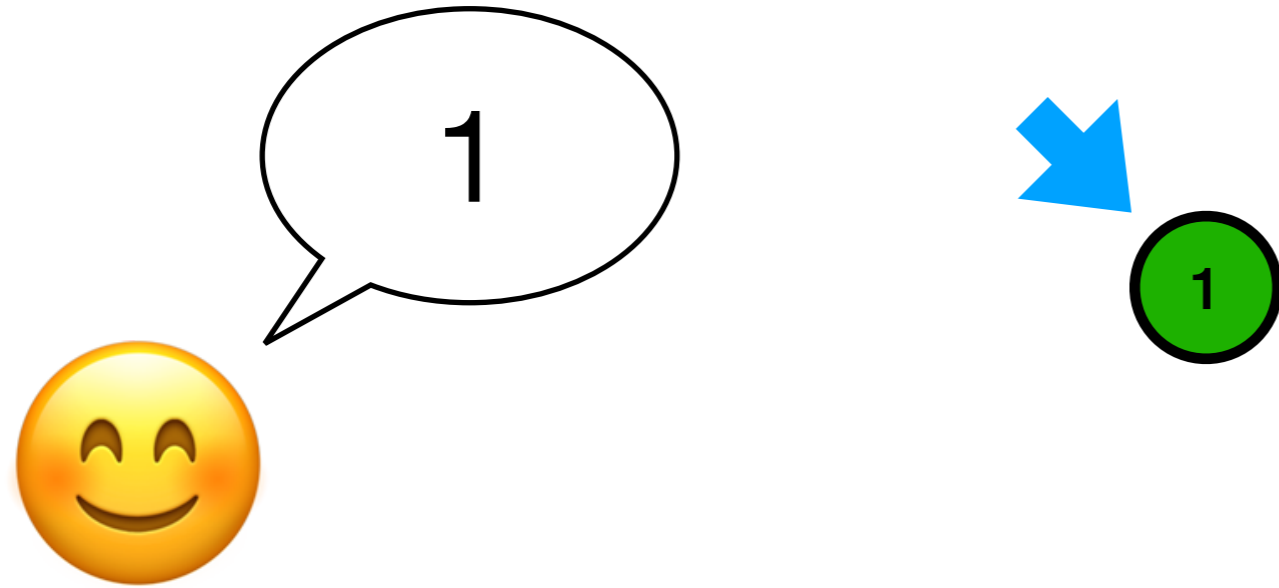
# Insertion dans un ABR

nil

# Insertion dans un ABR

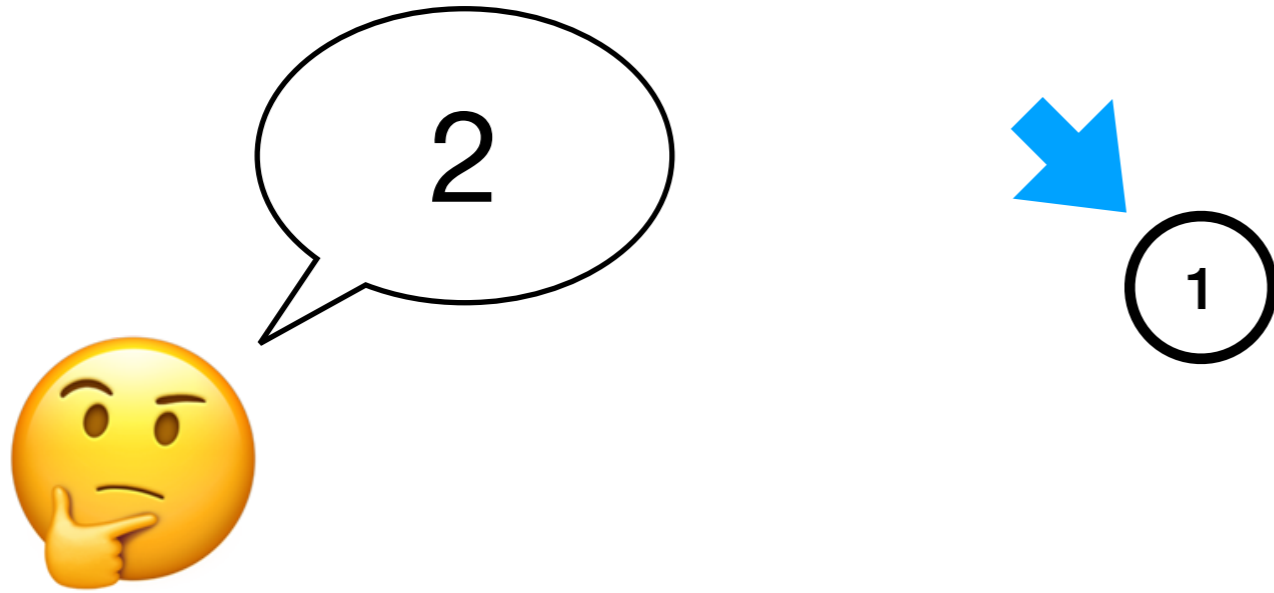


# Insertion dans un ABR

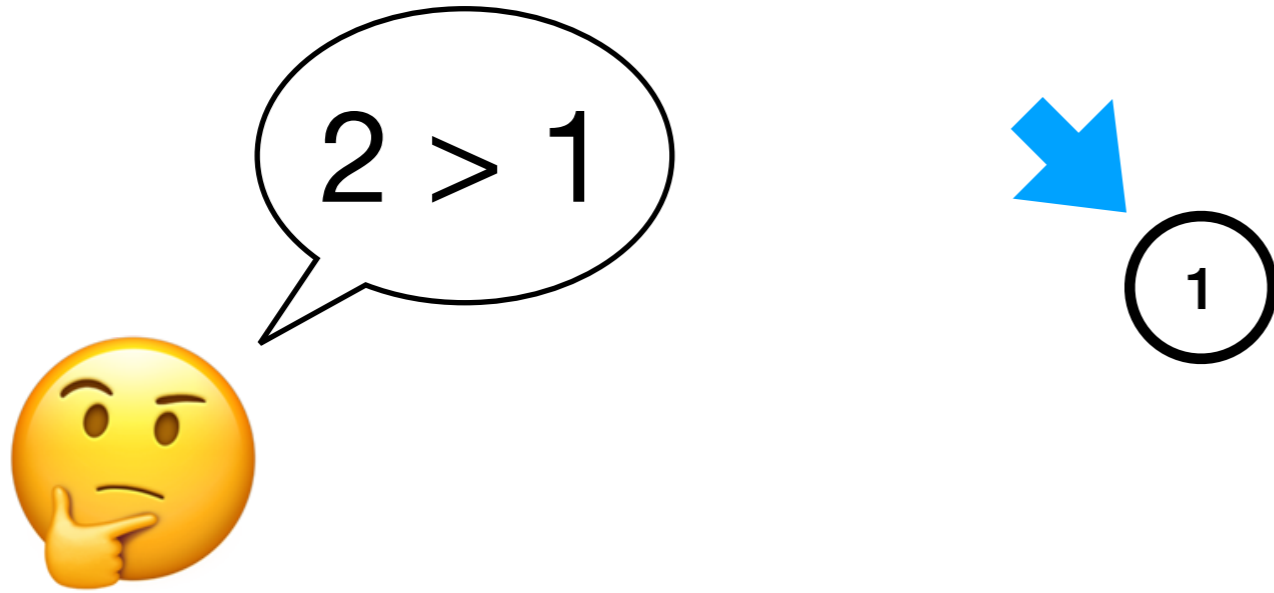




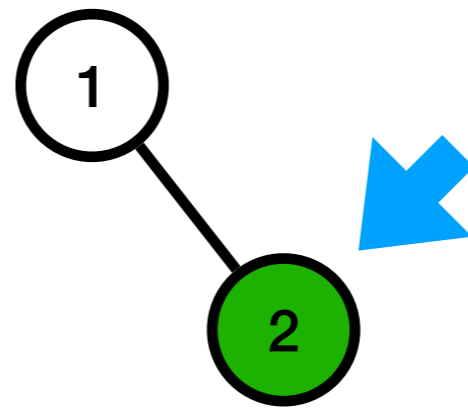
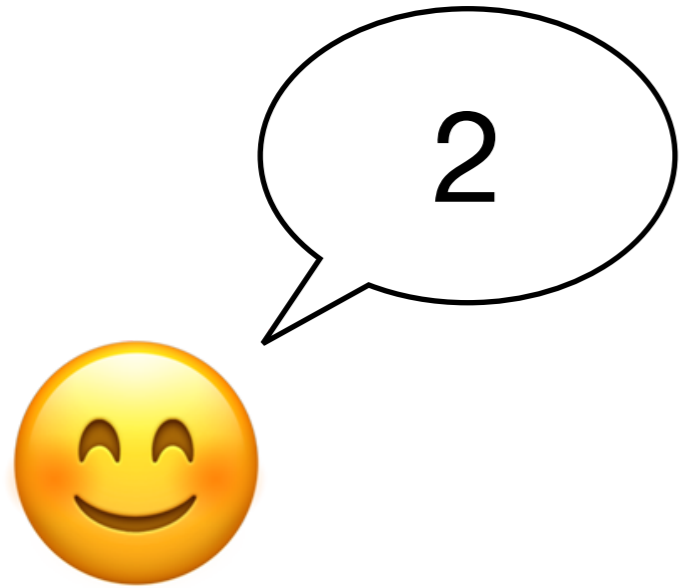
# Insertion dans un ABR



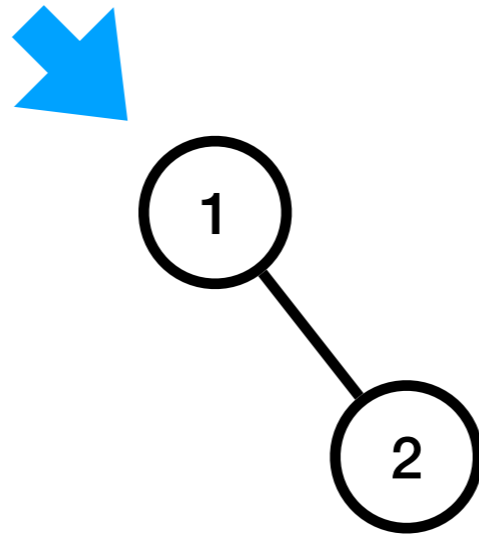
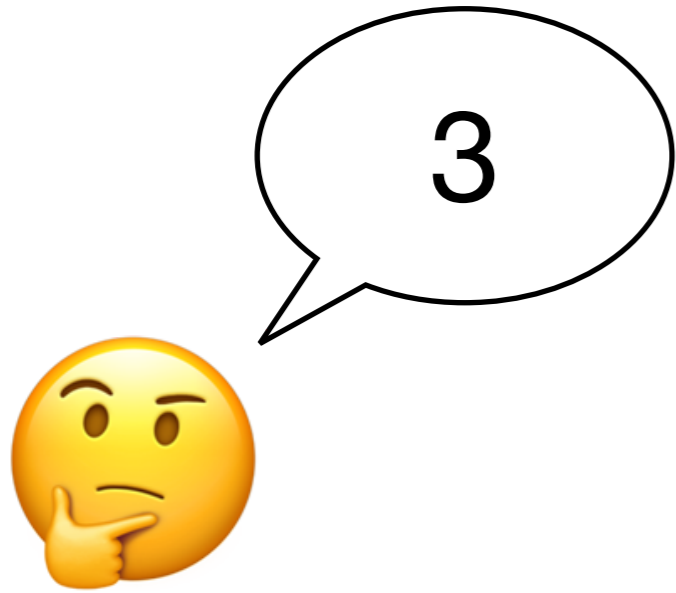
# Insertion dans un ABR



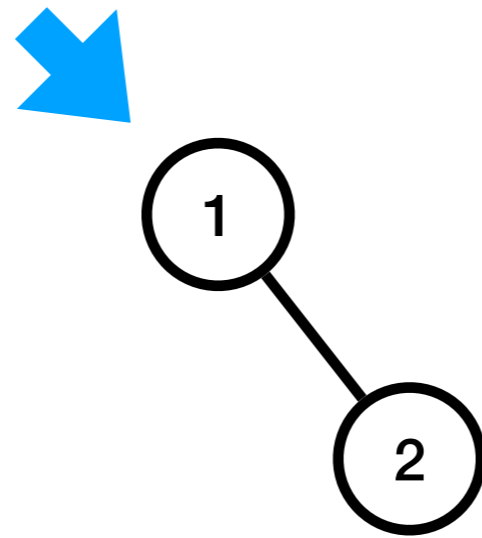
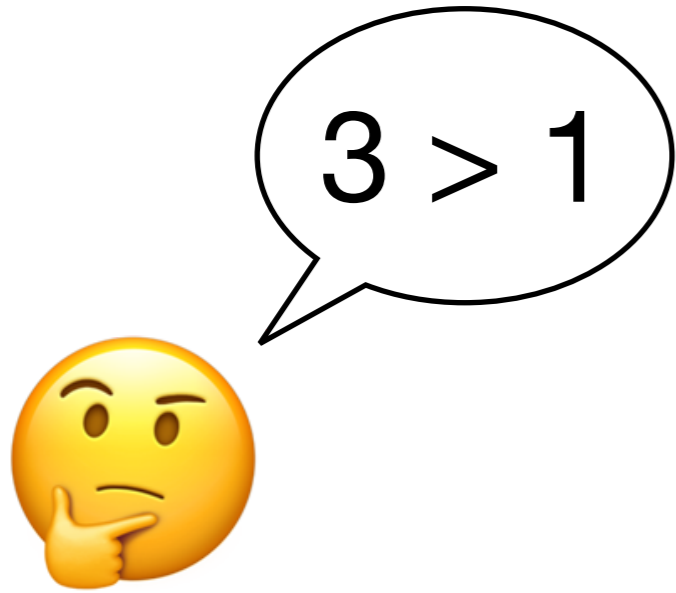
# Insertion dans un ABR



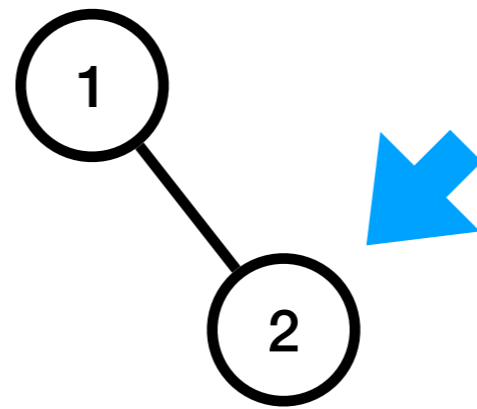
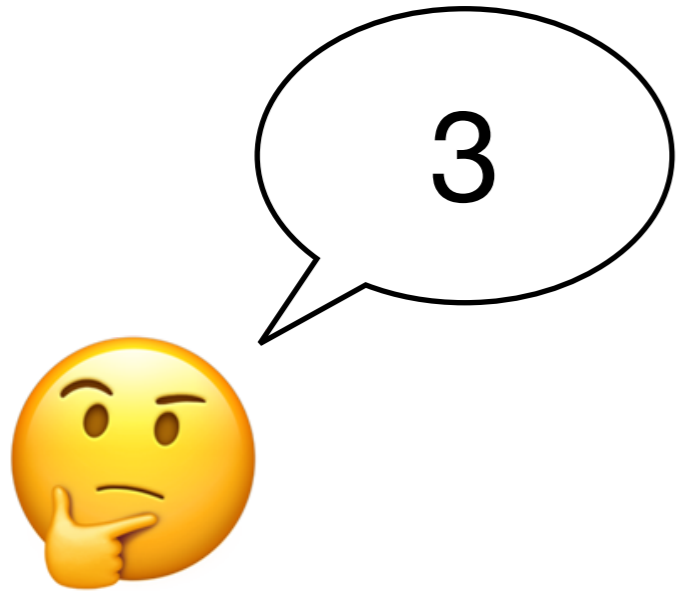
# Insertion dans un ABR



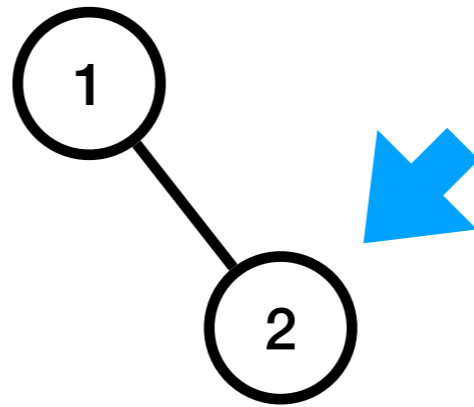
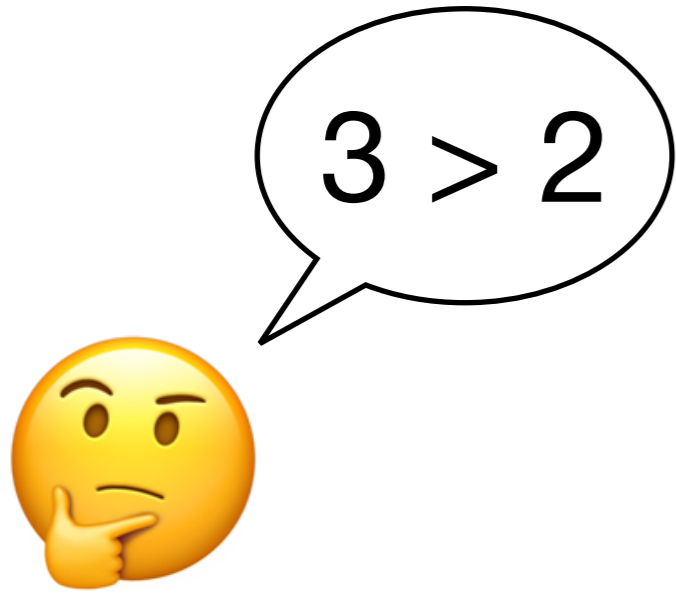
# Insertion dans un ABR



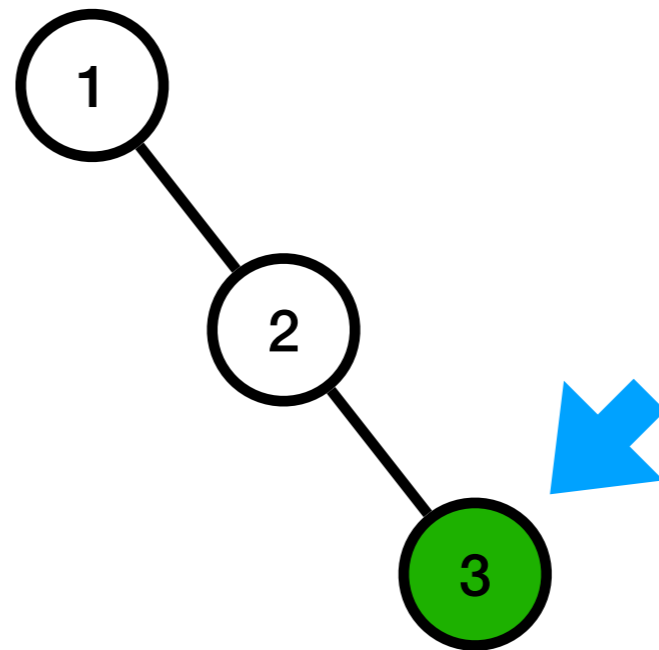
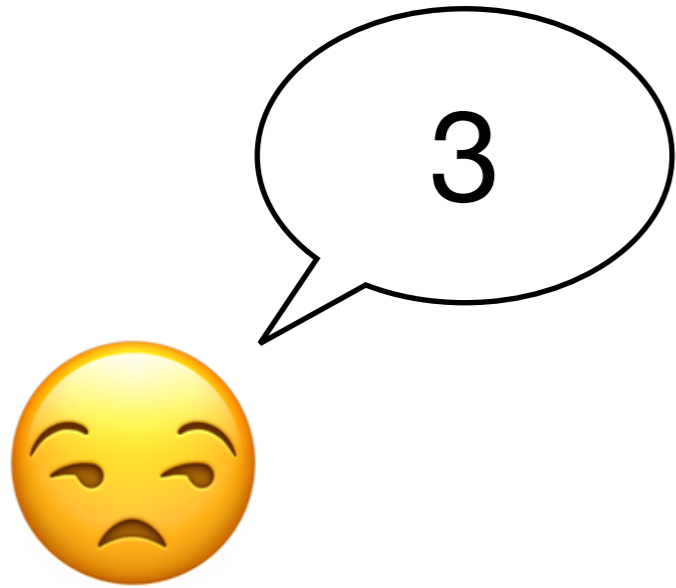
# Insertion dans un ABR



# Insertion dans un ABR

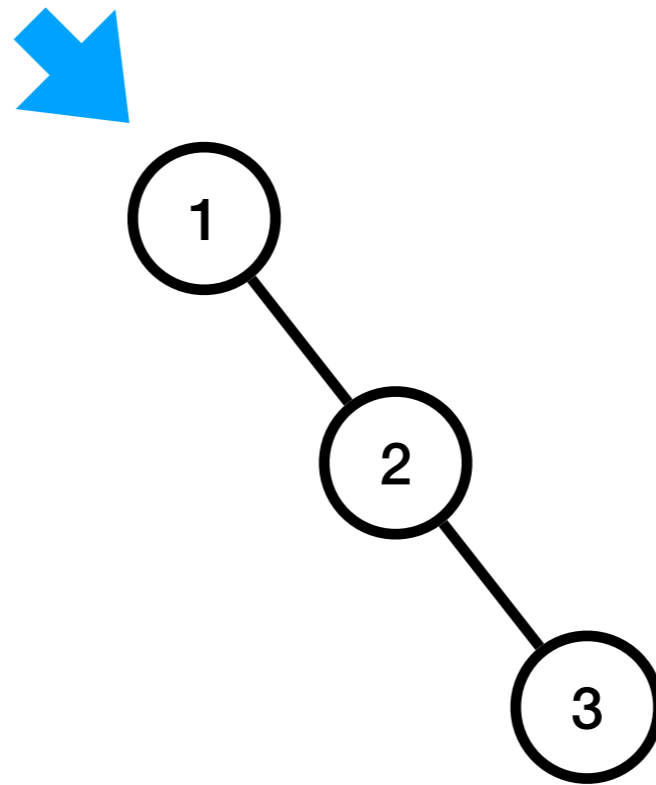
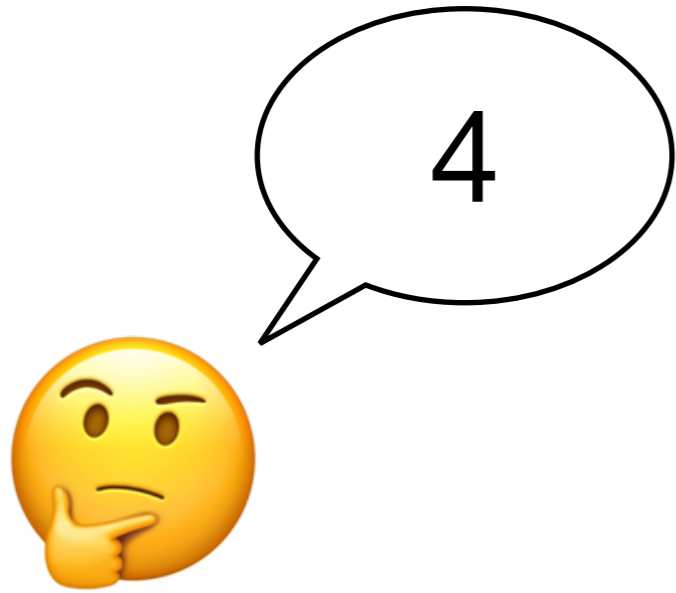


# Insertion dans un ABR

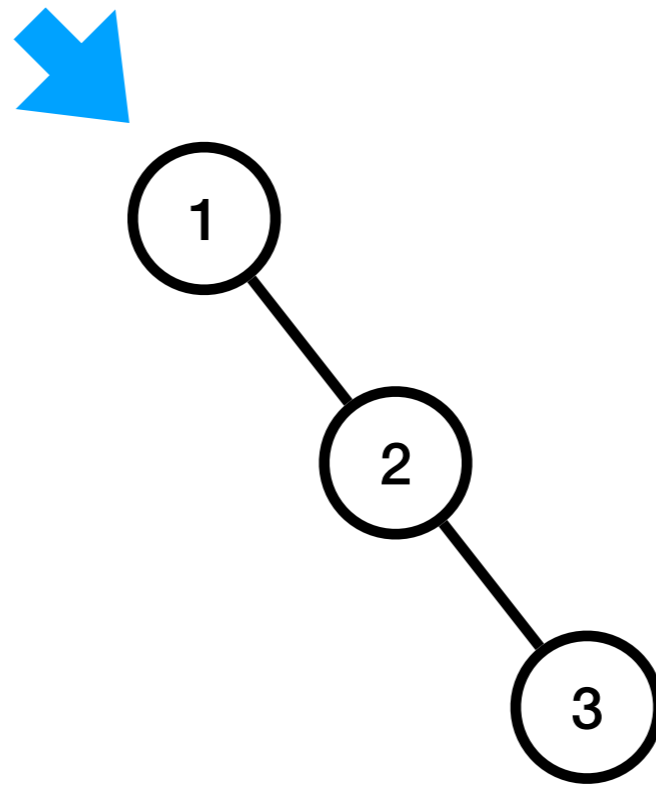
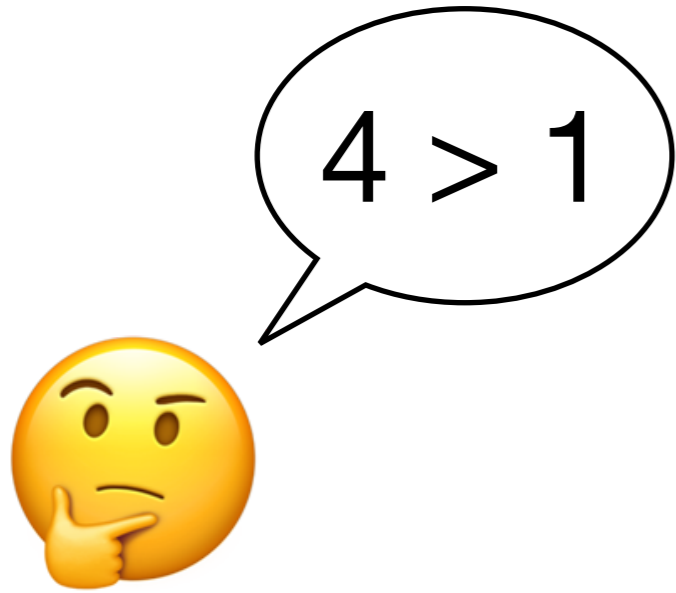




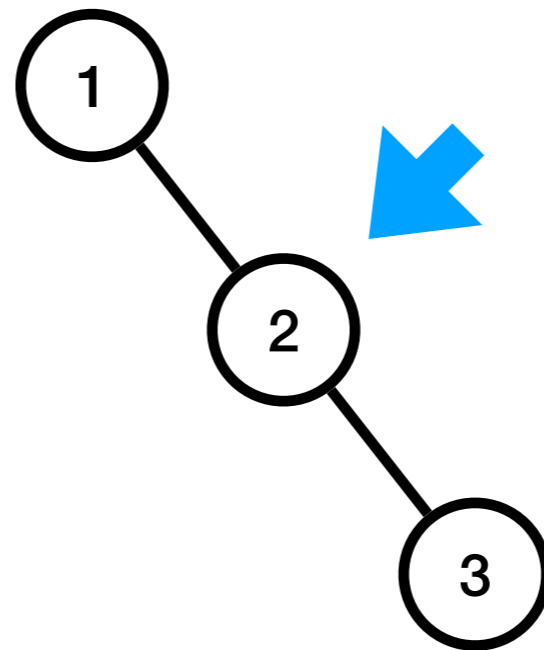
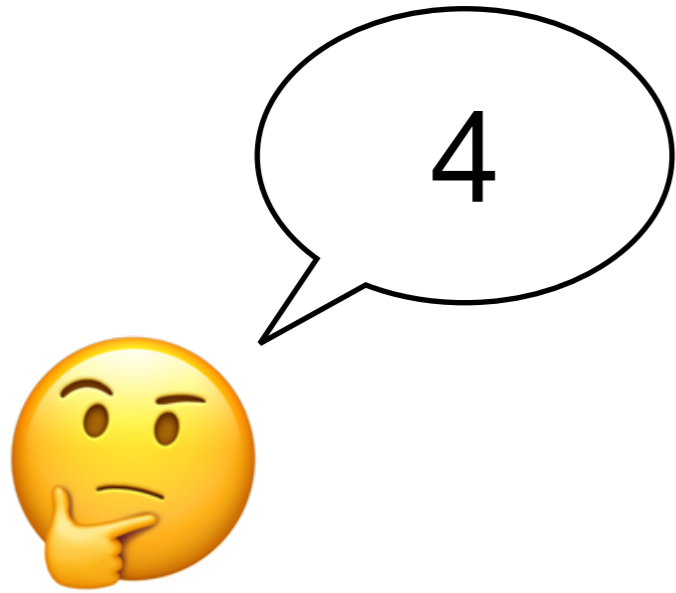
# Insertion dans un ABR



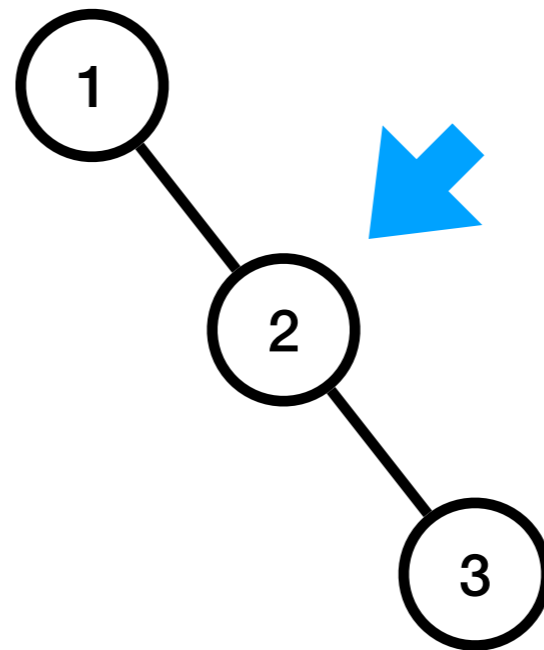
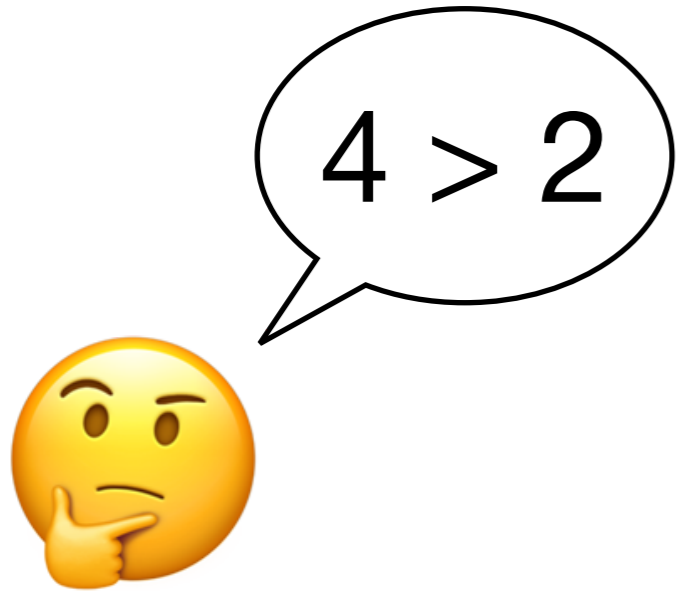
# Insertion dans un ABR



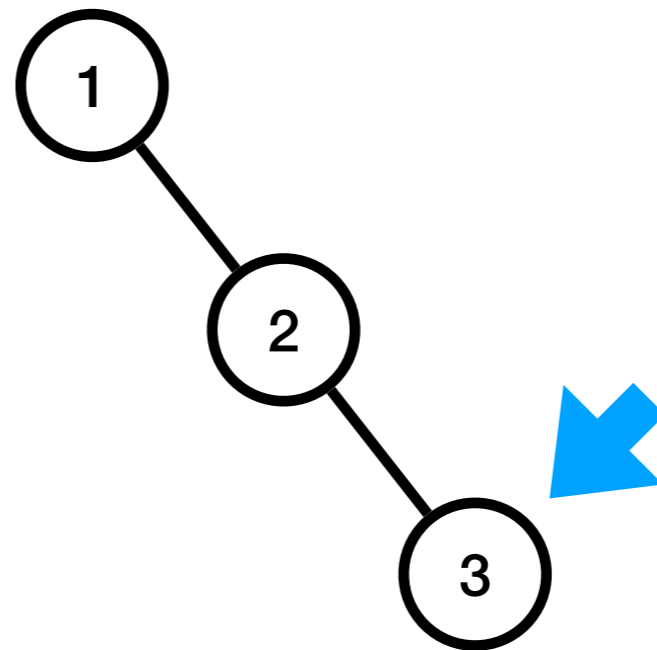
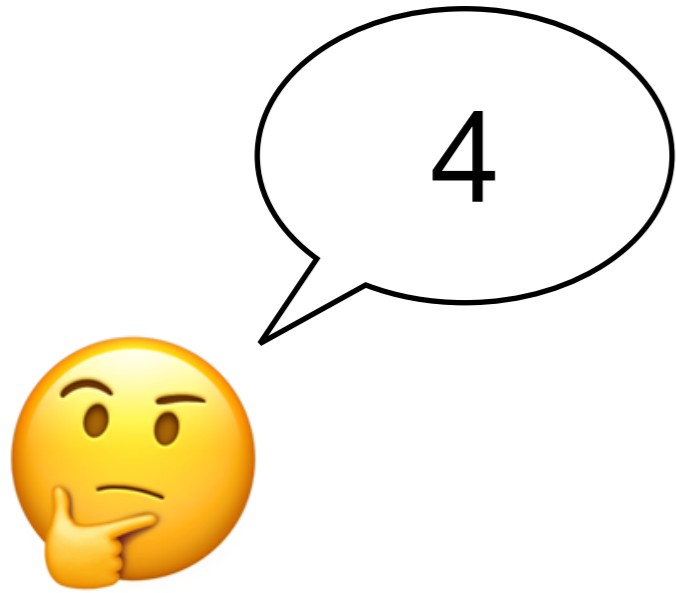
# Insertion dans un ABR



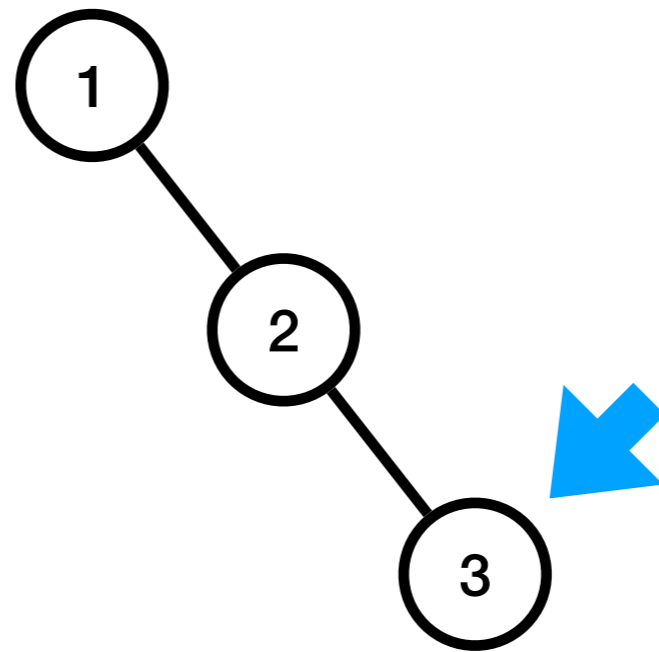
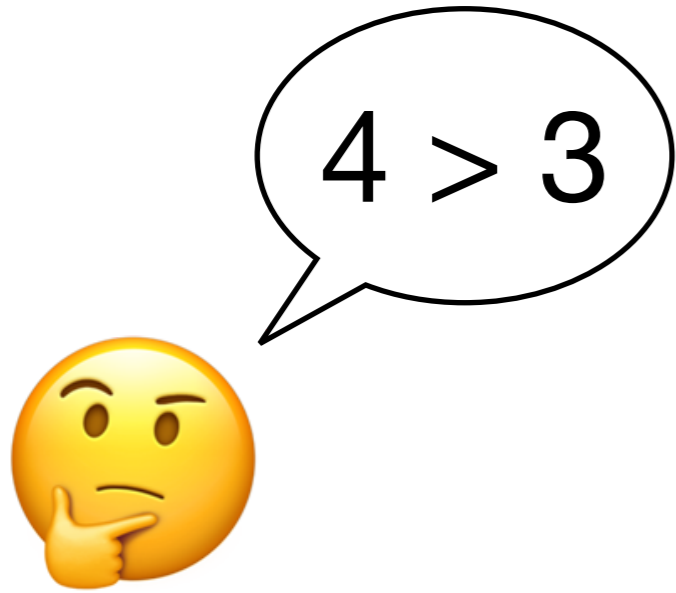
# Insertion dans un ABR



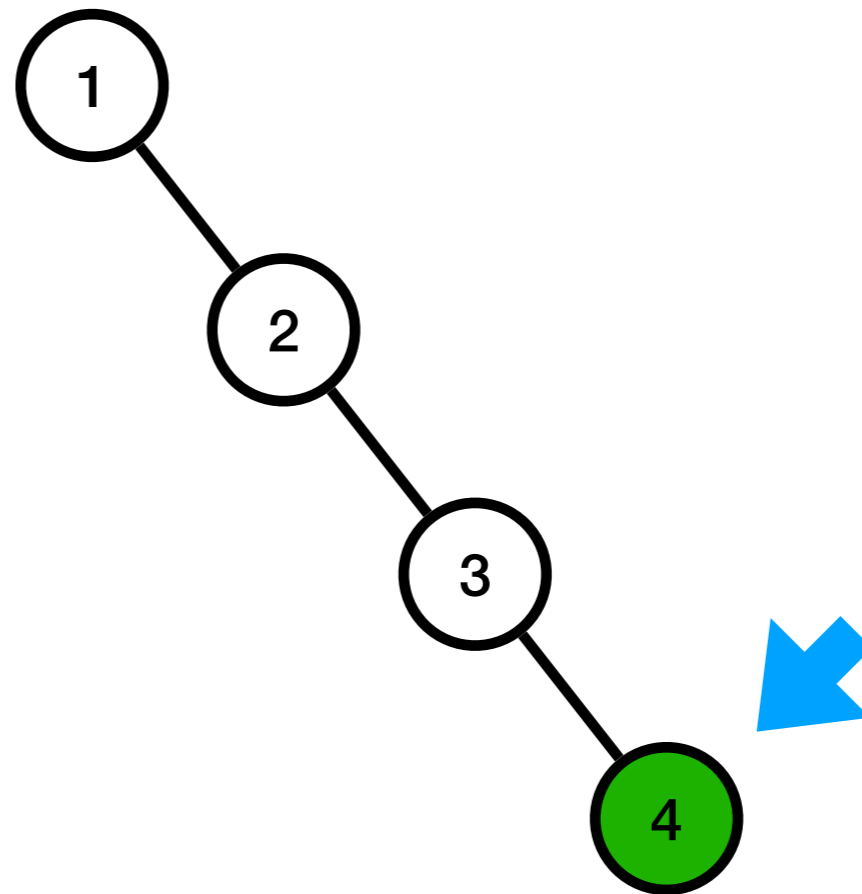
# Insertion dans un ABR



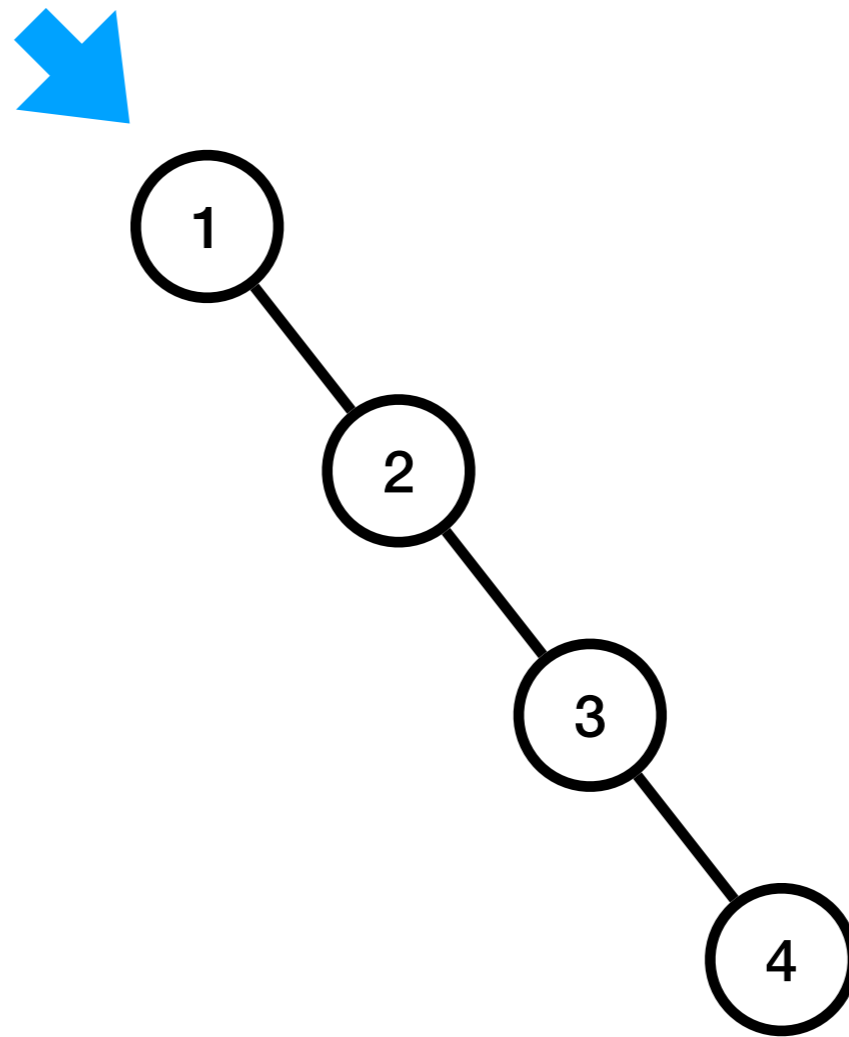
# Insertion dans un ABR



# Insertion dans un ABR

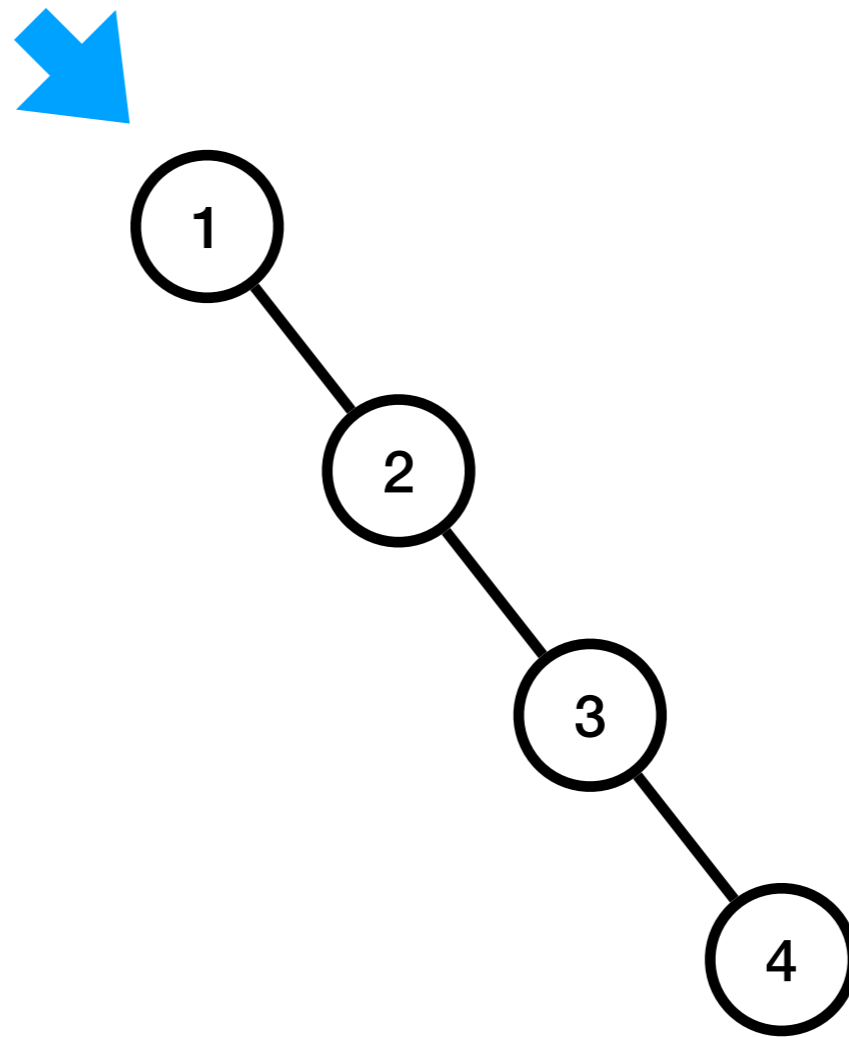
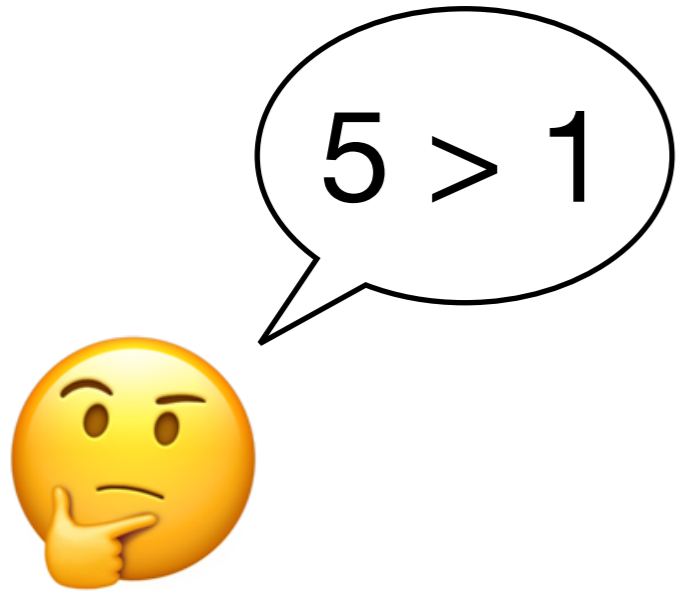


# Insertion dans un ABR

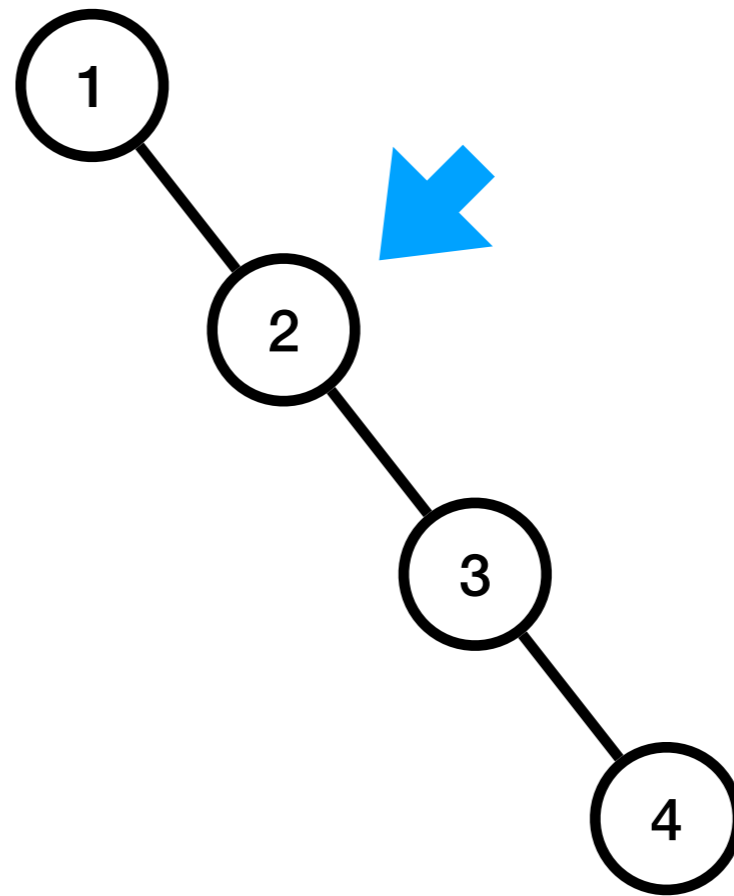




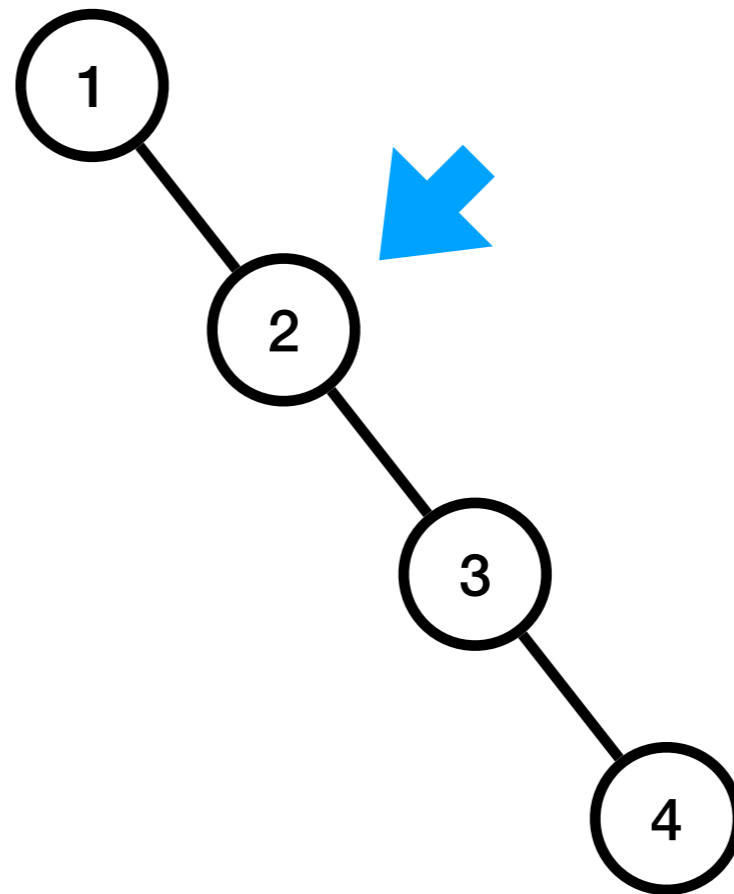
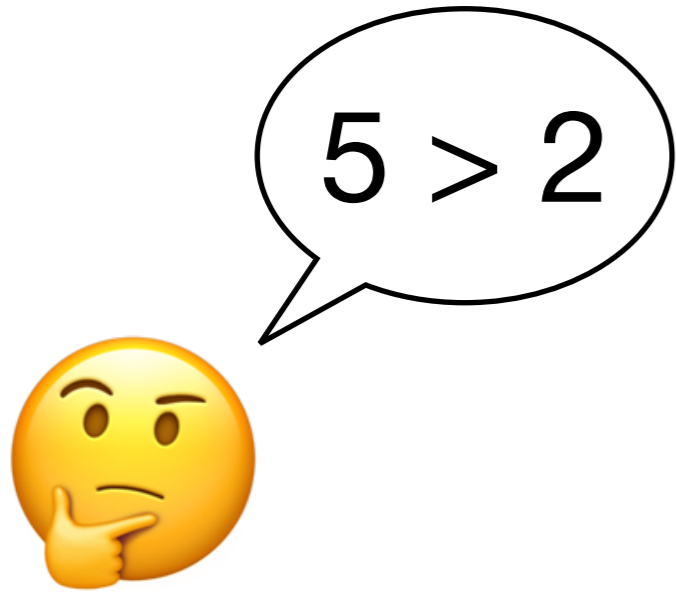
# Insertion dans un ABR



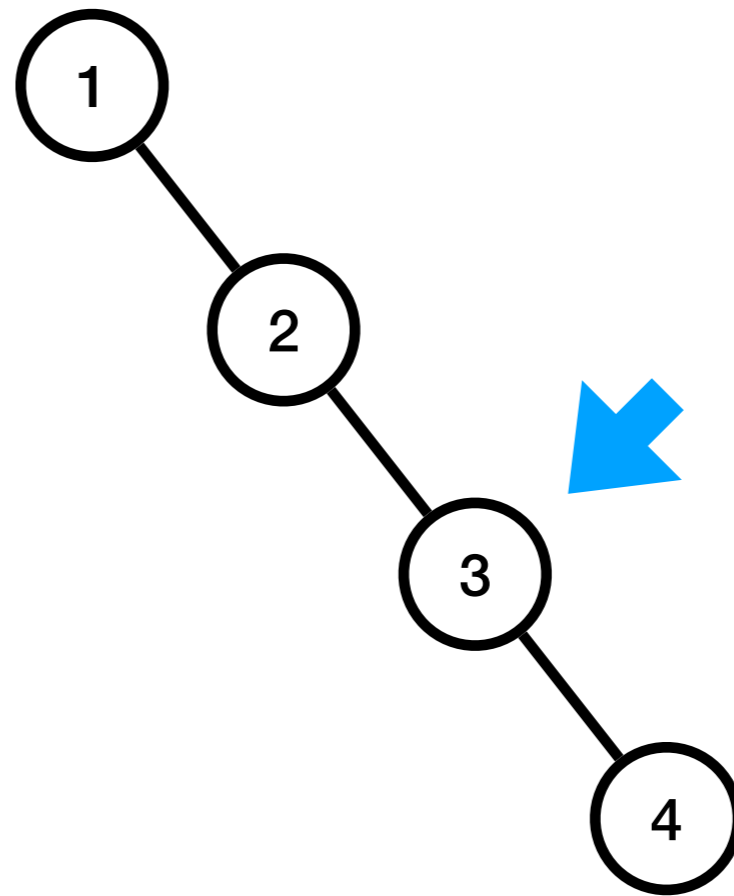
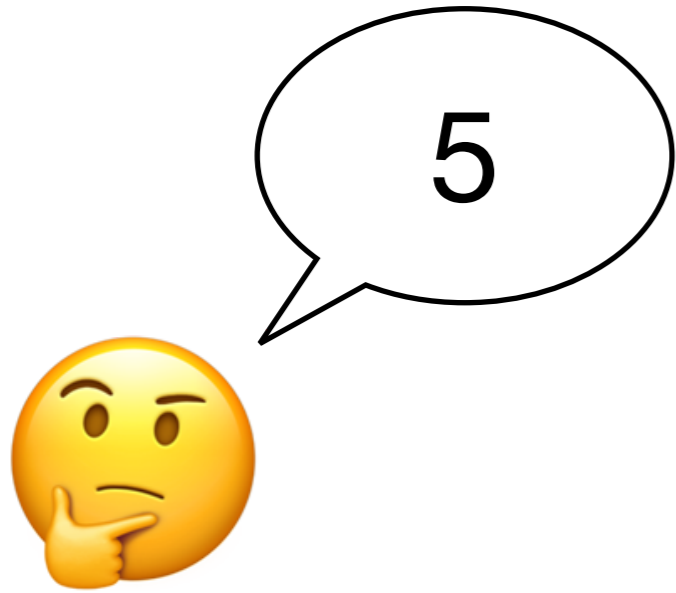
# Insertion dans un ABR



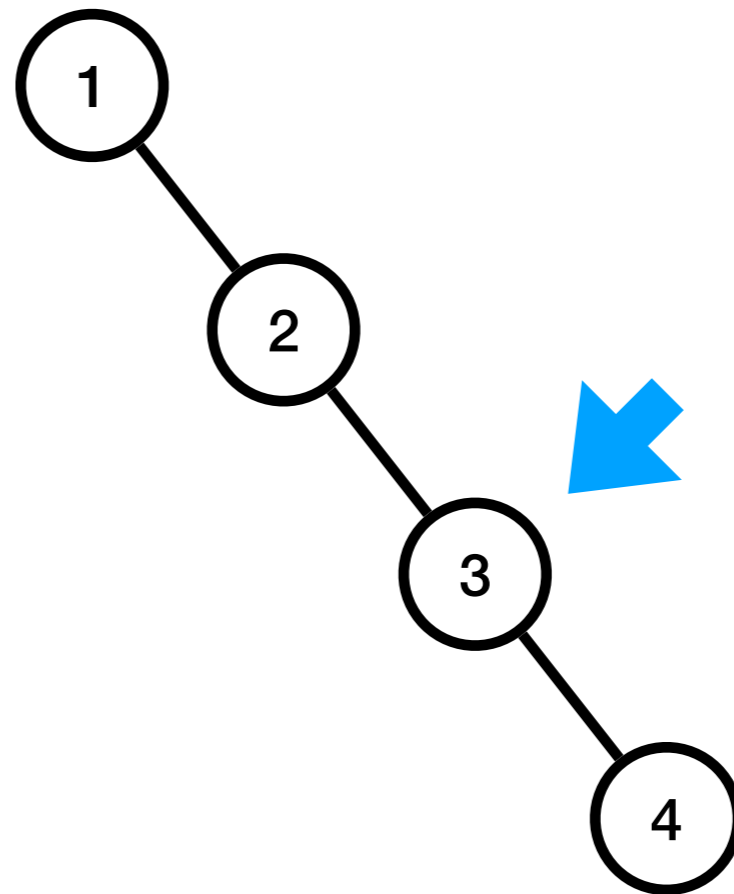
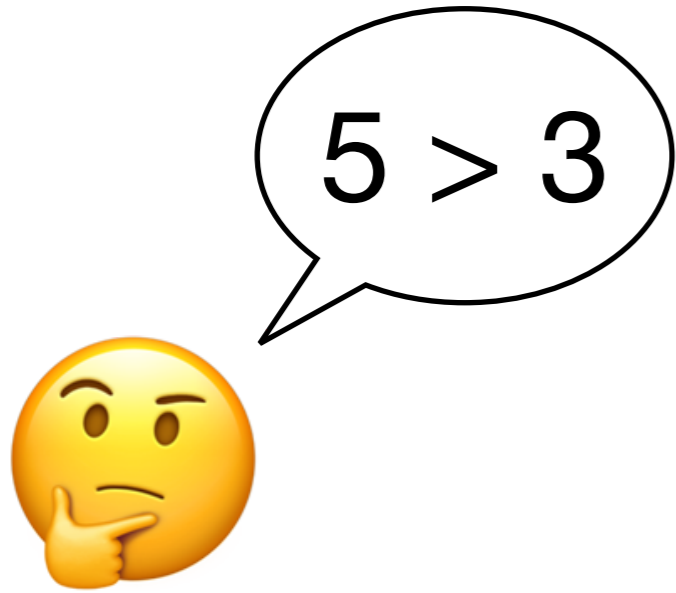
# Insertion dans un ABR



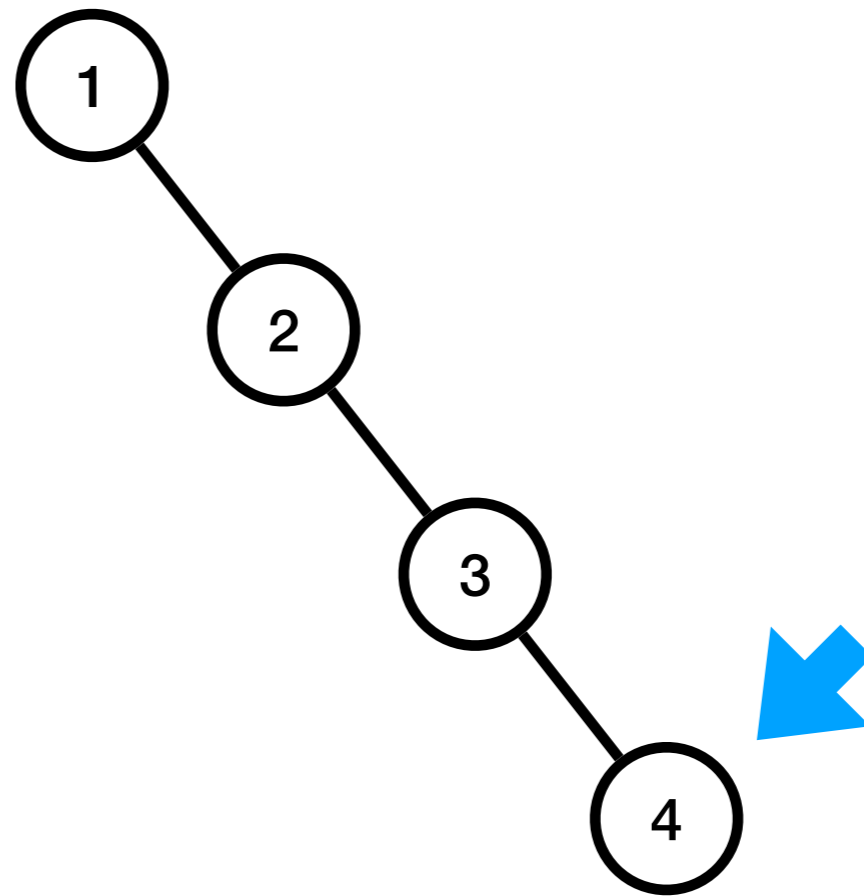
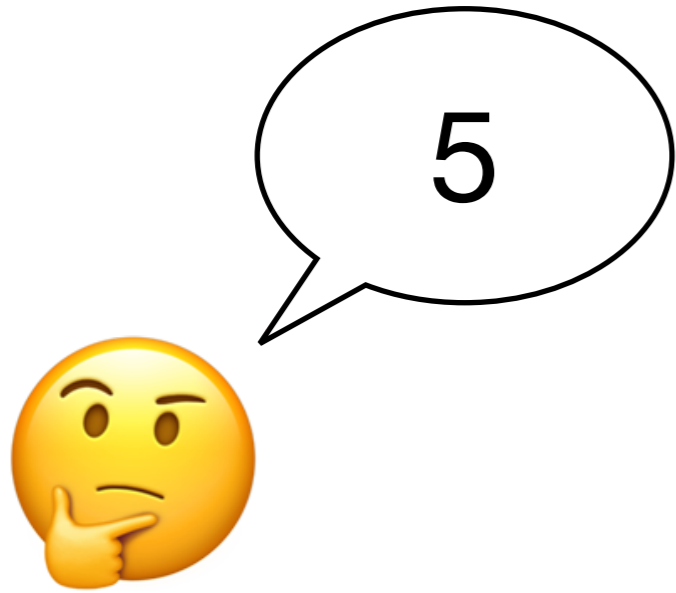
# Insertion dans un ABR



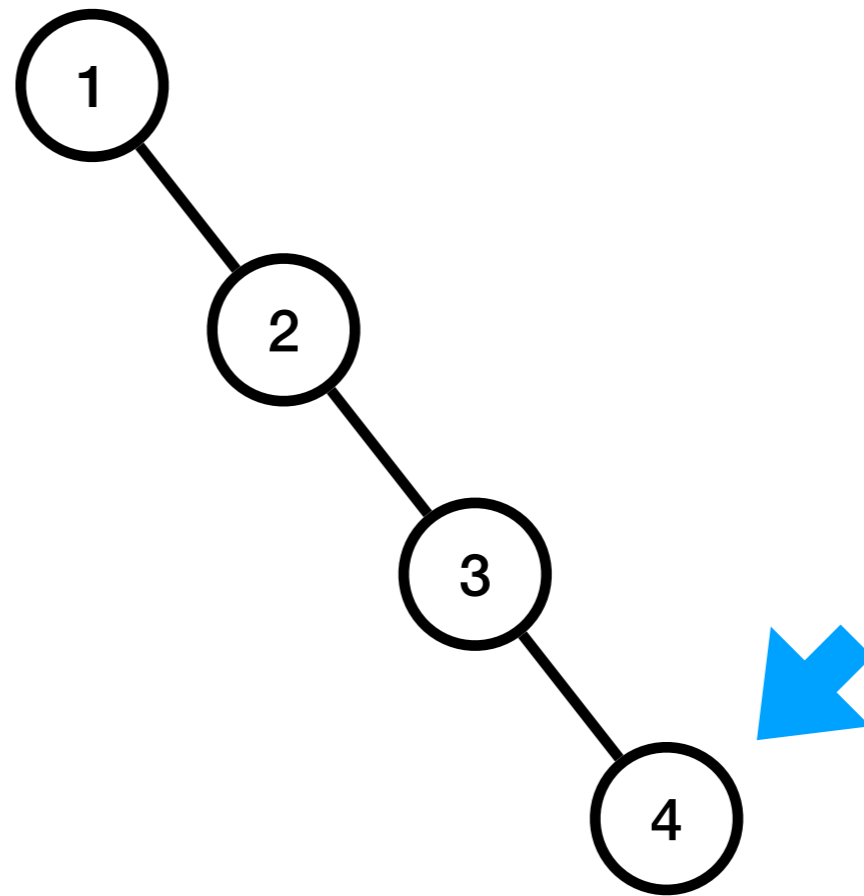
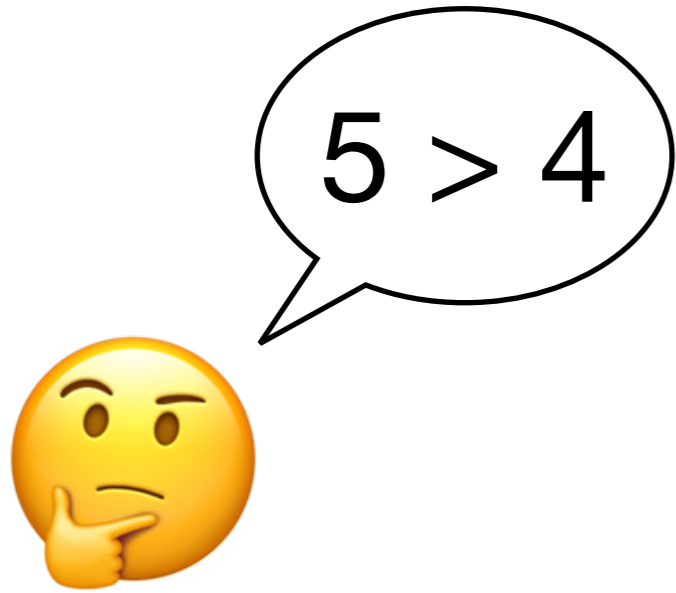
# Insertion dans un ABR



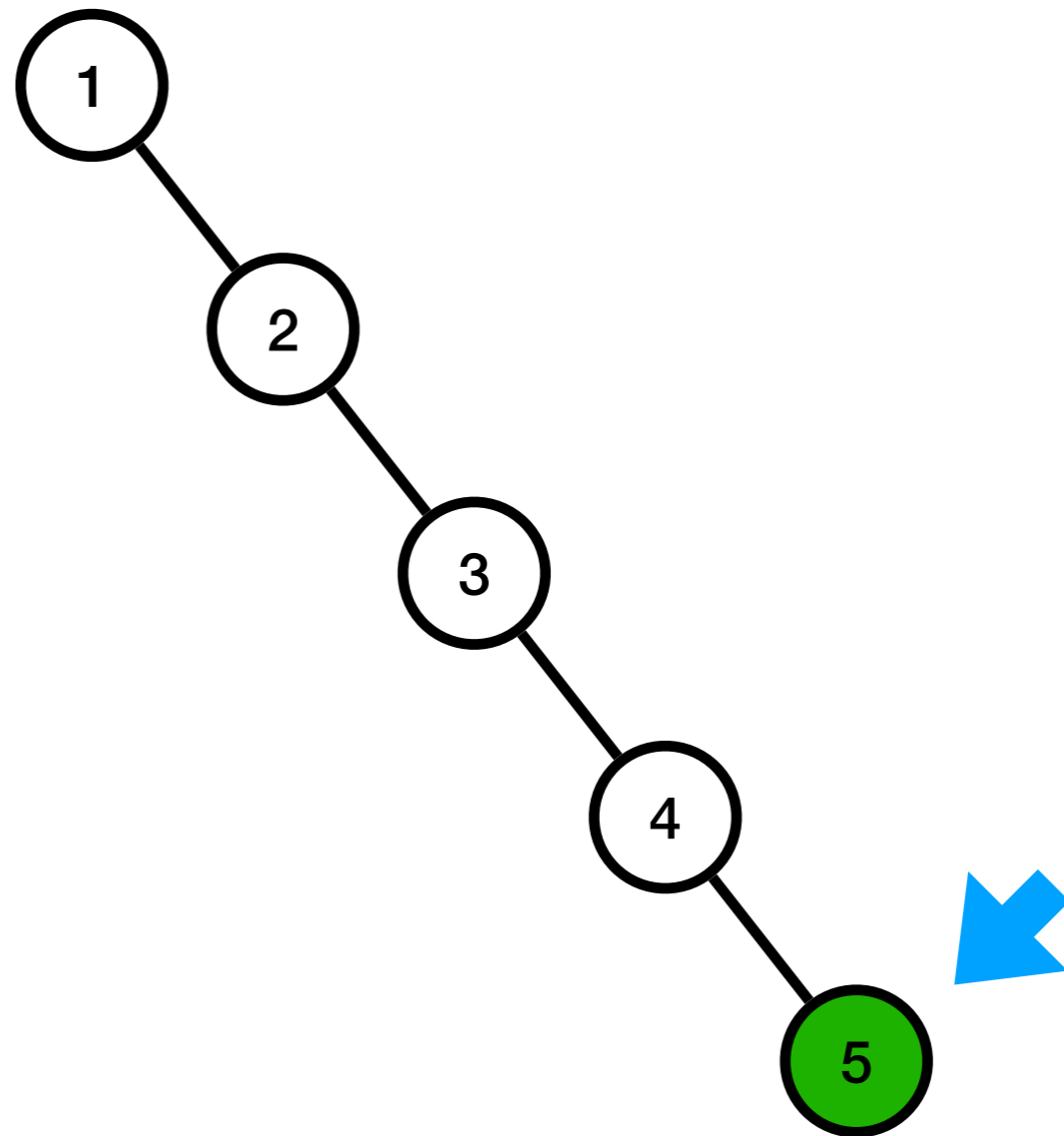
# Insertion dans un ABR



# Insertion dans un ABR

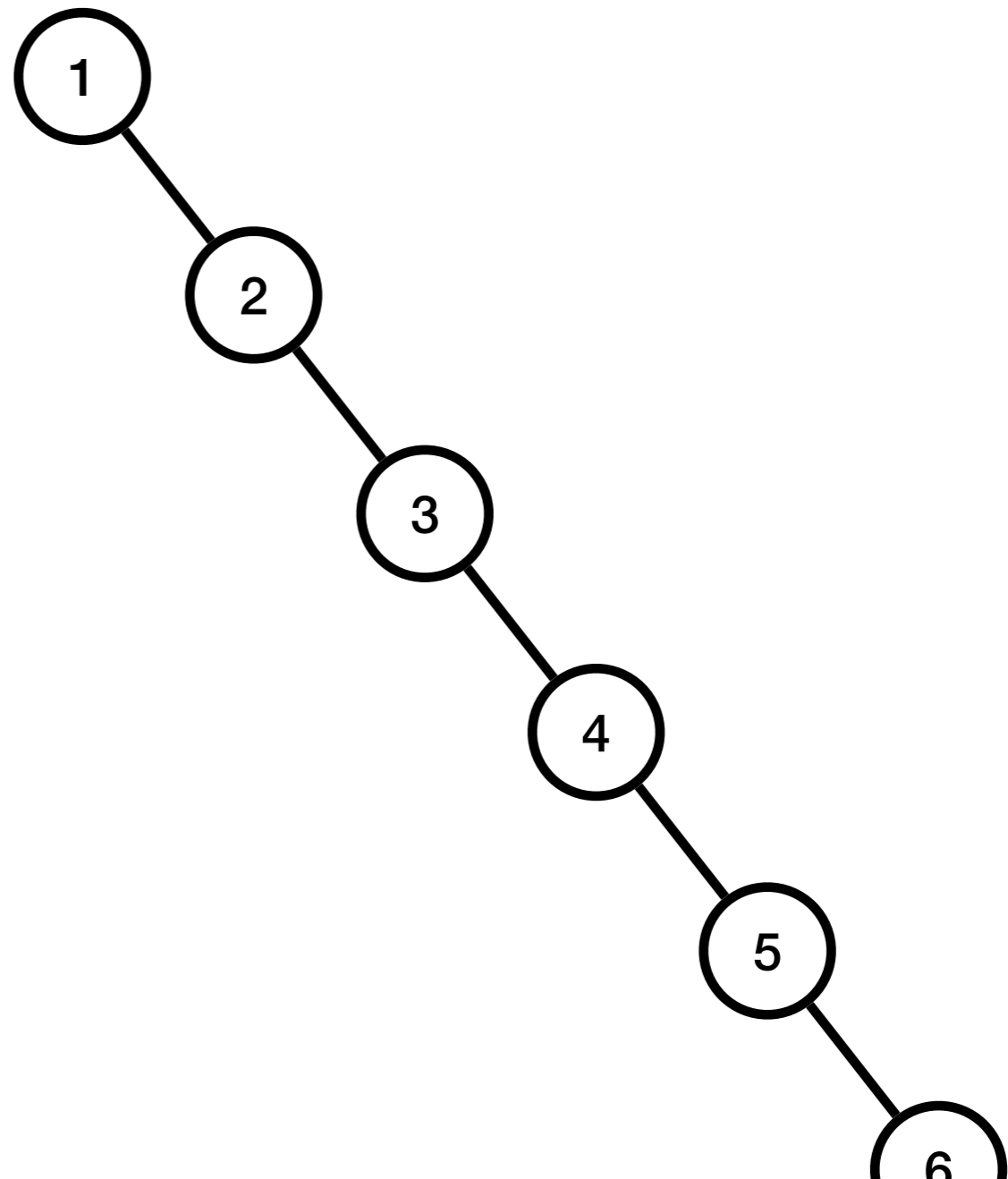


# Insertion dans un ABR





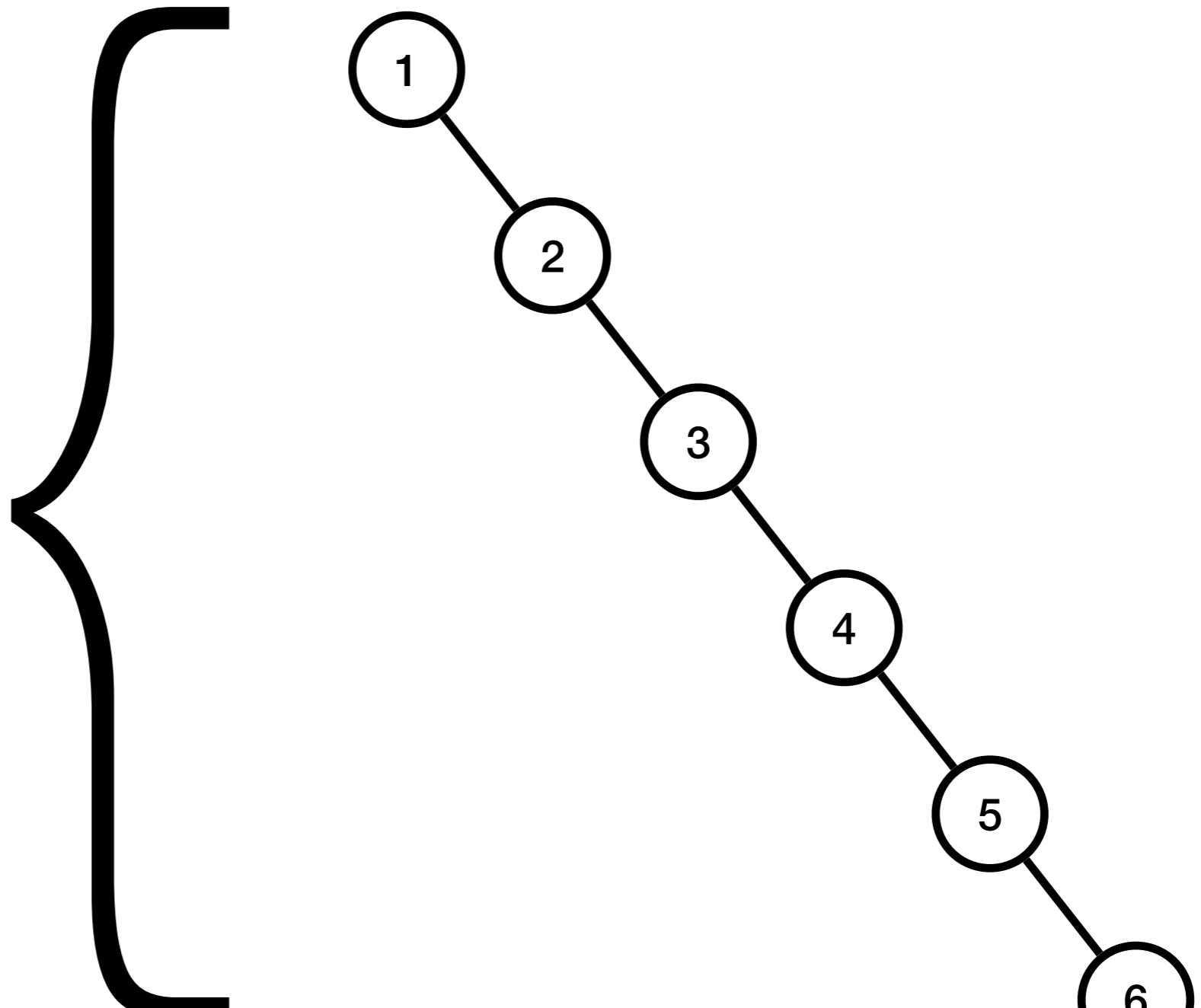
# Insertion dans un ABR



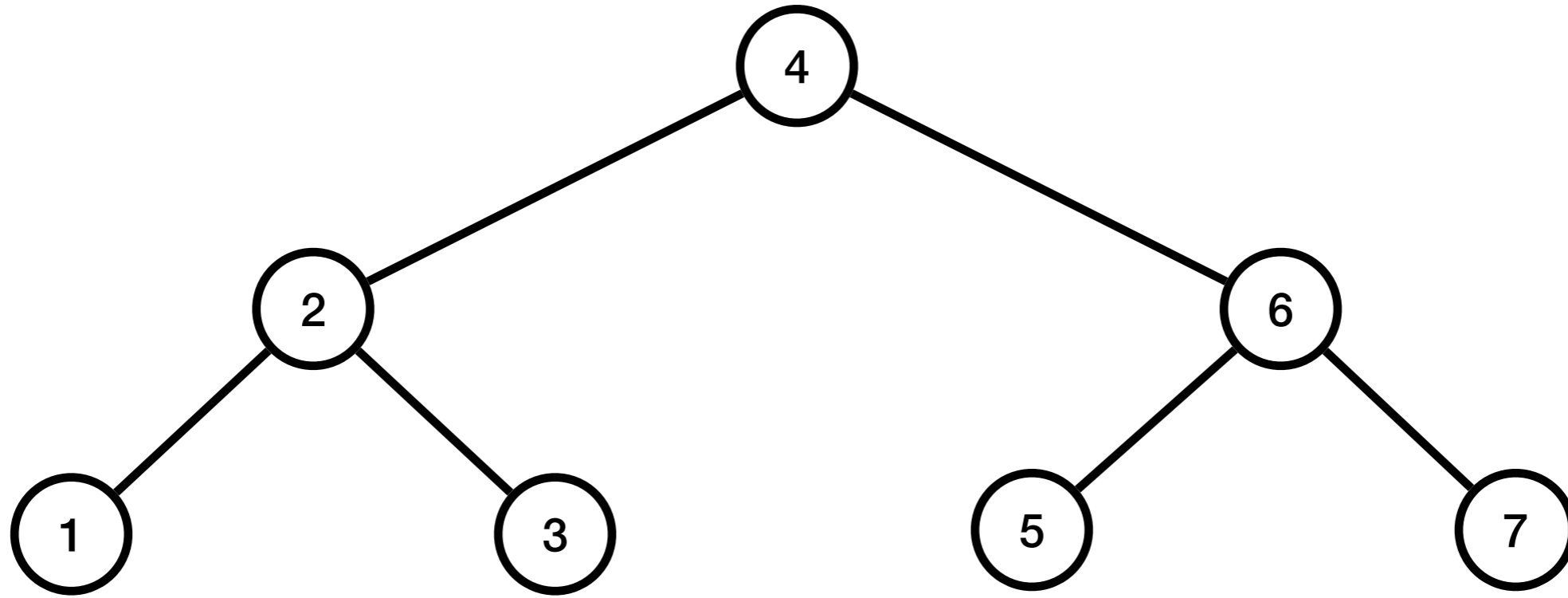
# Insertion dans un ABR



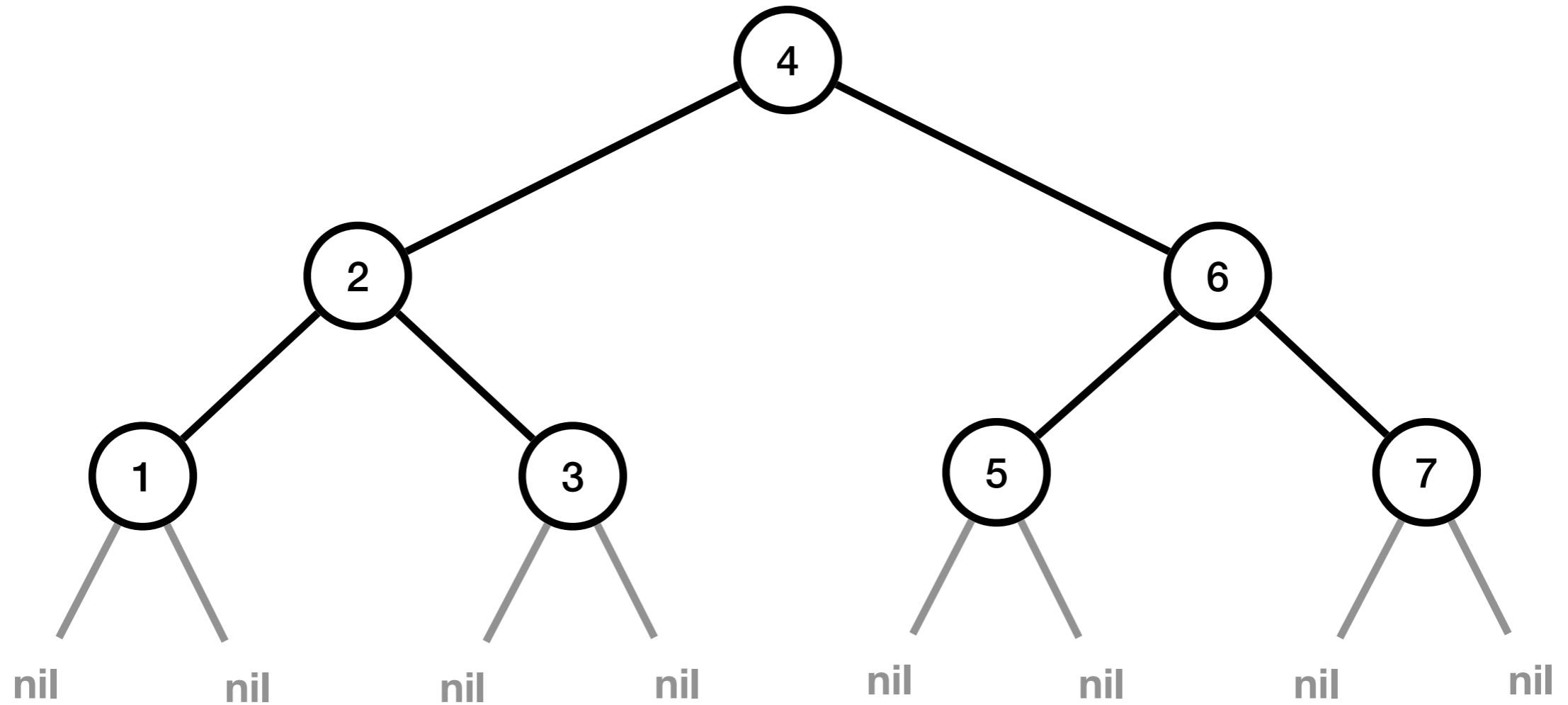
$$h = n - 1$$



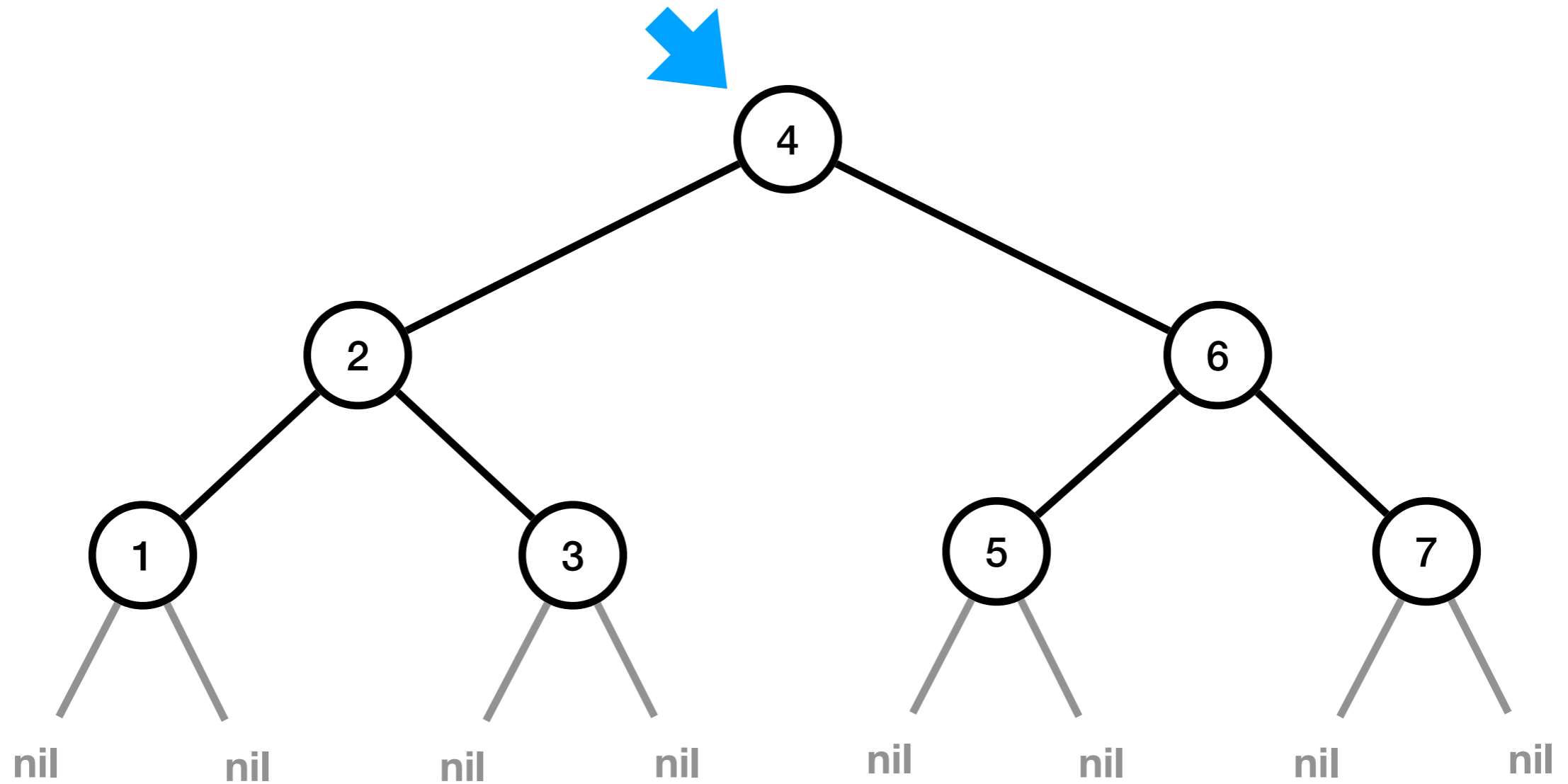
# Parcours infixe d'ABR



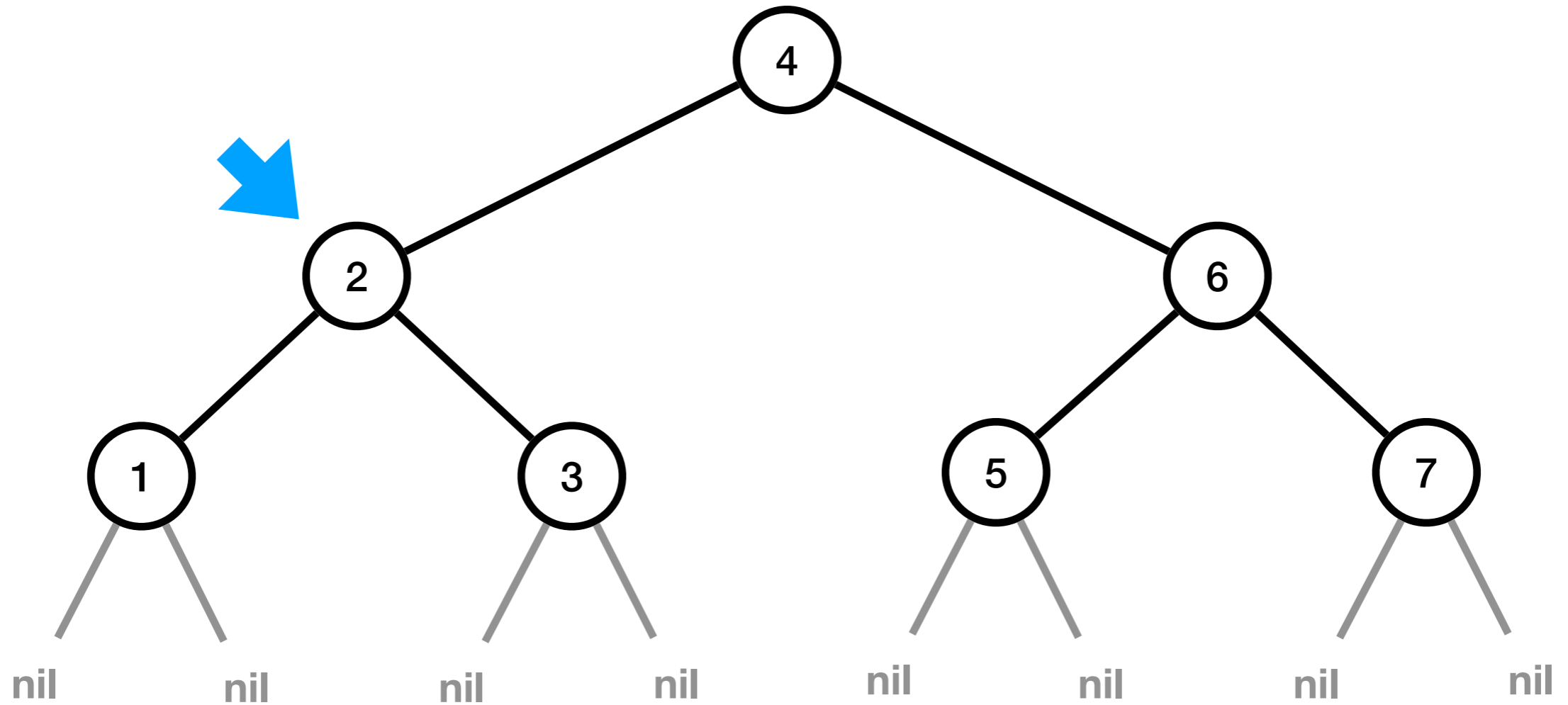
# Parcours infixe d'ABR



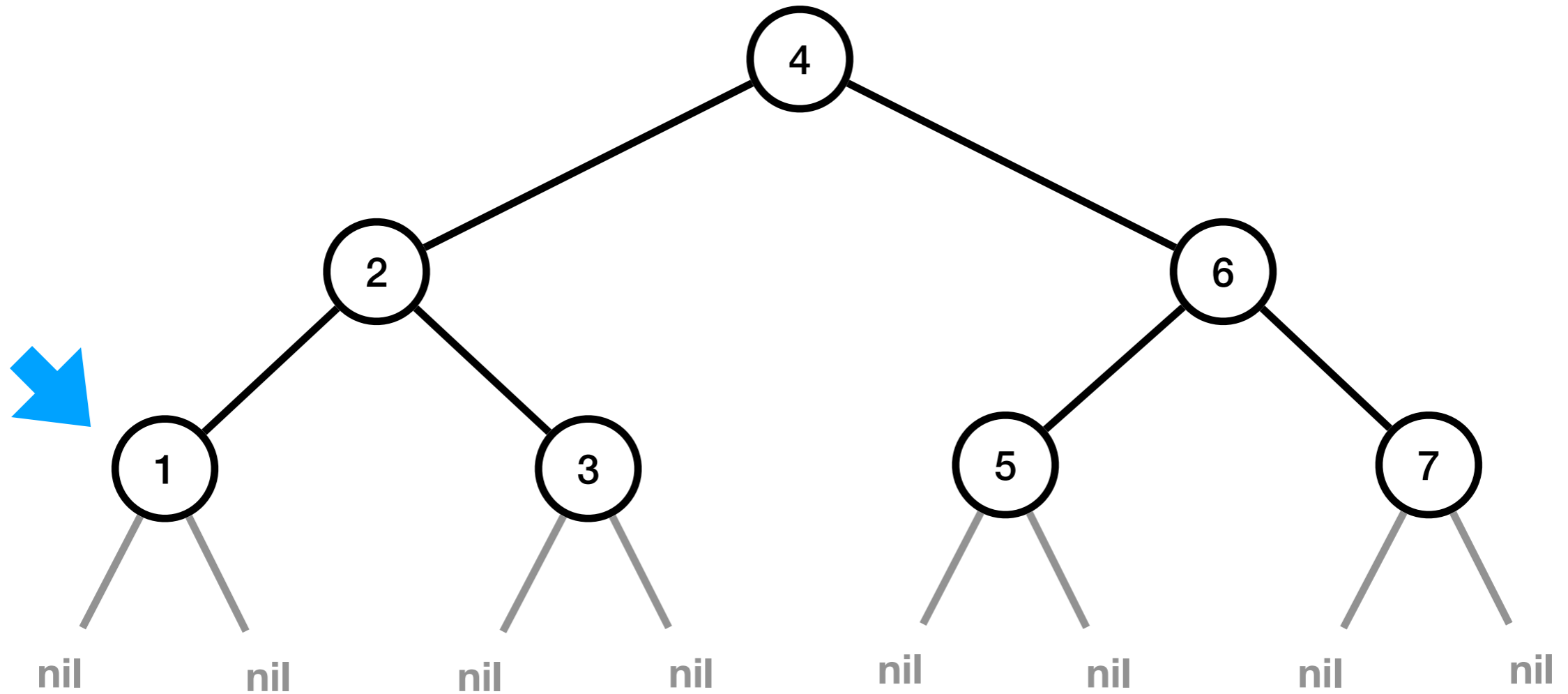
# Parcours infixe d'ABR



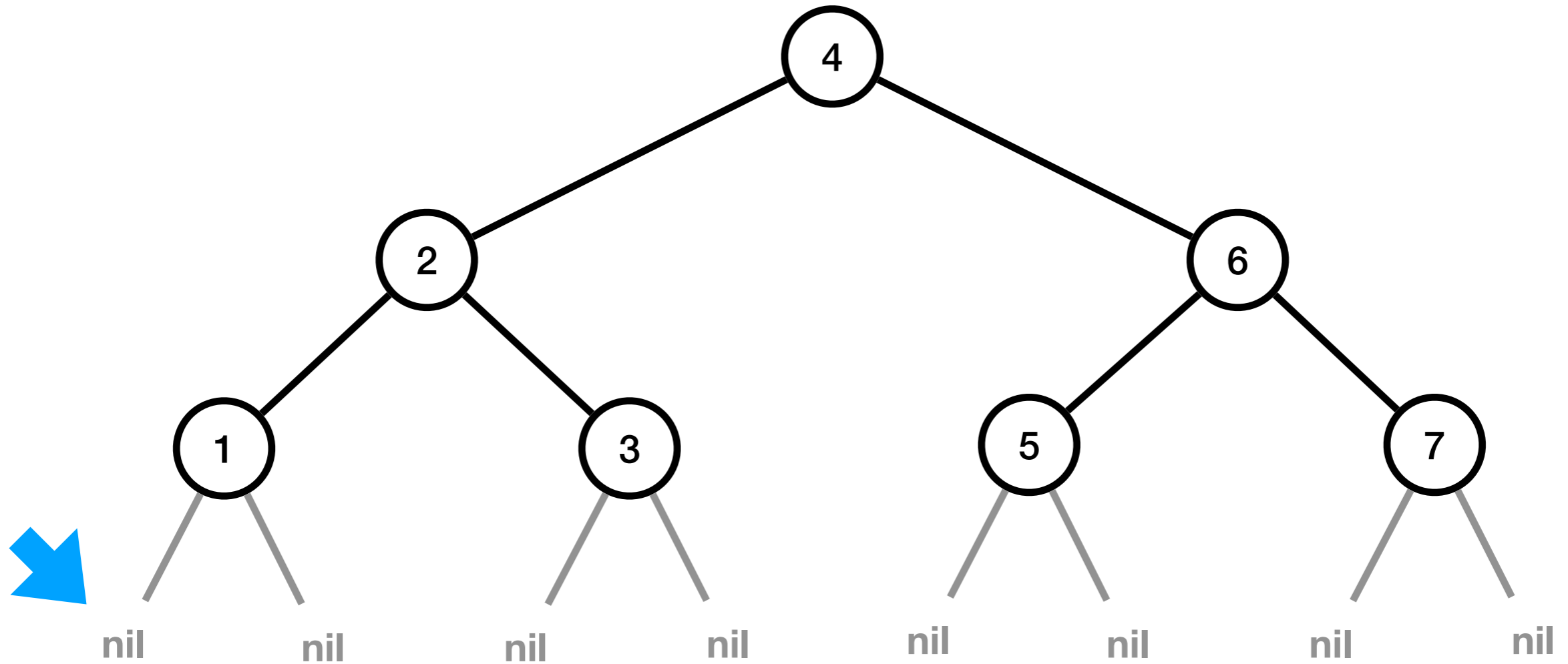
# Parcours infixe d'ABR



# Parcours infixe d'ABR

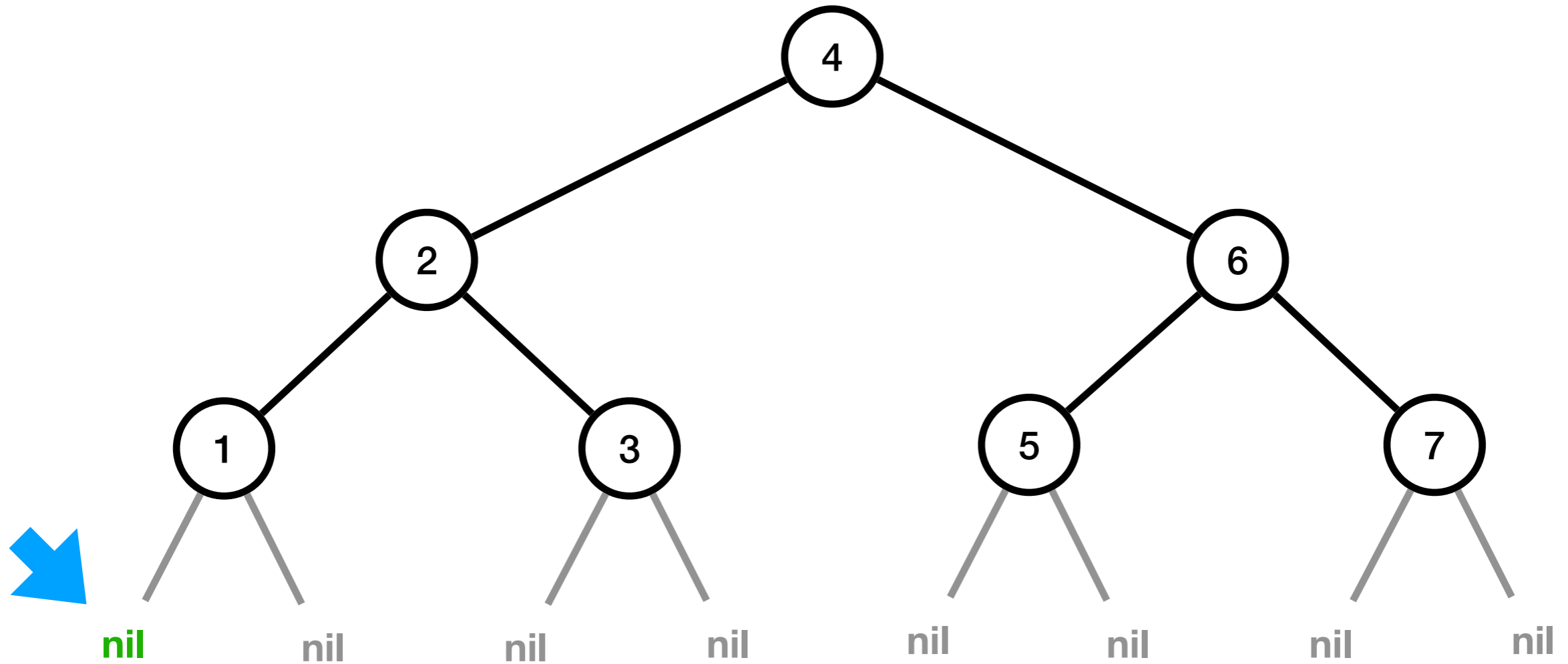


# Parcours infixe d'ABR

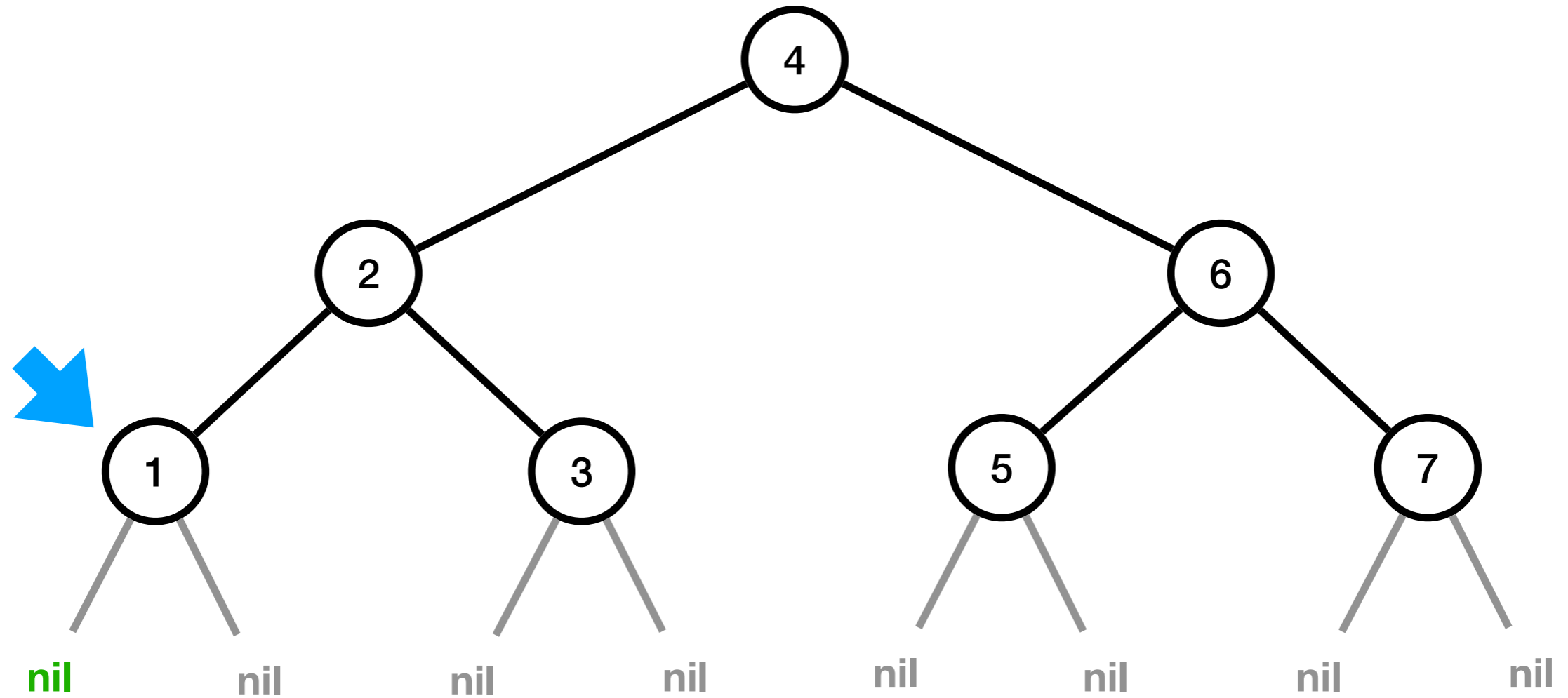




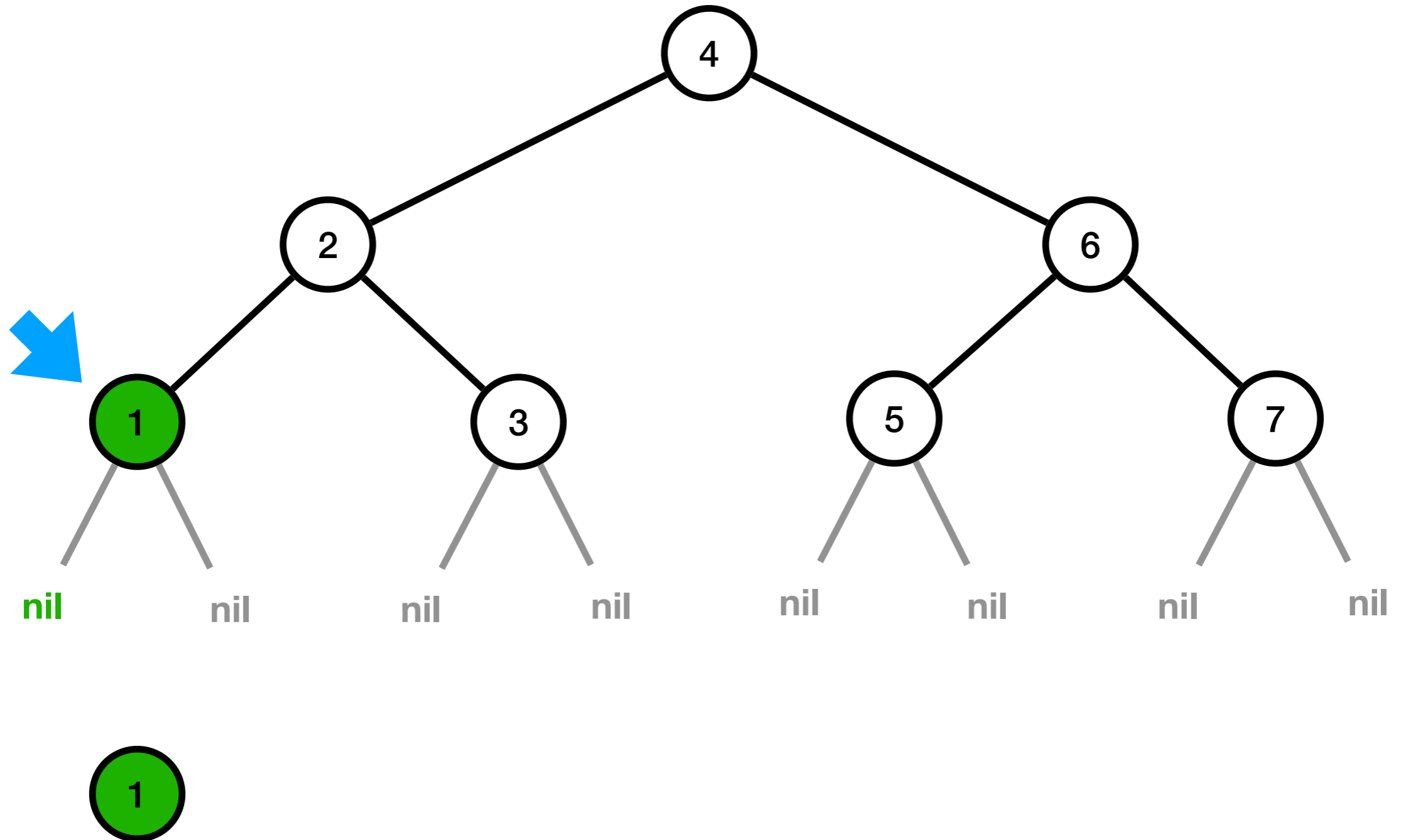
# Parcours infixe d'ABR



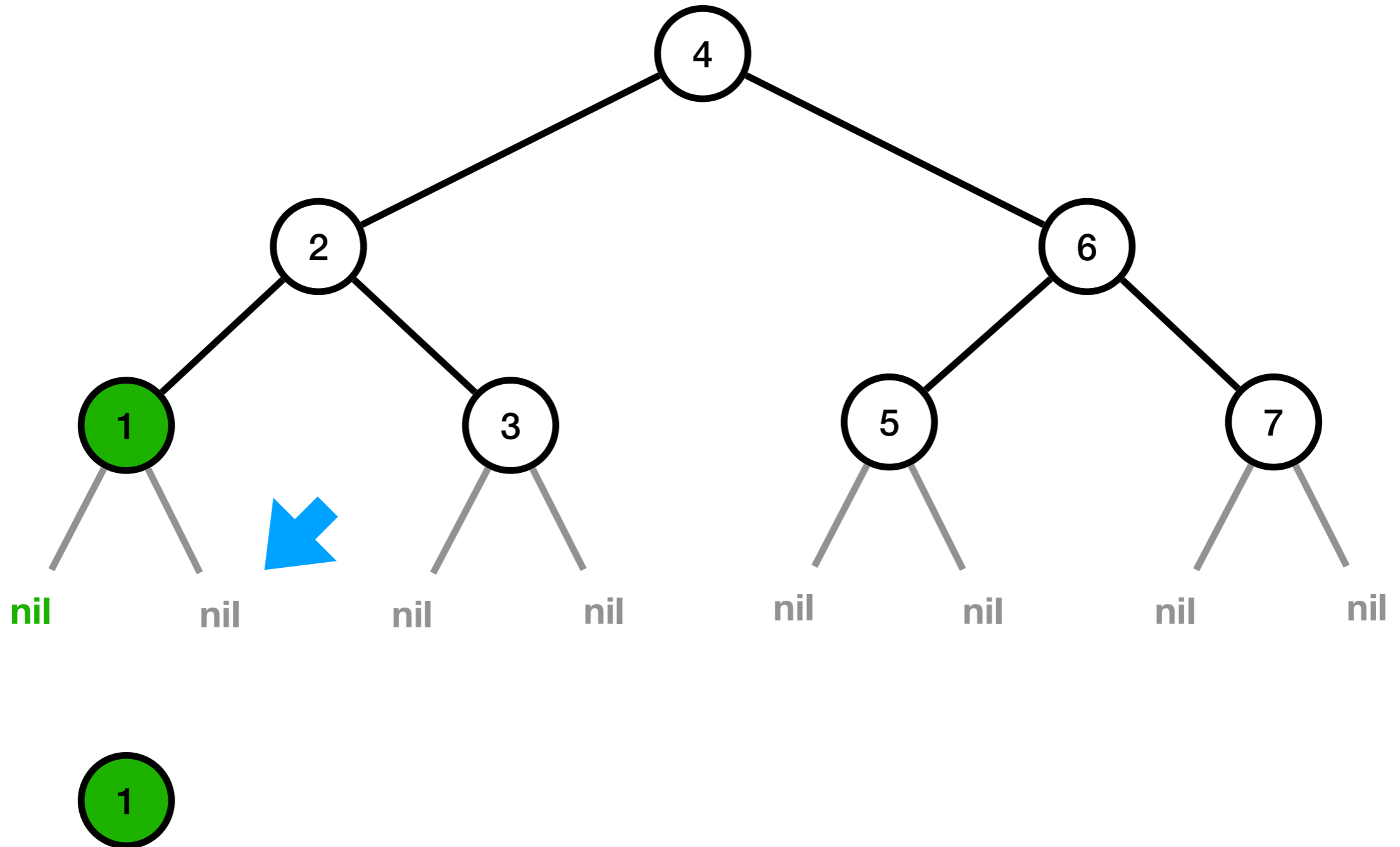
# Parcours infixe d'ABR



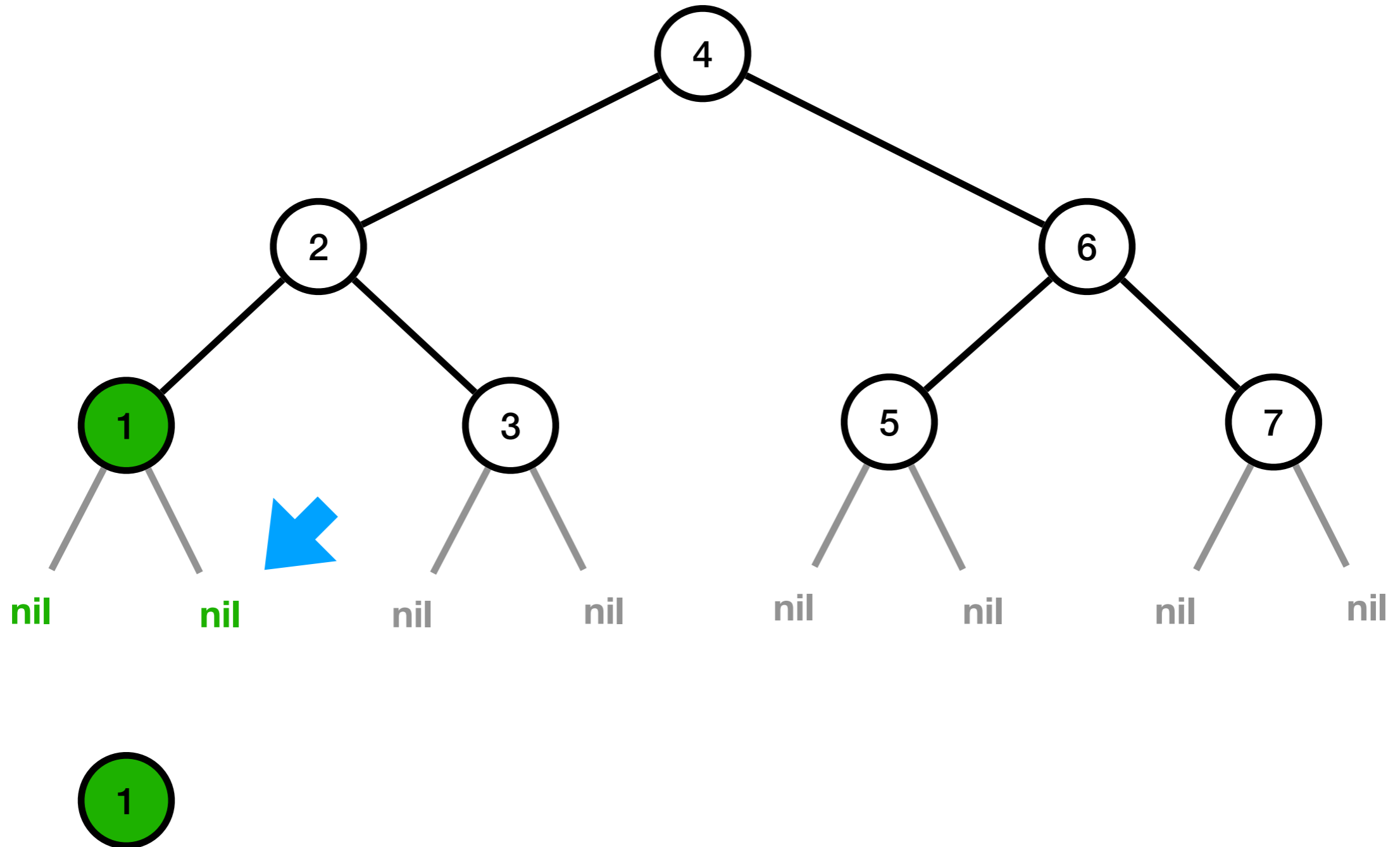
# Parcours infixe d'ABR



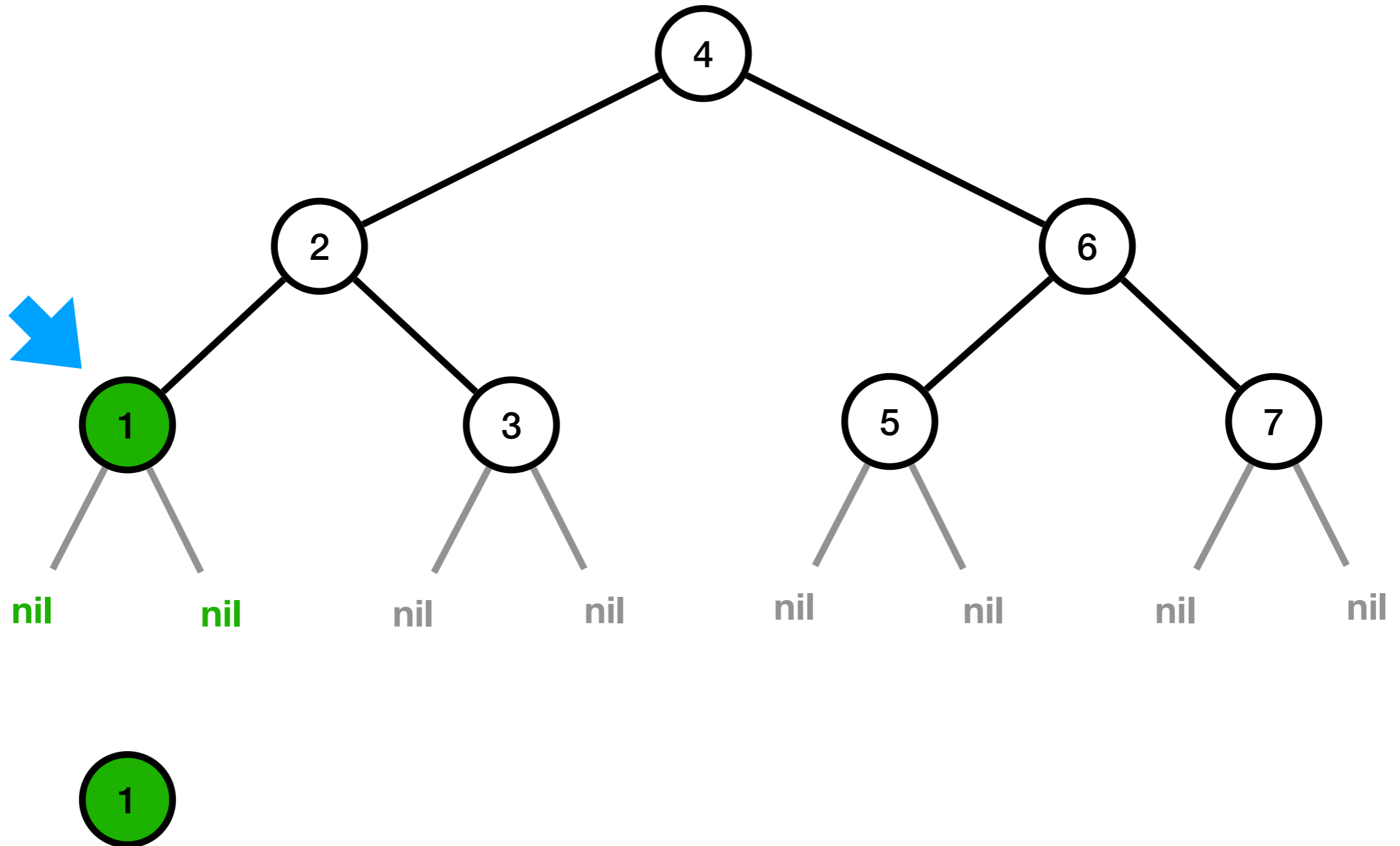
# Parcours infixe d'ABR



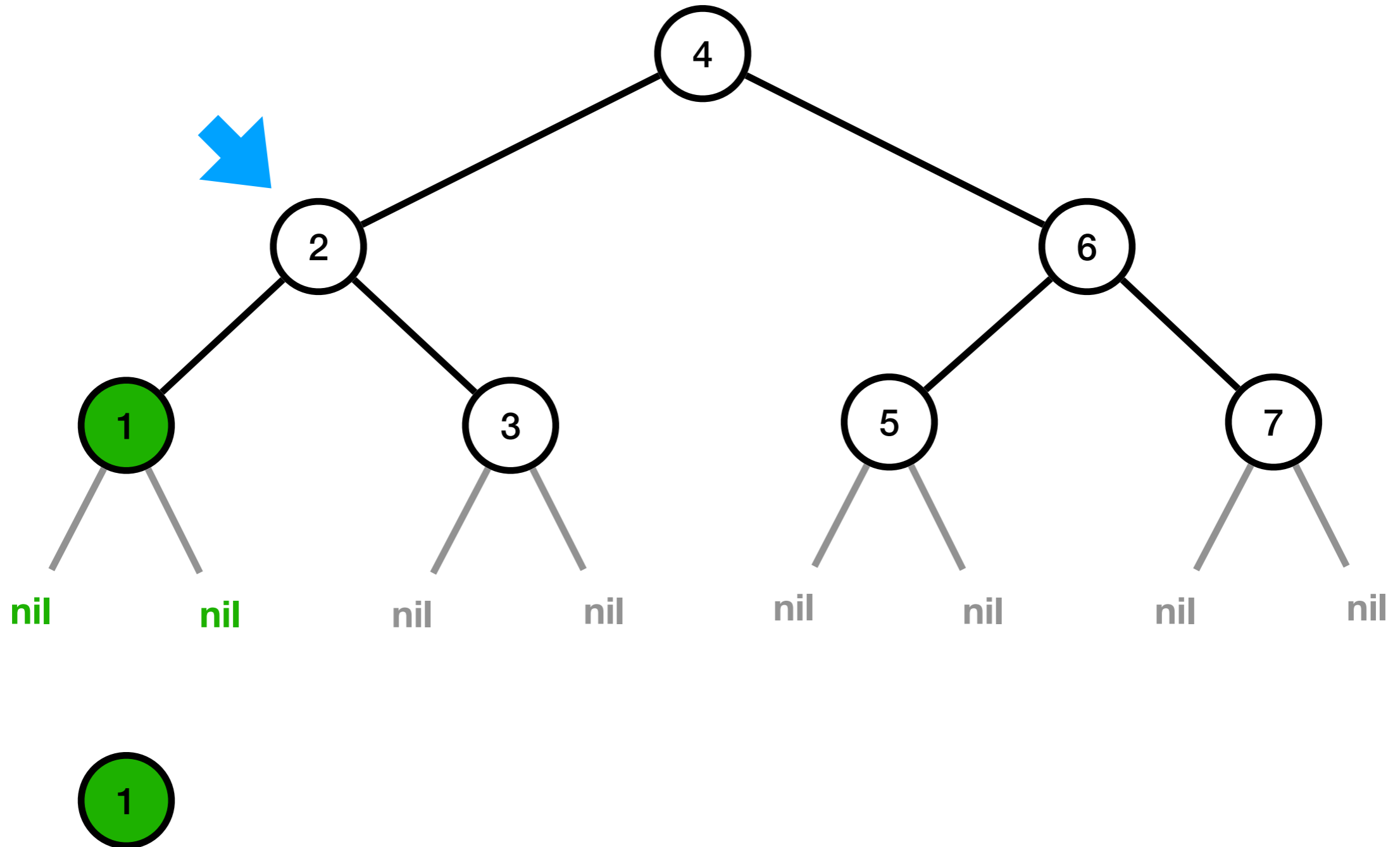
# Parcours infixe d'ABR



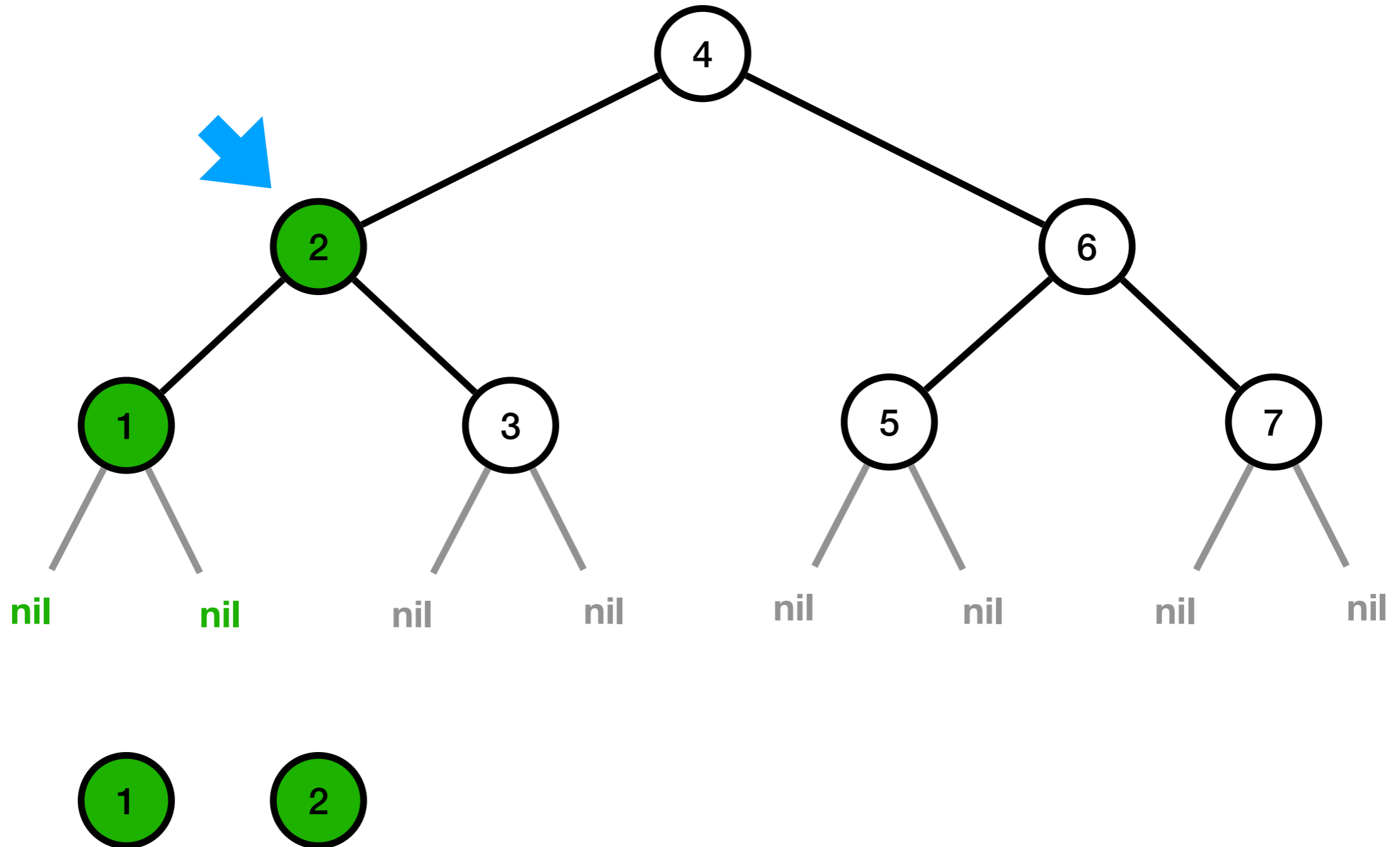
# Parcours infixe d'ABR



# Parcours infixe d'ABR

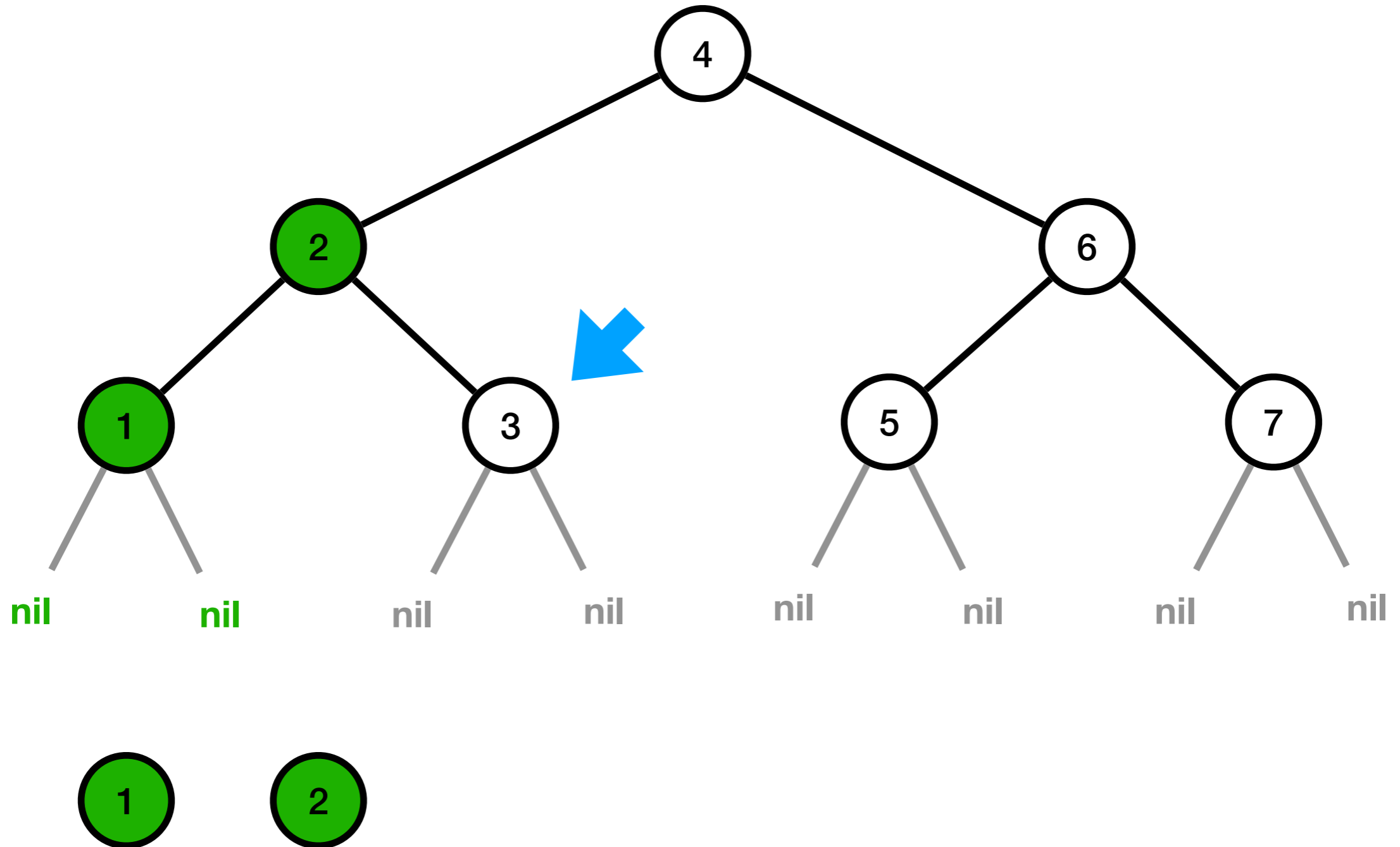


# Parcours infixe d'ABR

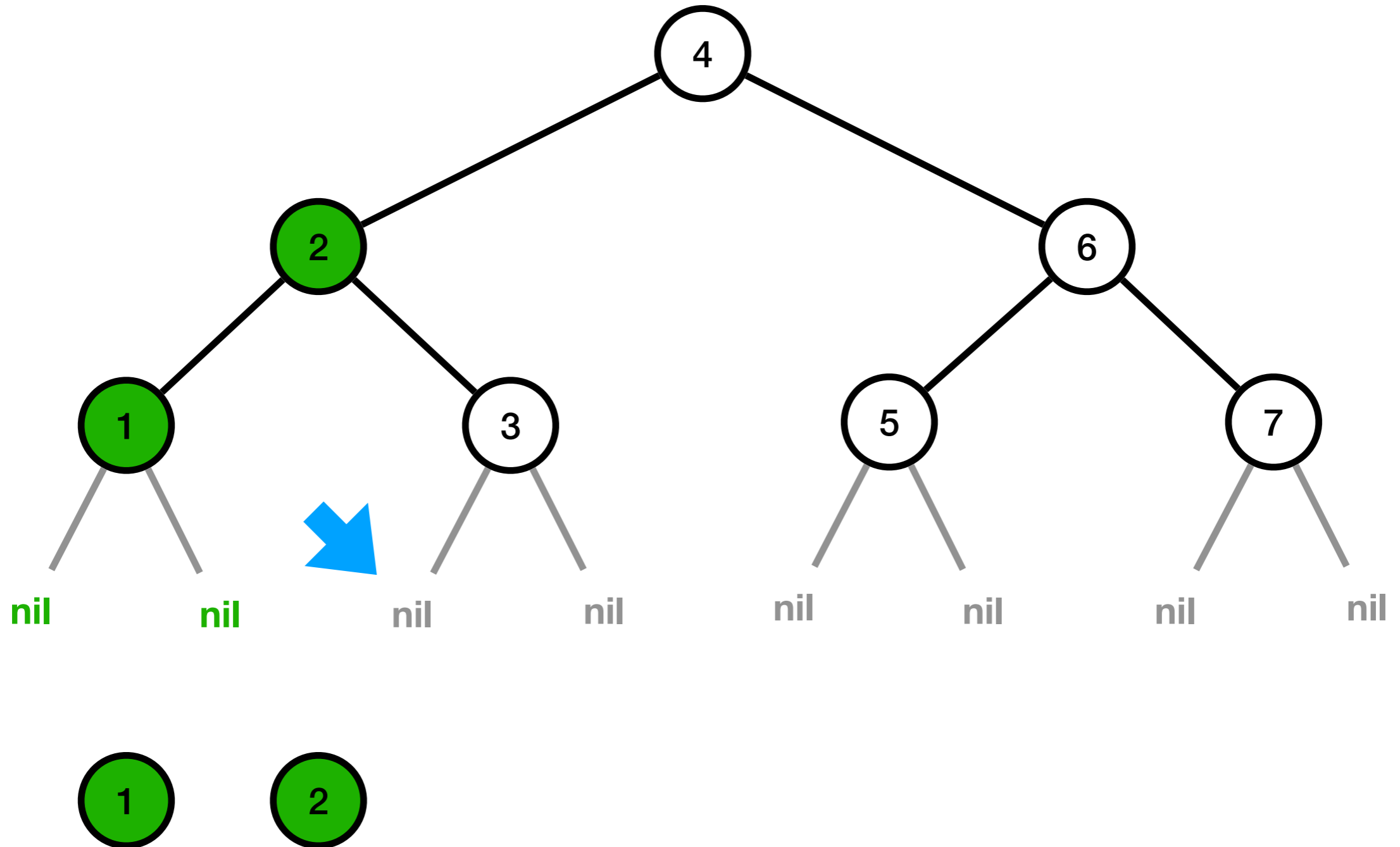




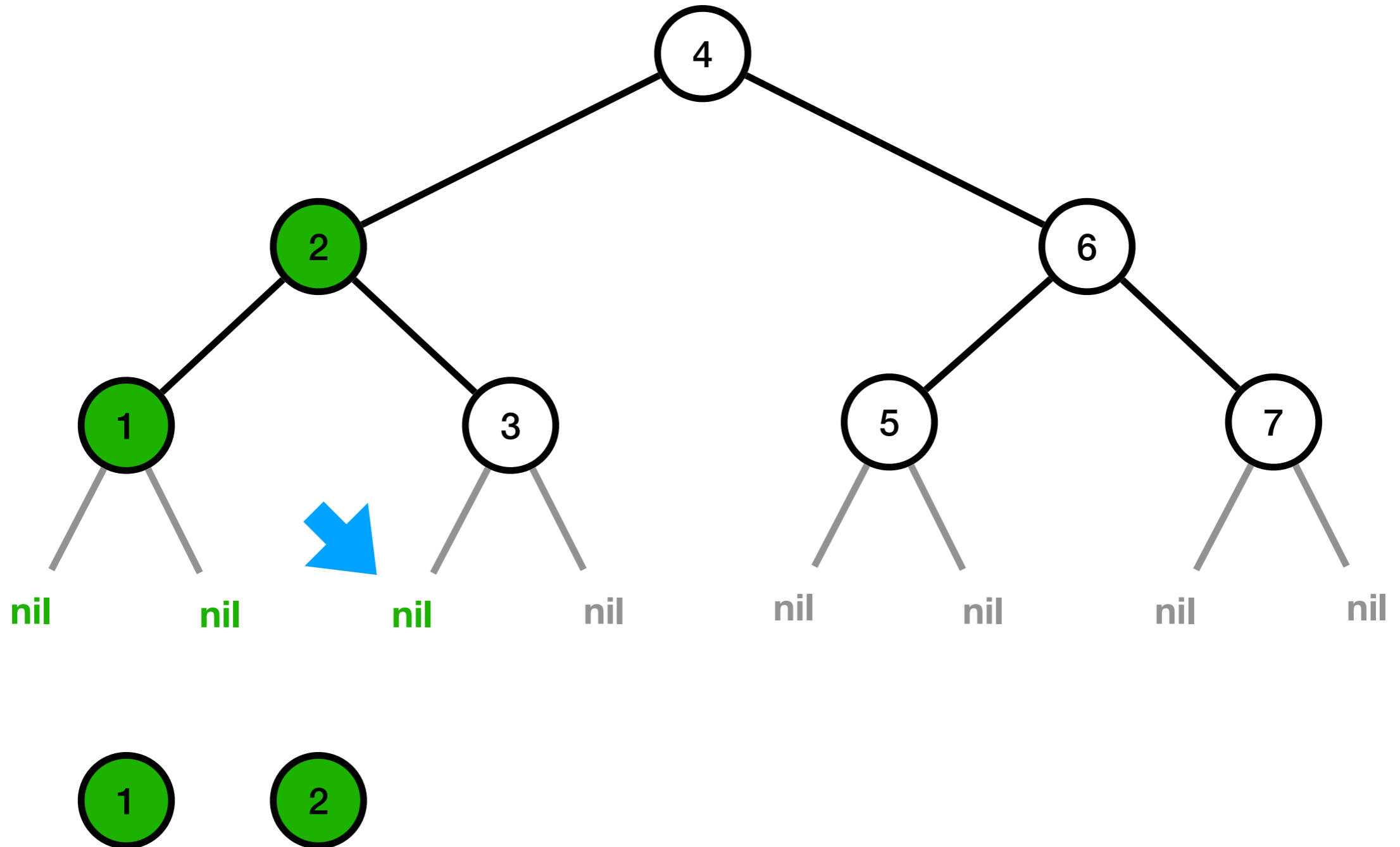
# Parcours infixe d'ABR



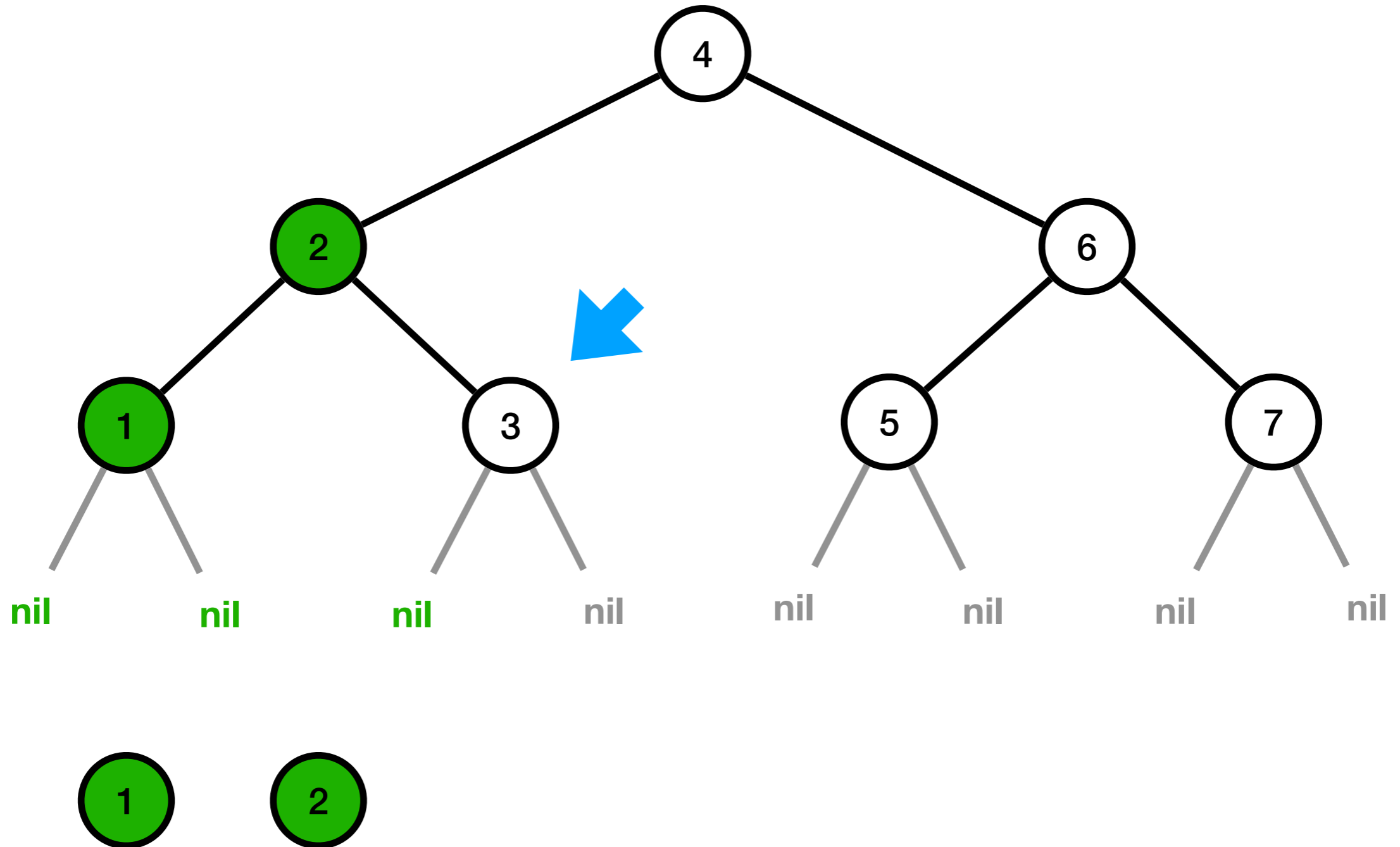
# Parcours infixe d'ABR



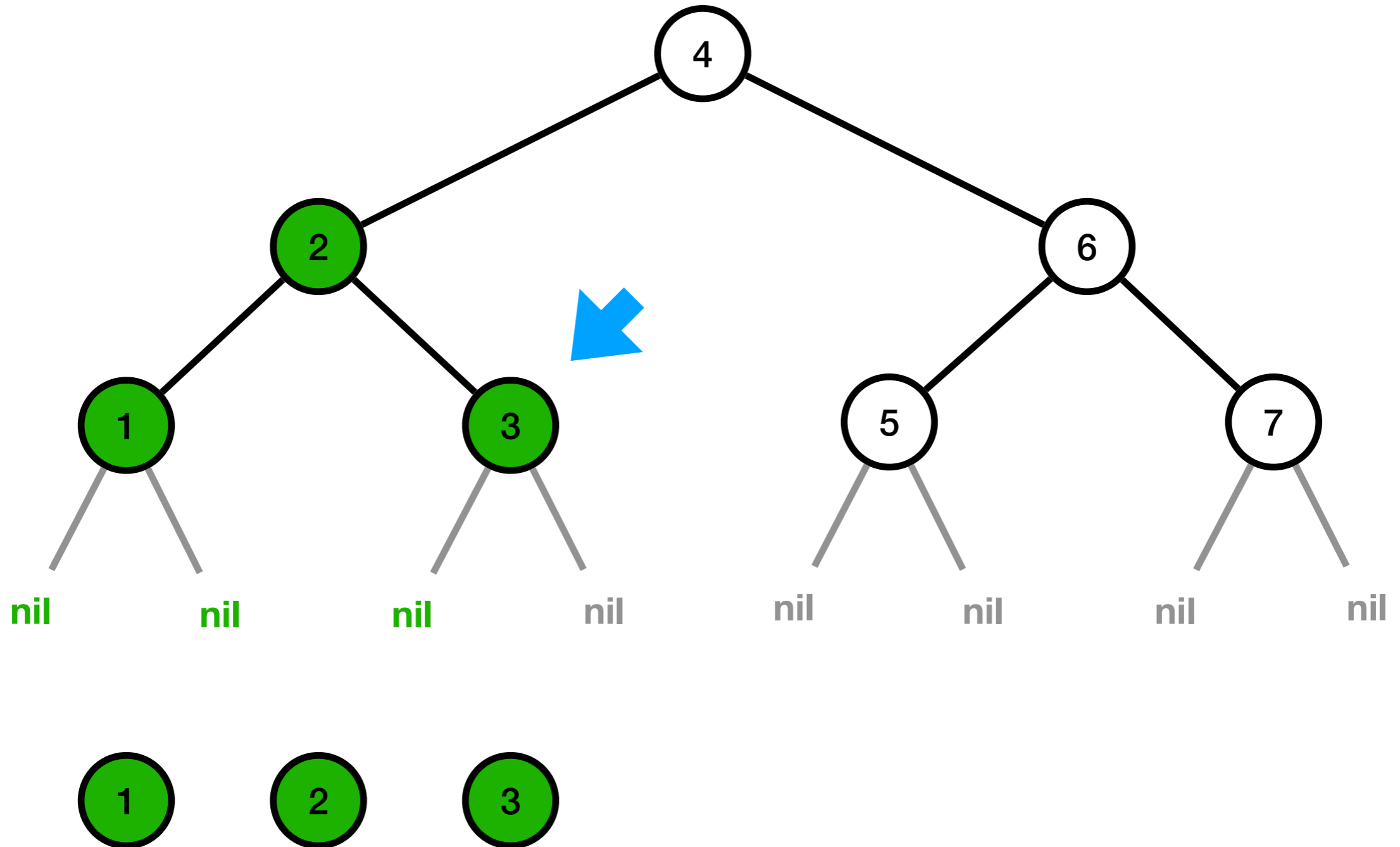
# Parcours infixe d'ABR



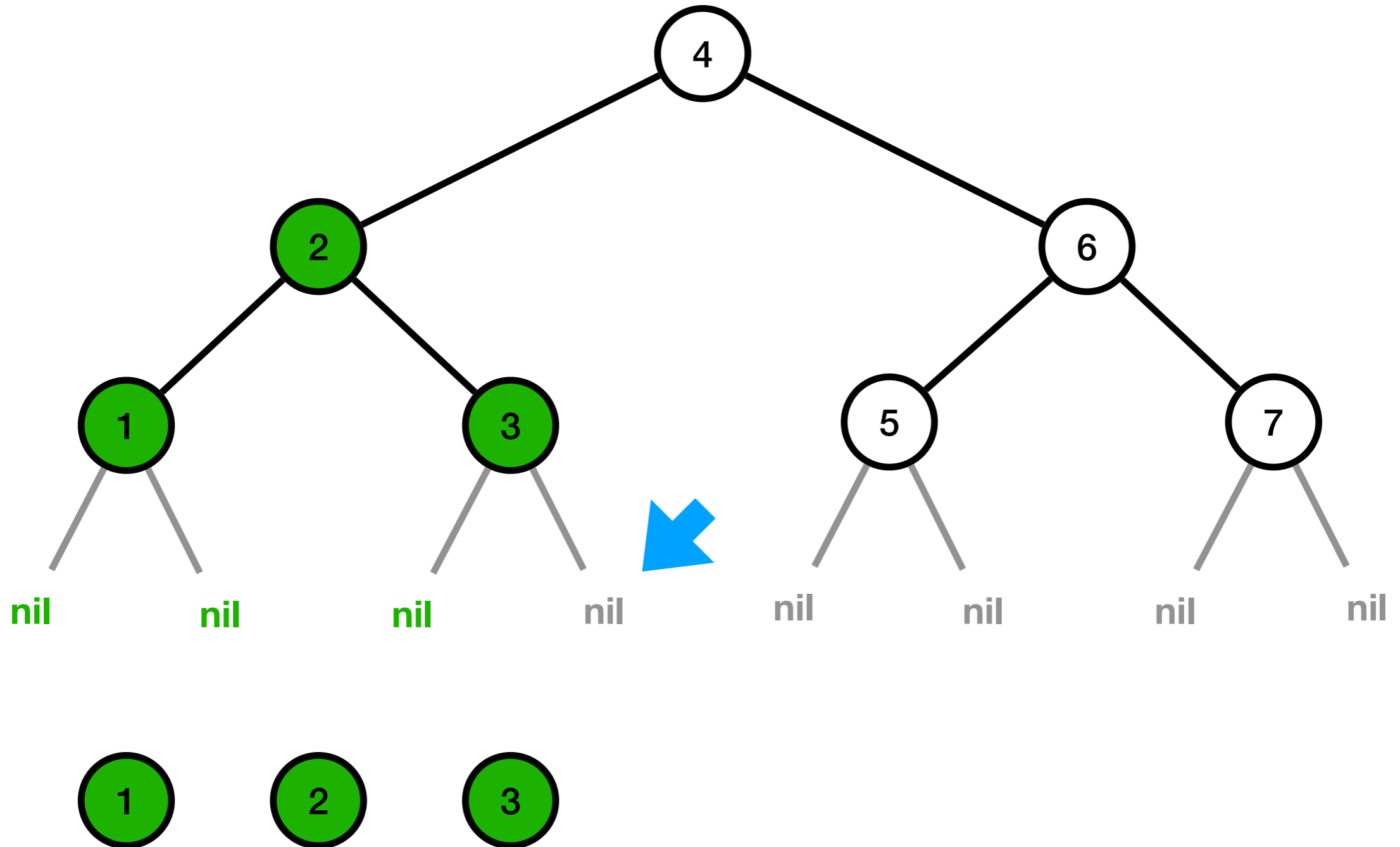
# Parcours infixe d'ABR



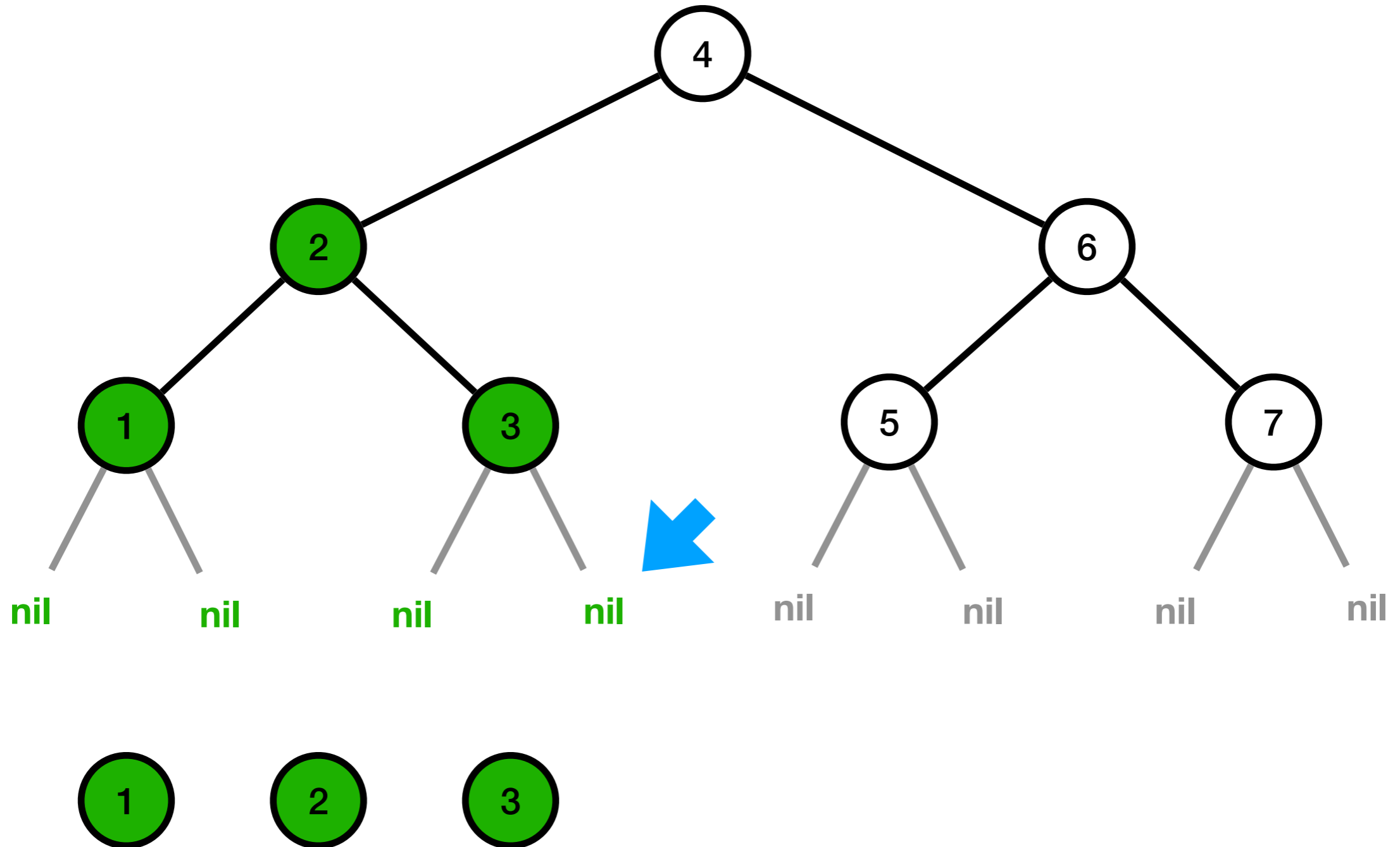
# Parcours infixe d'ABR



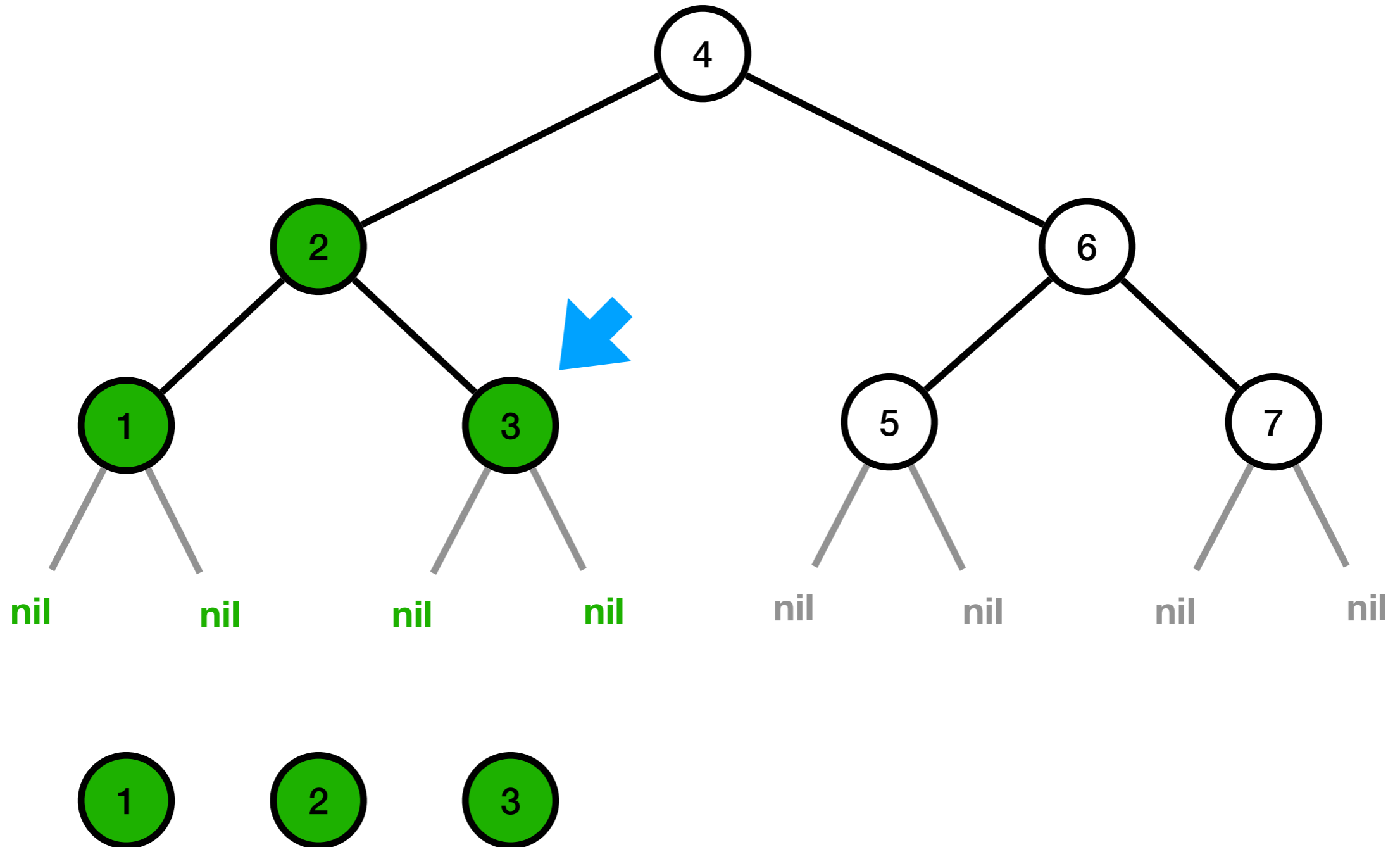
# Parcours infixe d'ABR



# Parcours infixe d'ABR

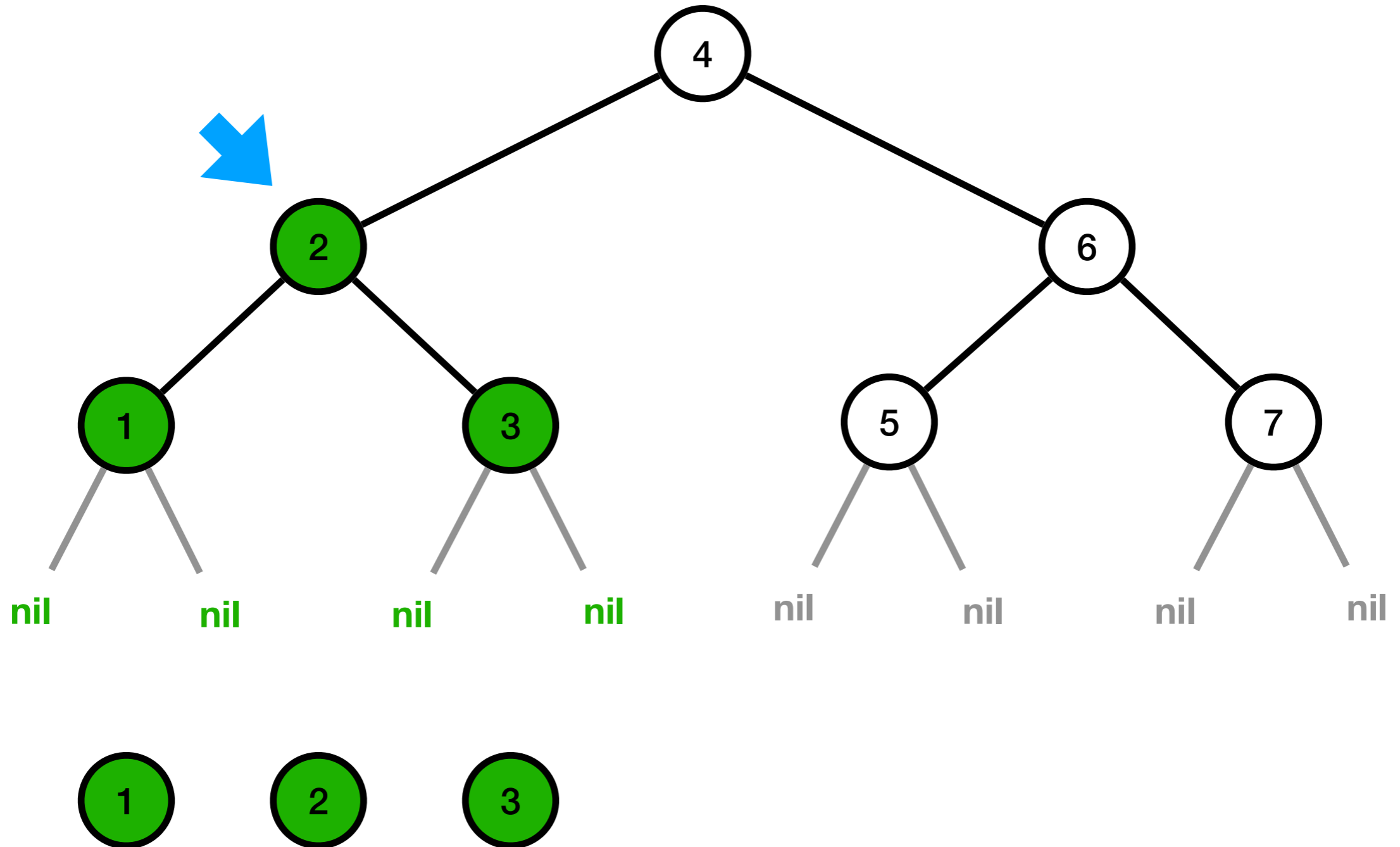


# Parcours infixe d'ABR

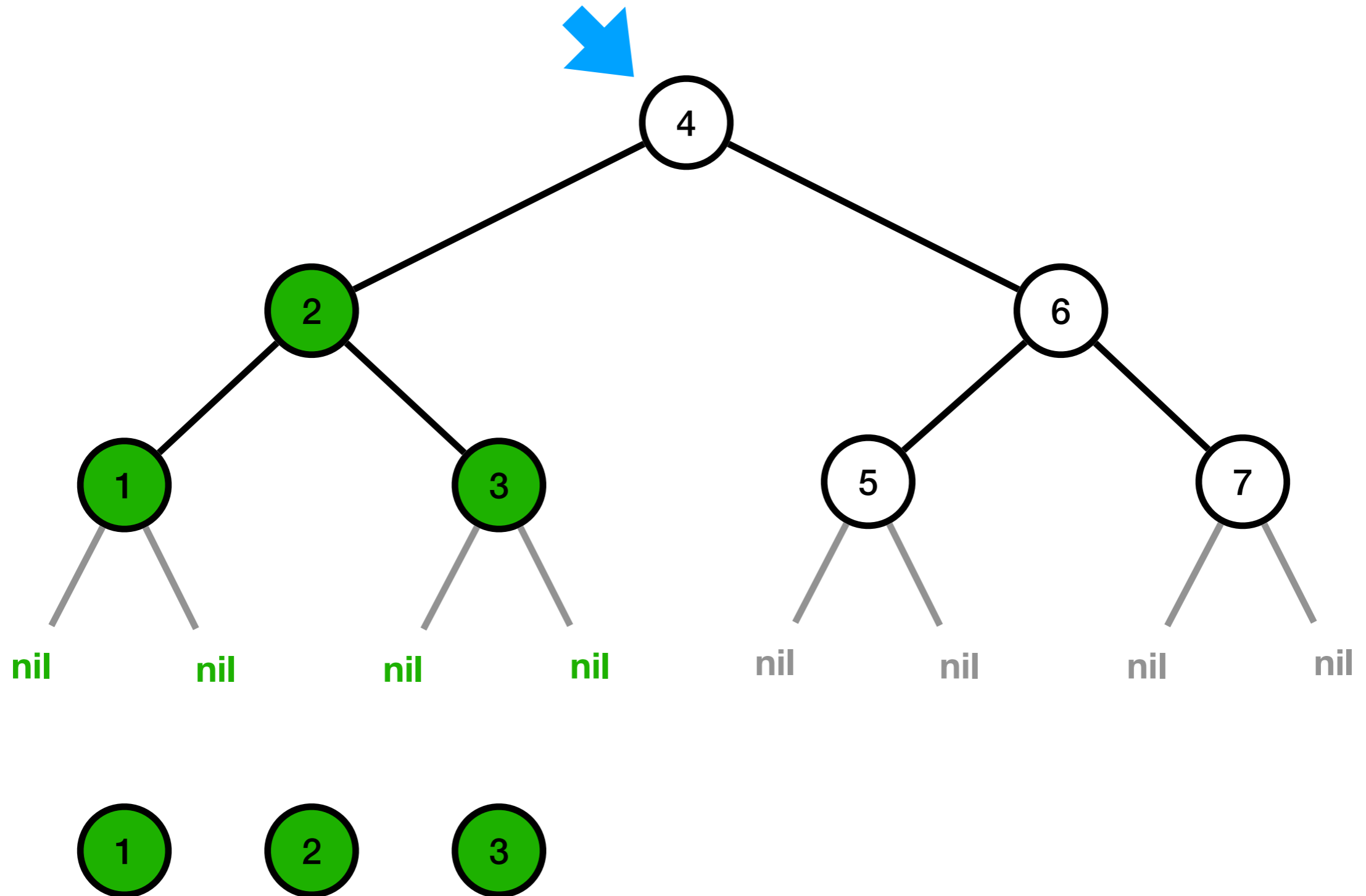




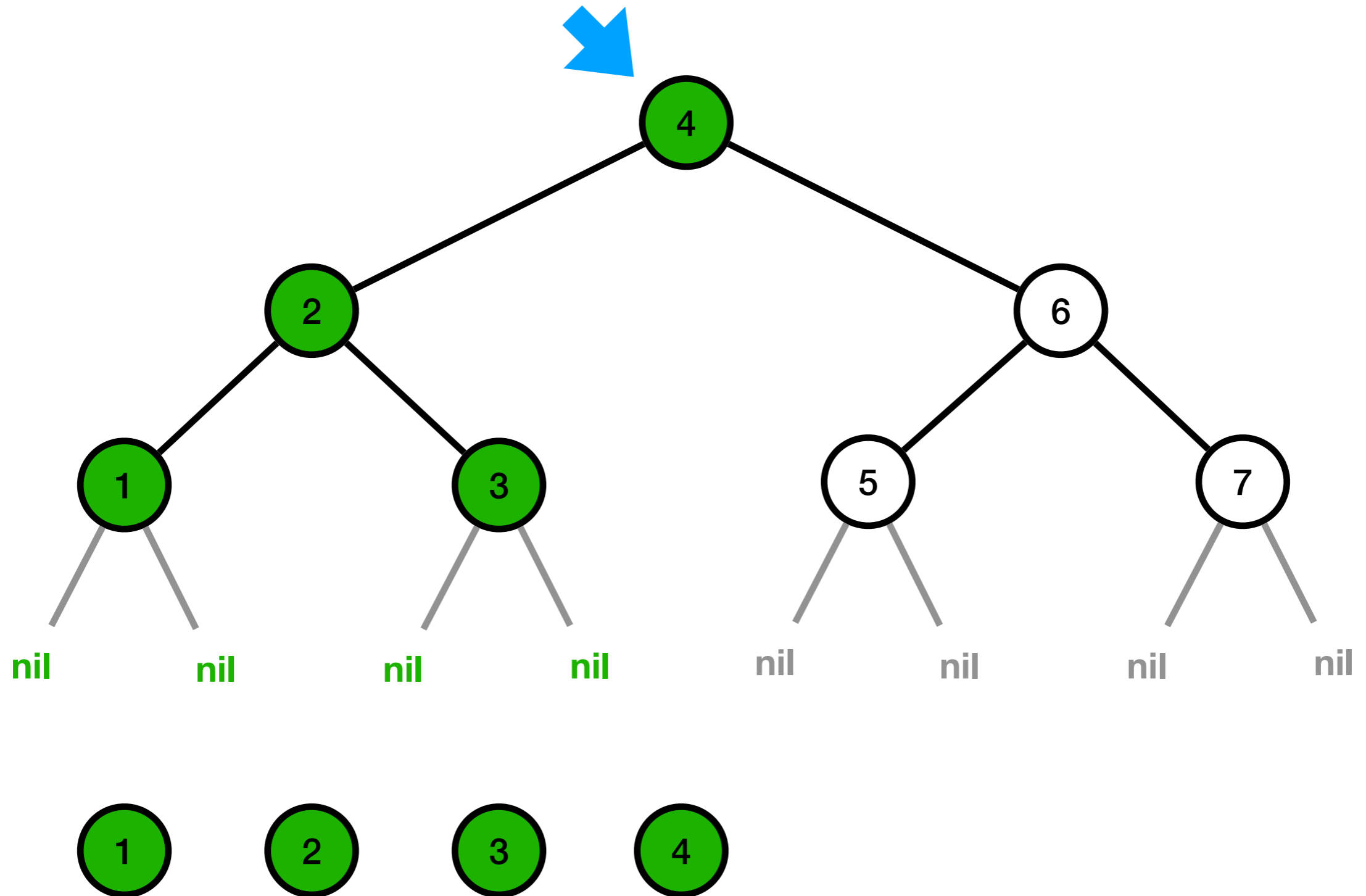
# Parcours infixe d'ABR



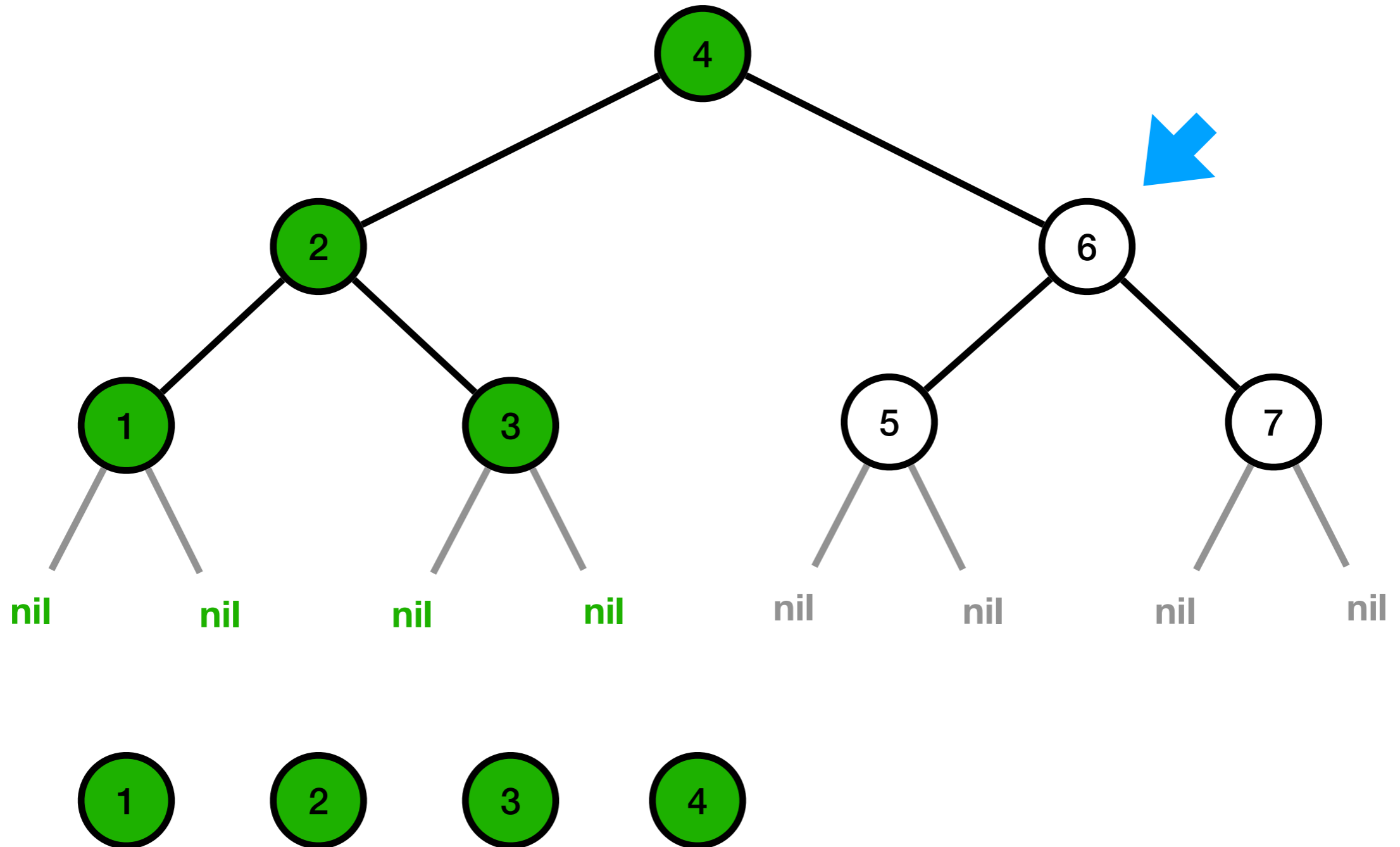
# Parcours infixe d'ABR



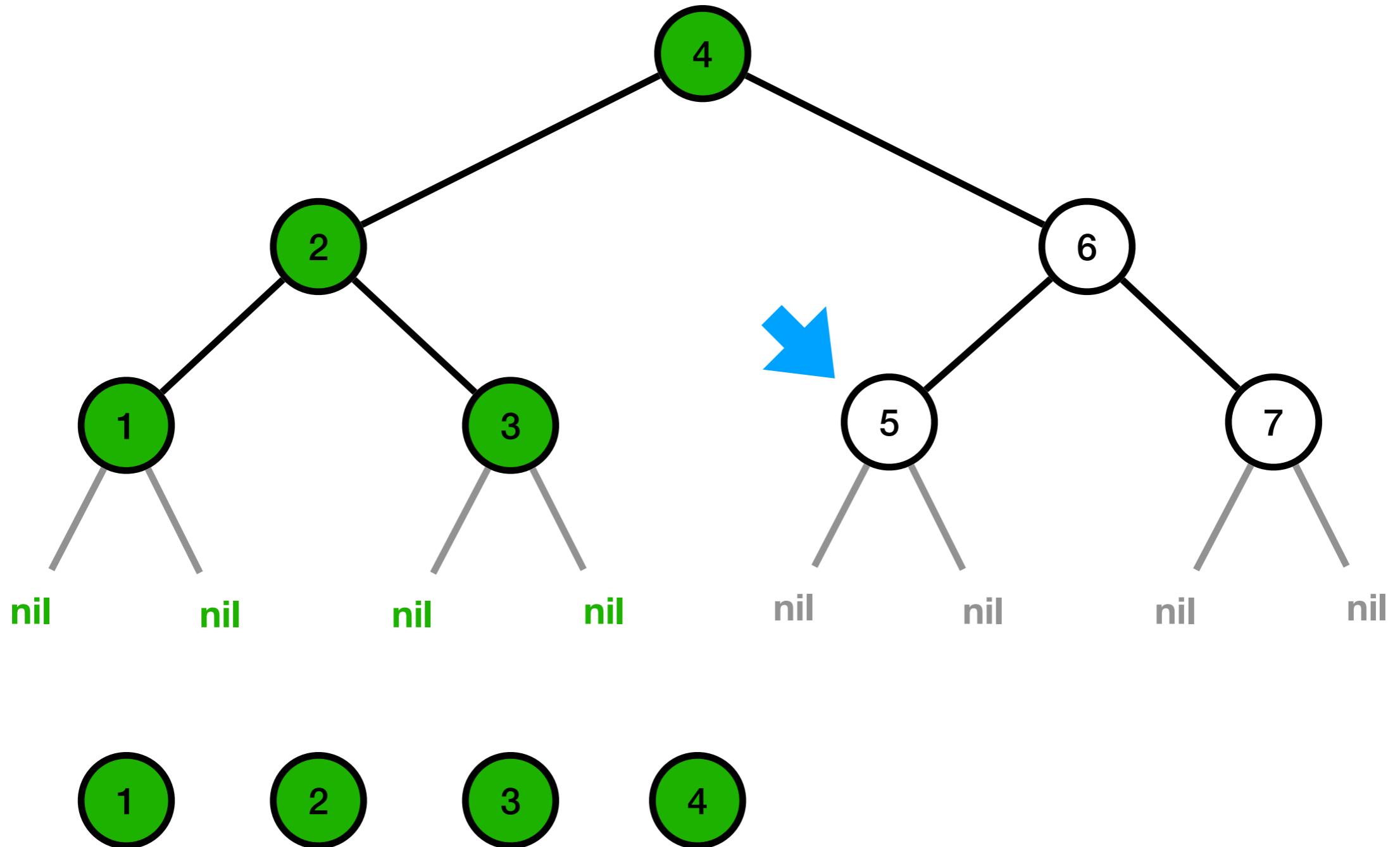
# Parcours infixe d'ABR



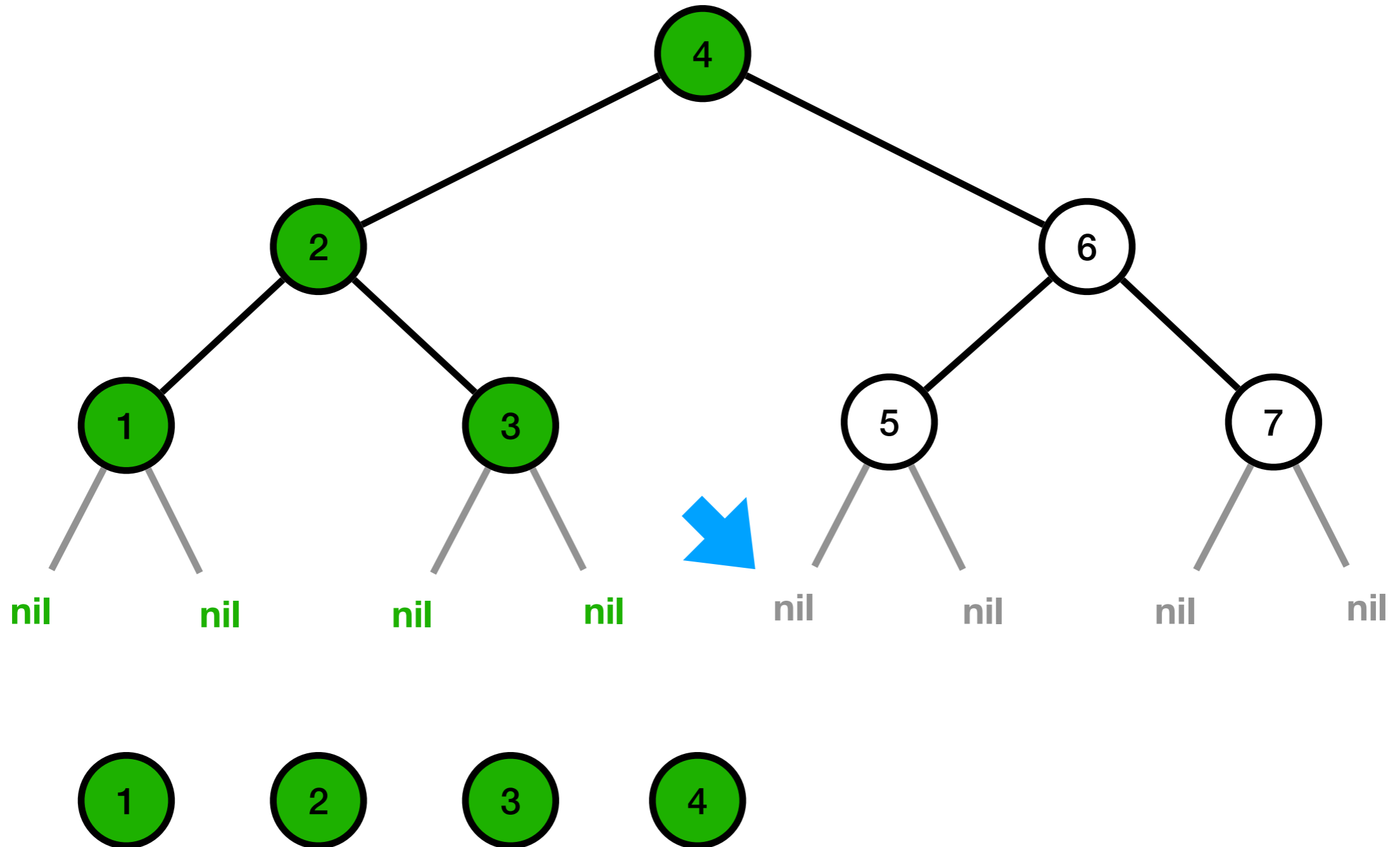
# Parcours infixe d'ABR



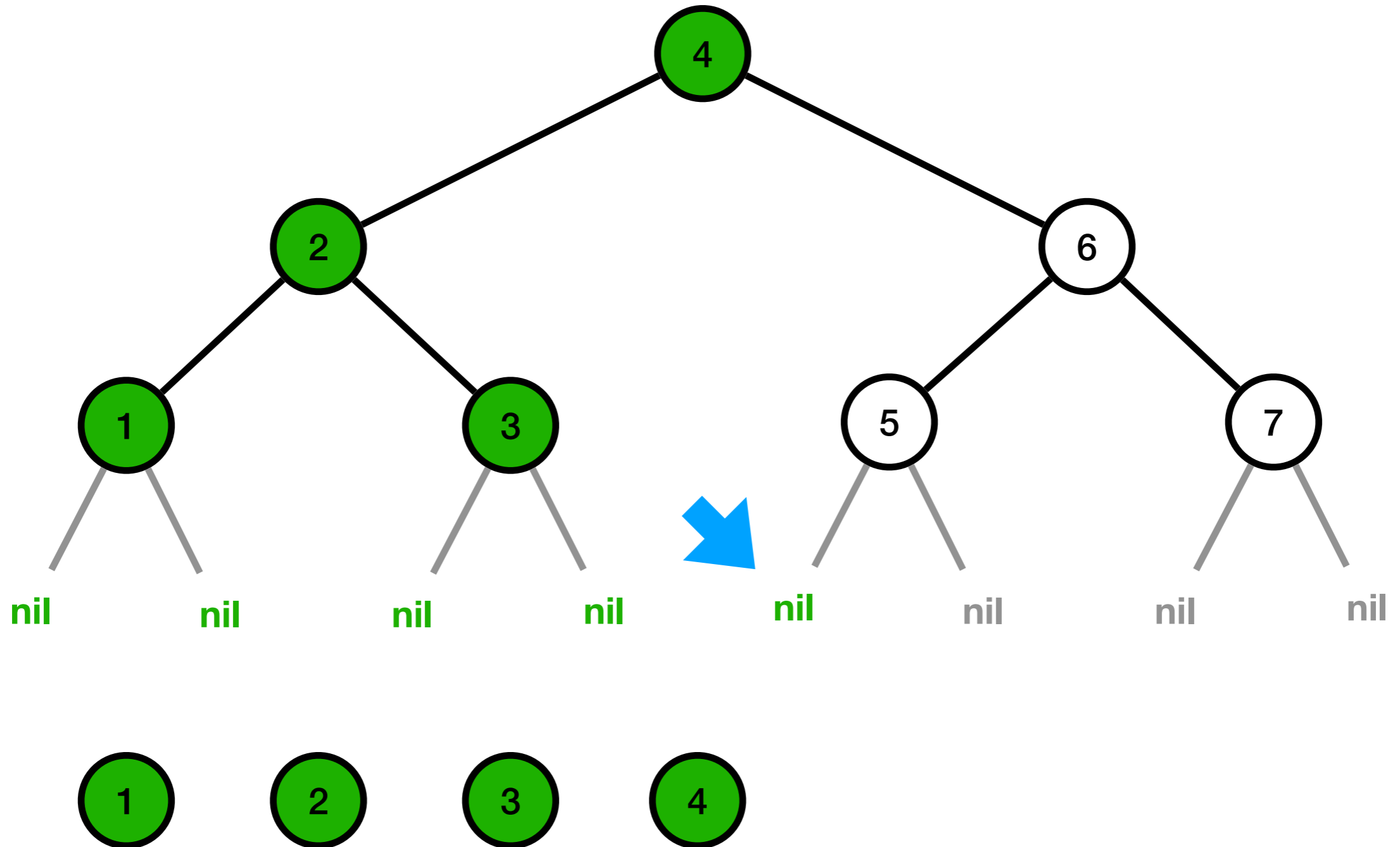
# Parcours infixe d'ABR



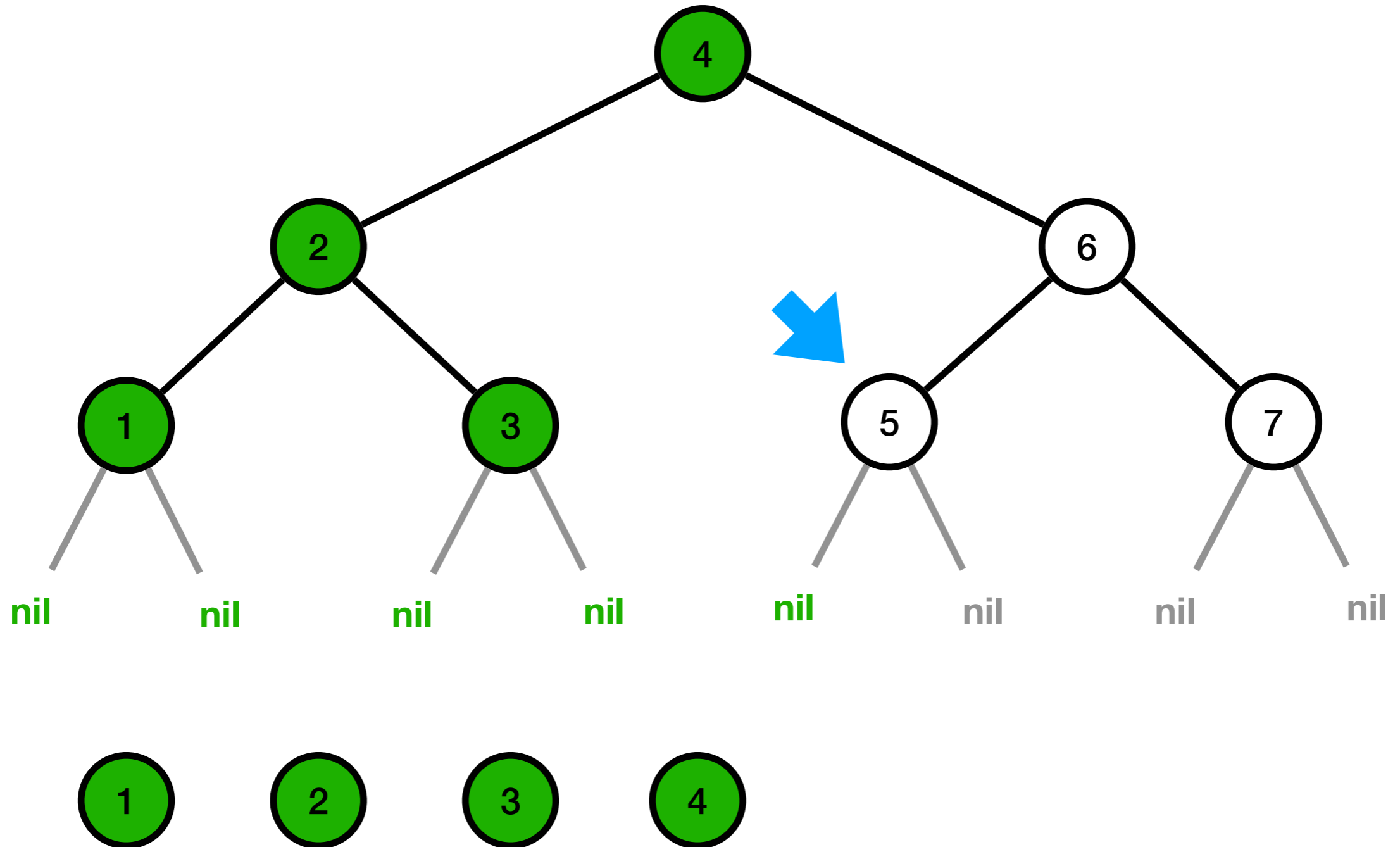
# Parcours infixe d'ABR



# Parcours infixe d'ABR

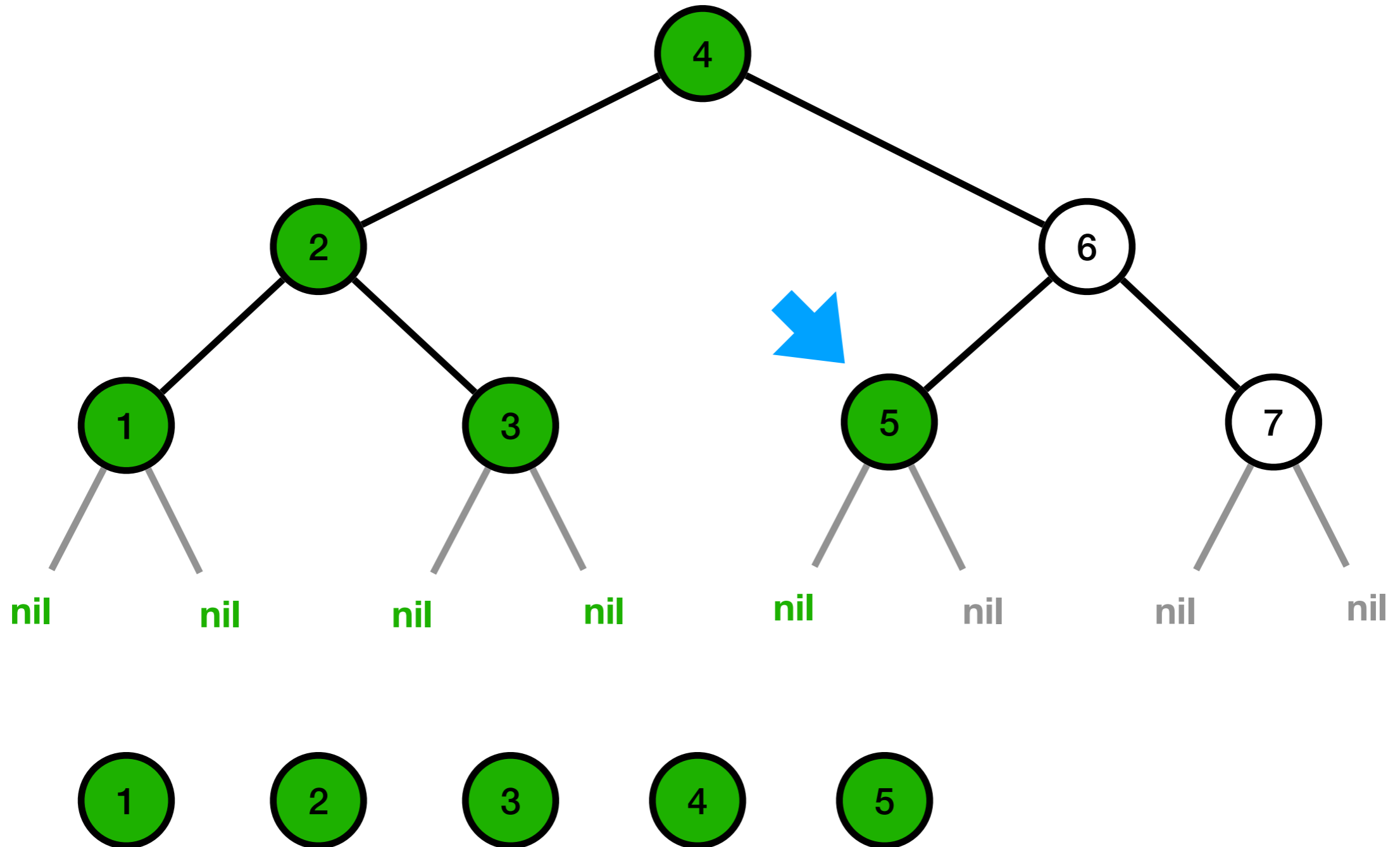


# Parcours infixe d'ABR

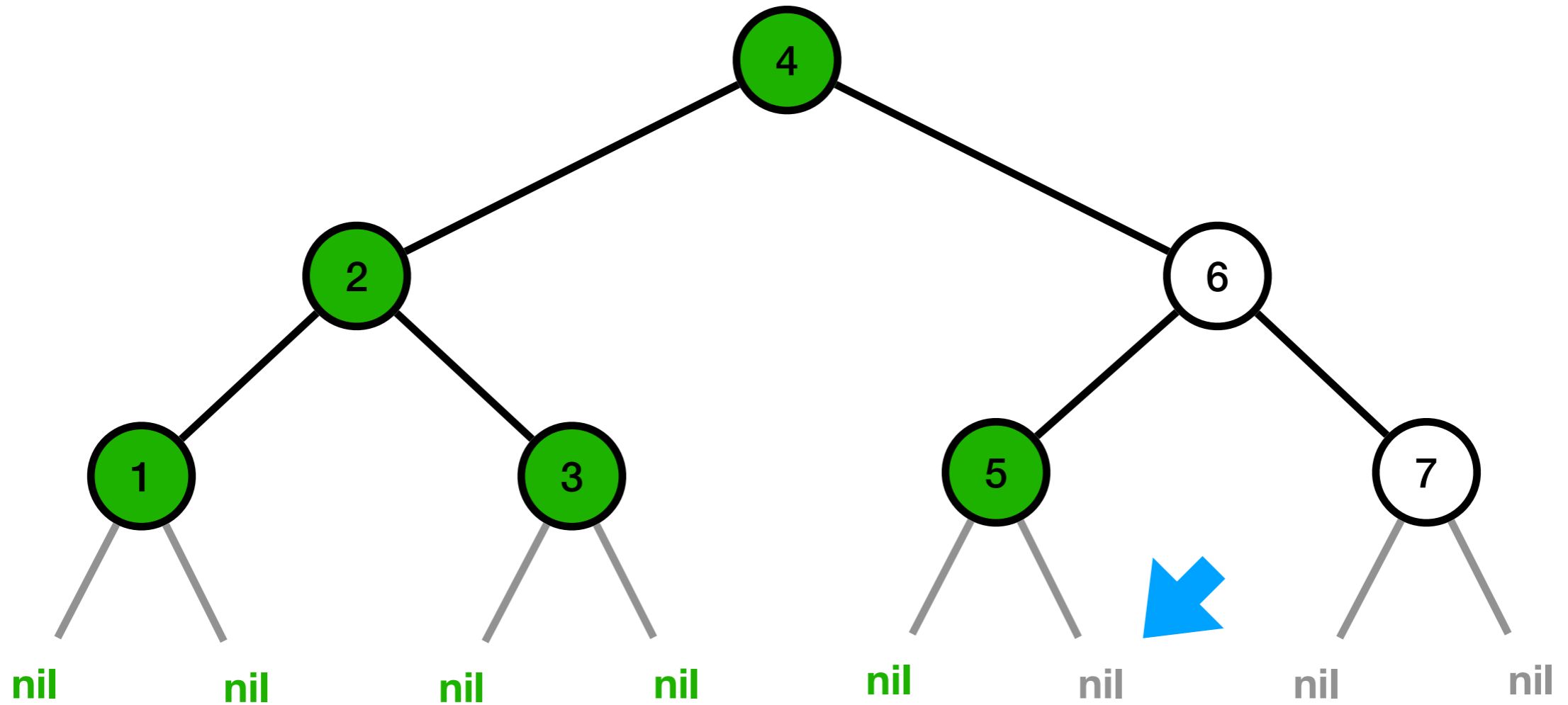




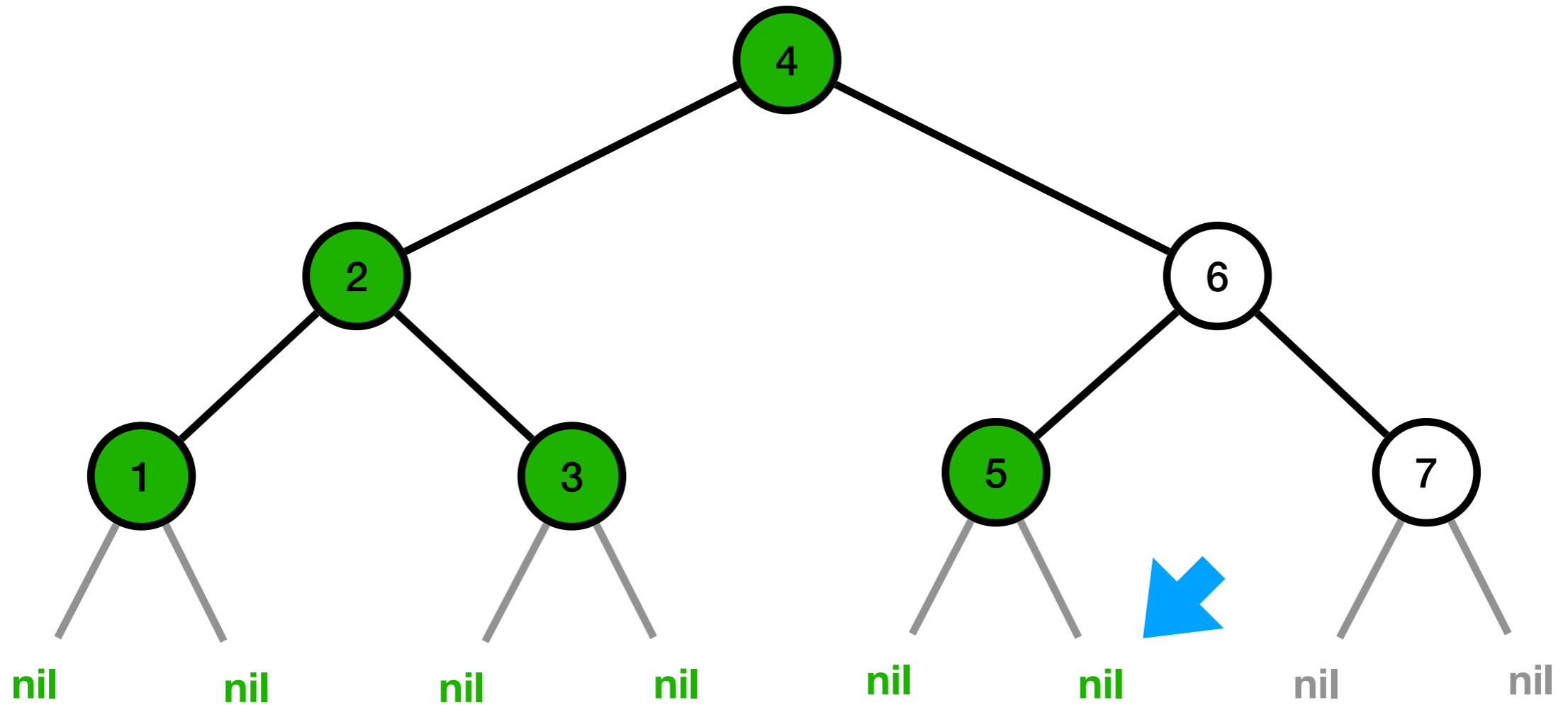
# Parcours infixe d'ABR



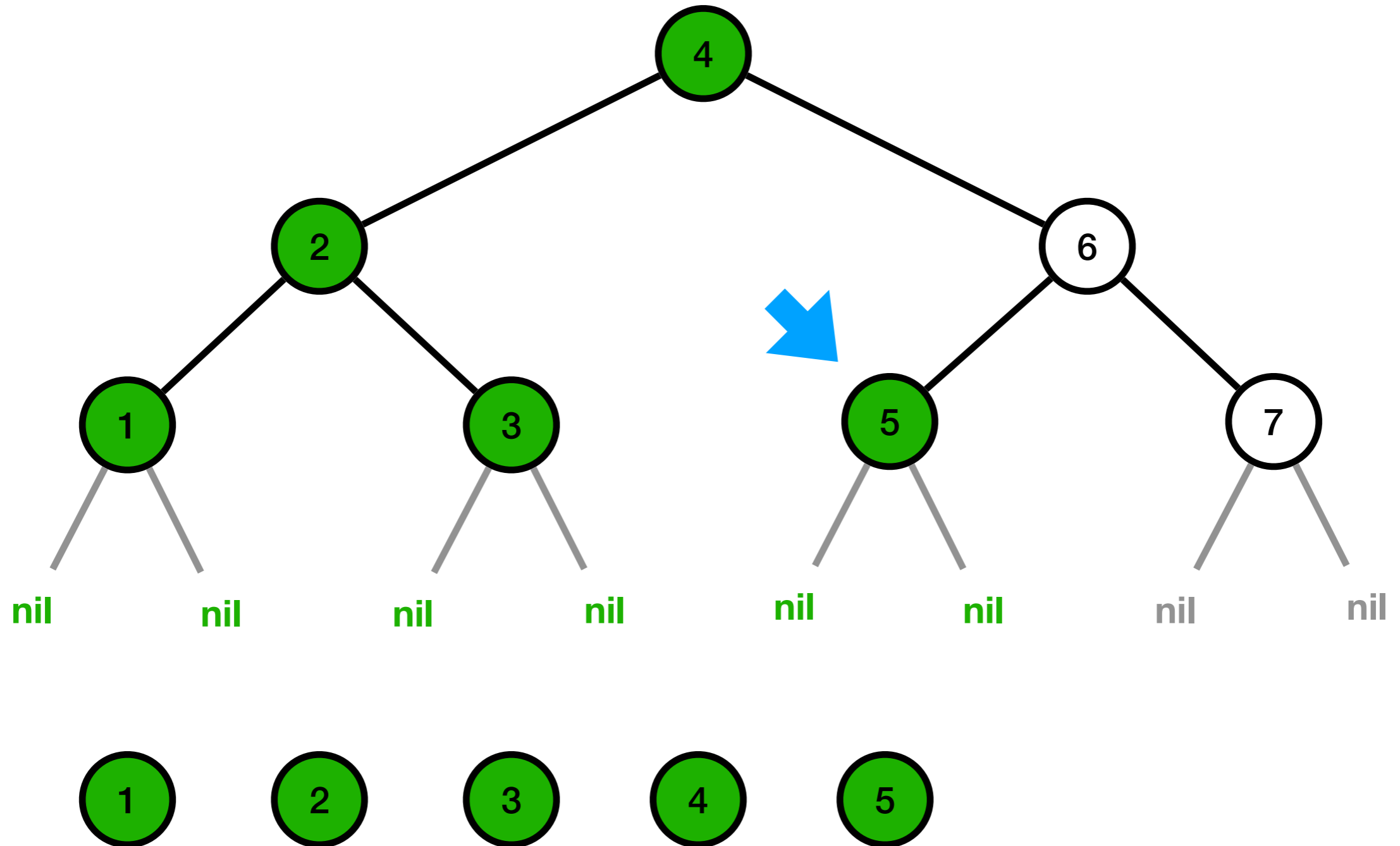
# Parcours infixe d'ABR



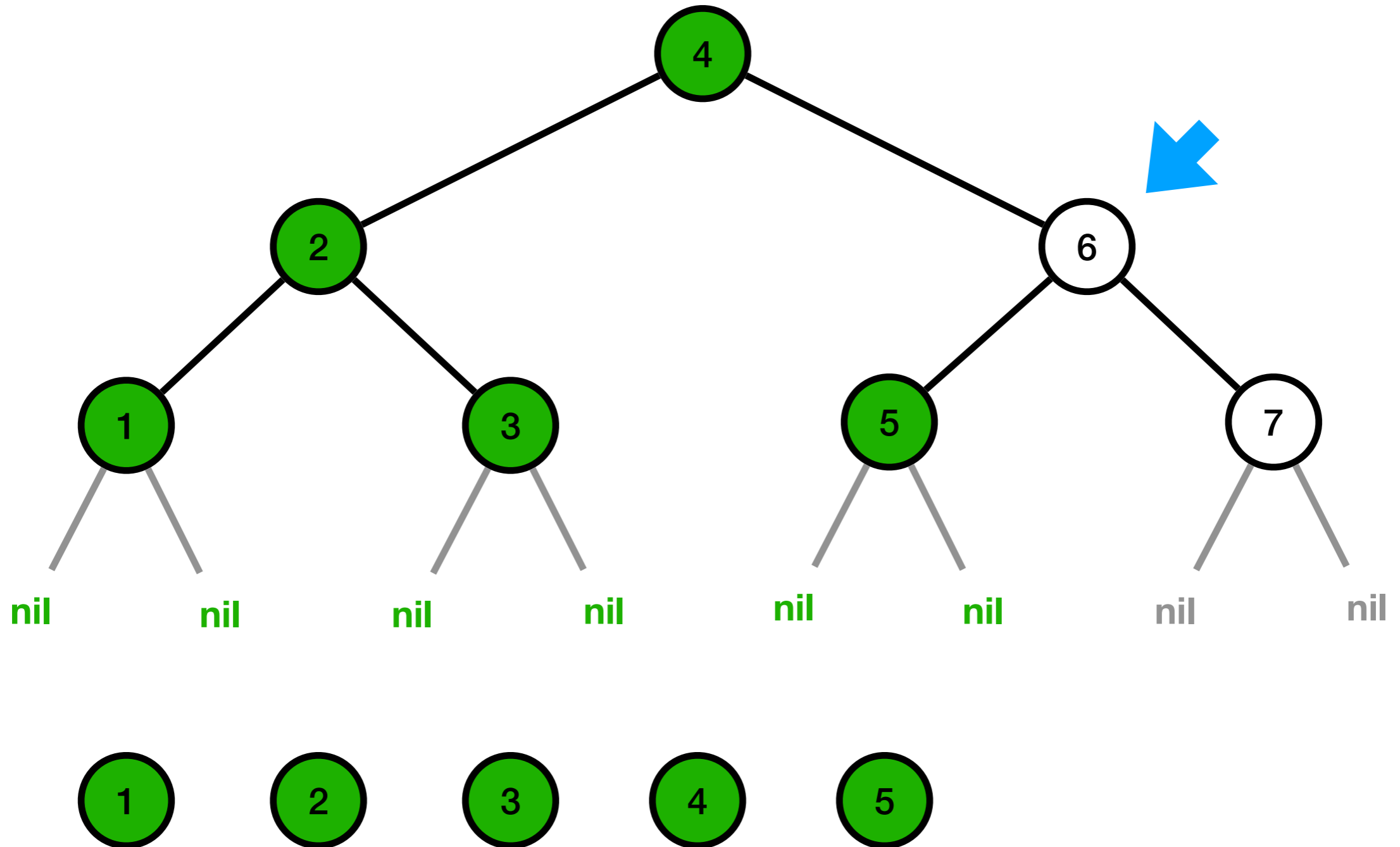
# Parcours infixe d'ABR



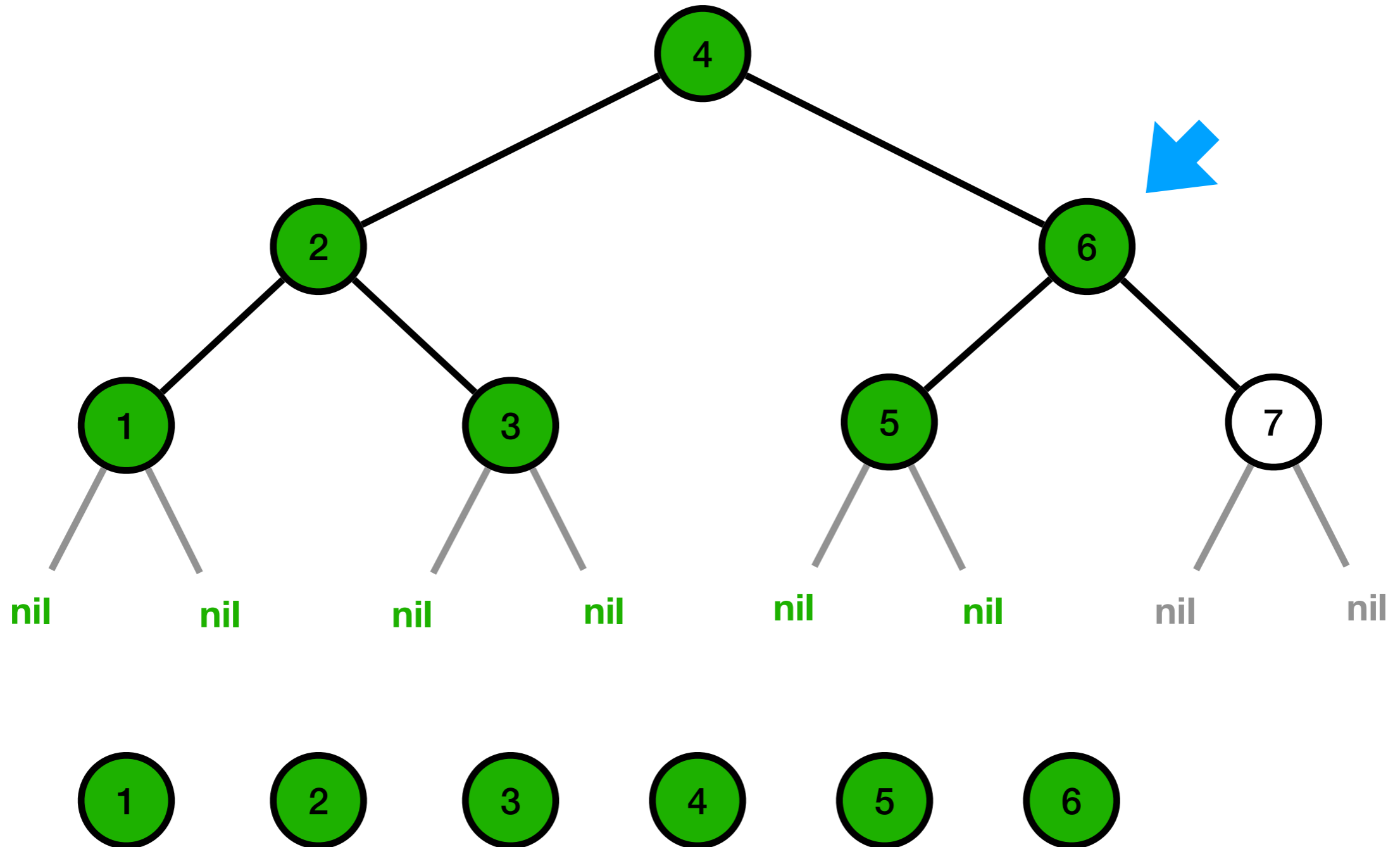
# Parcours infixe d'ABR



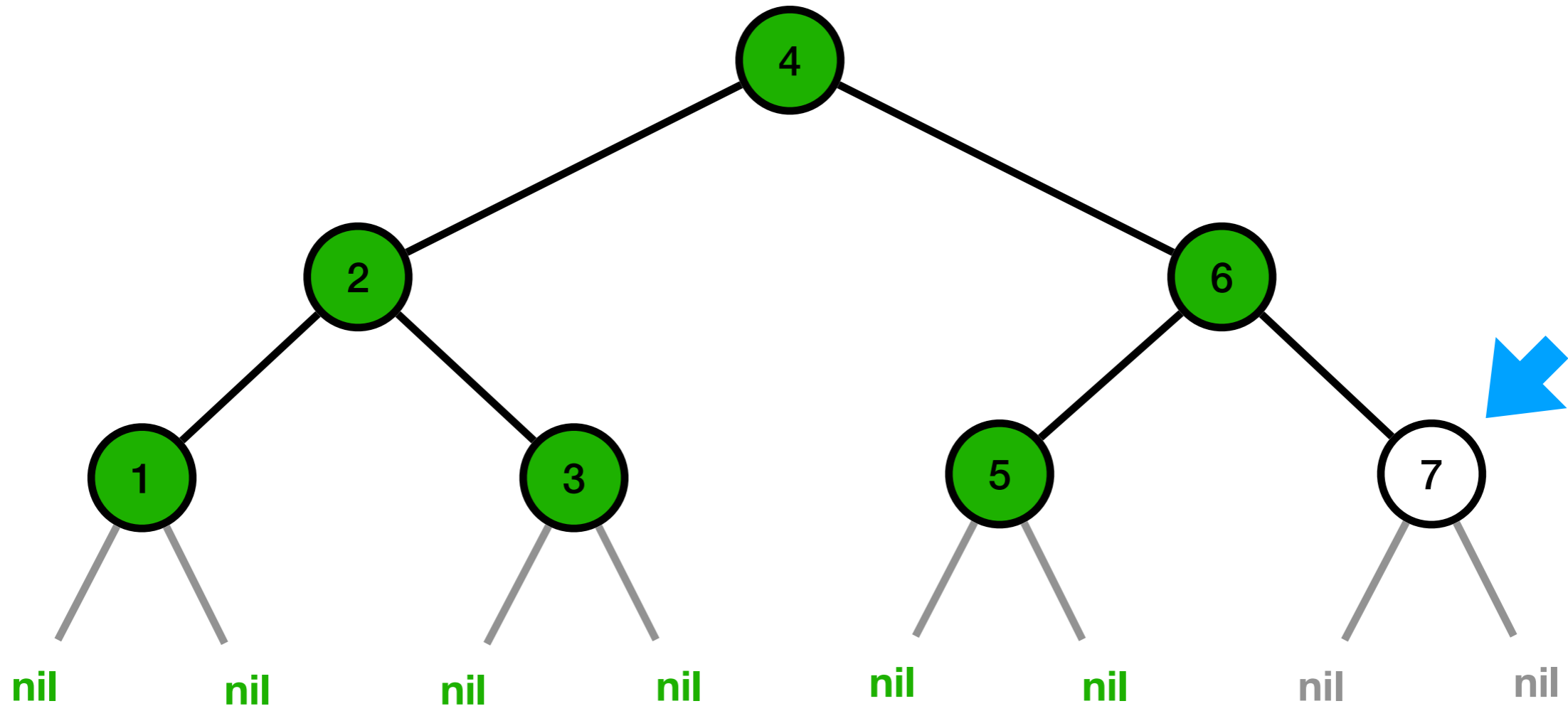
# Parcours infixe d'ABR



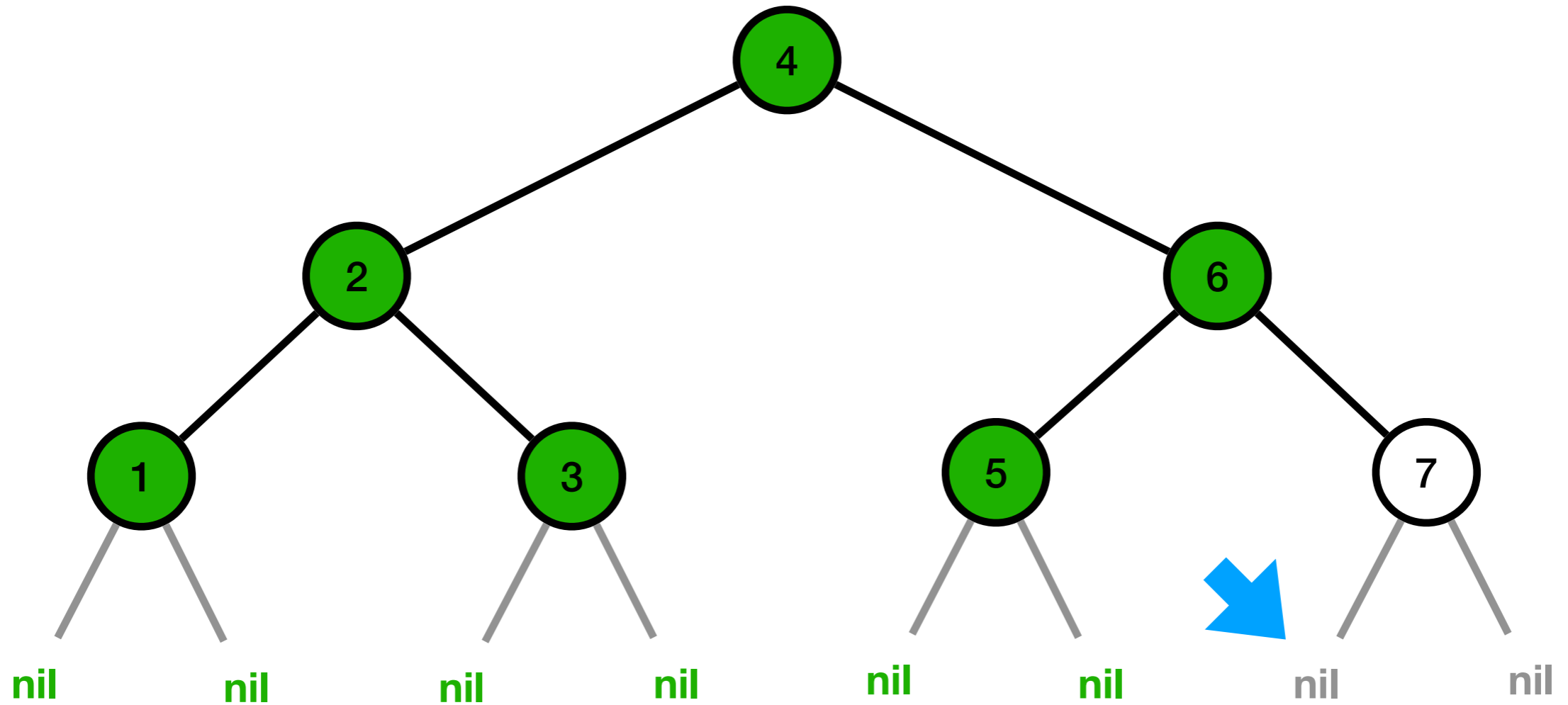
# Parcours infixe d'ABR



# Parcours infixe d'ABR

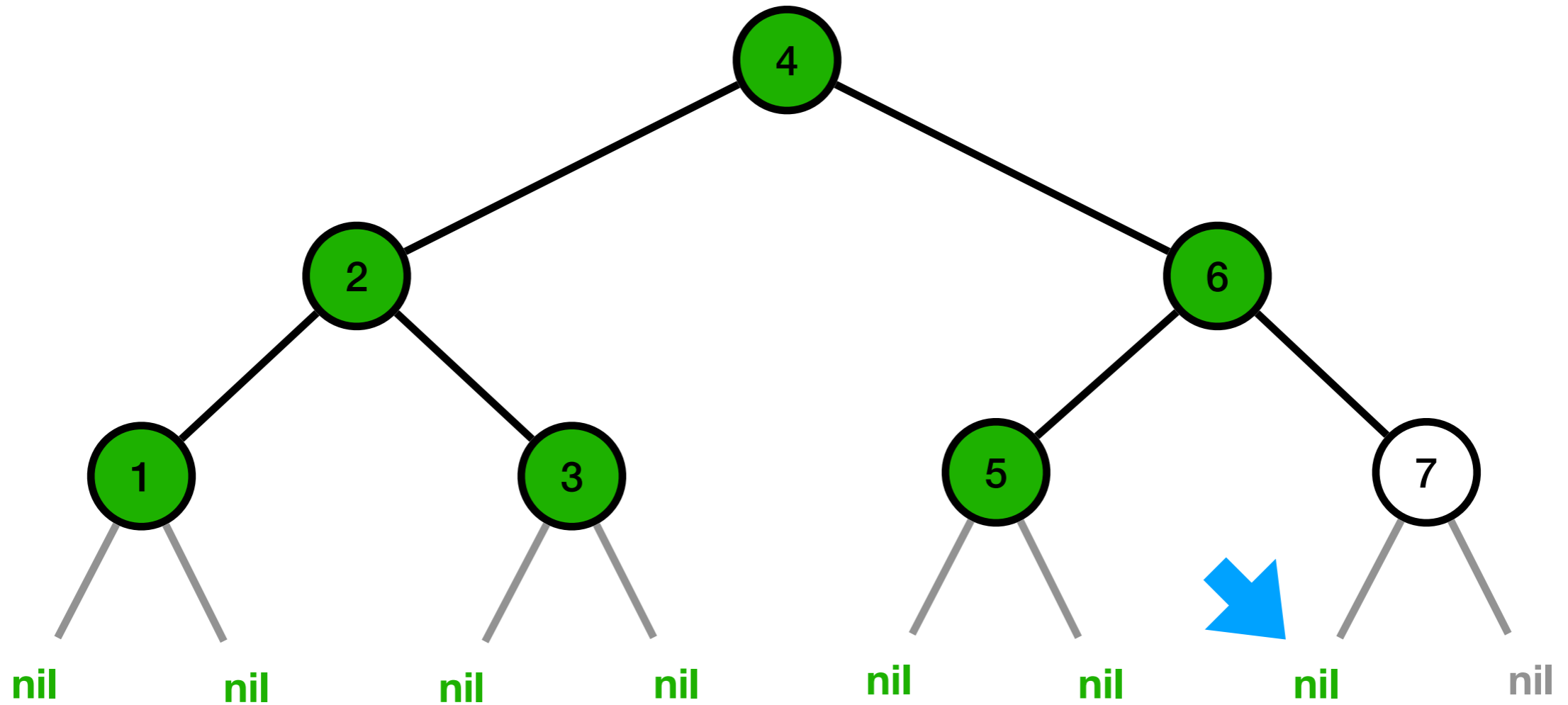


# Parcours infixe d'ABR

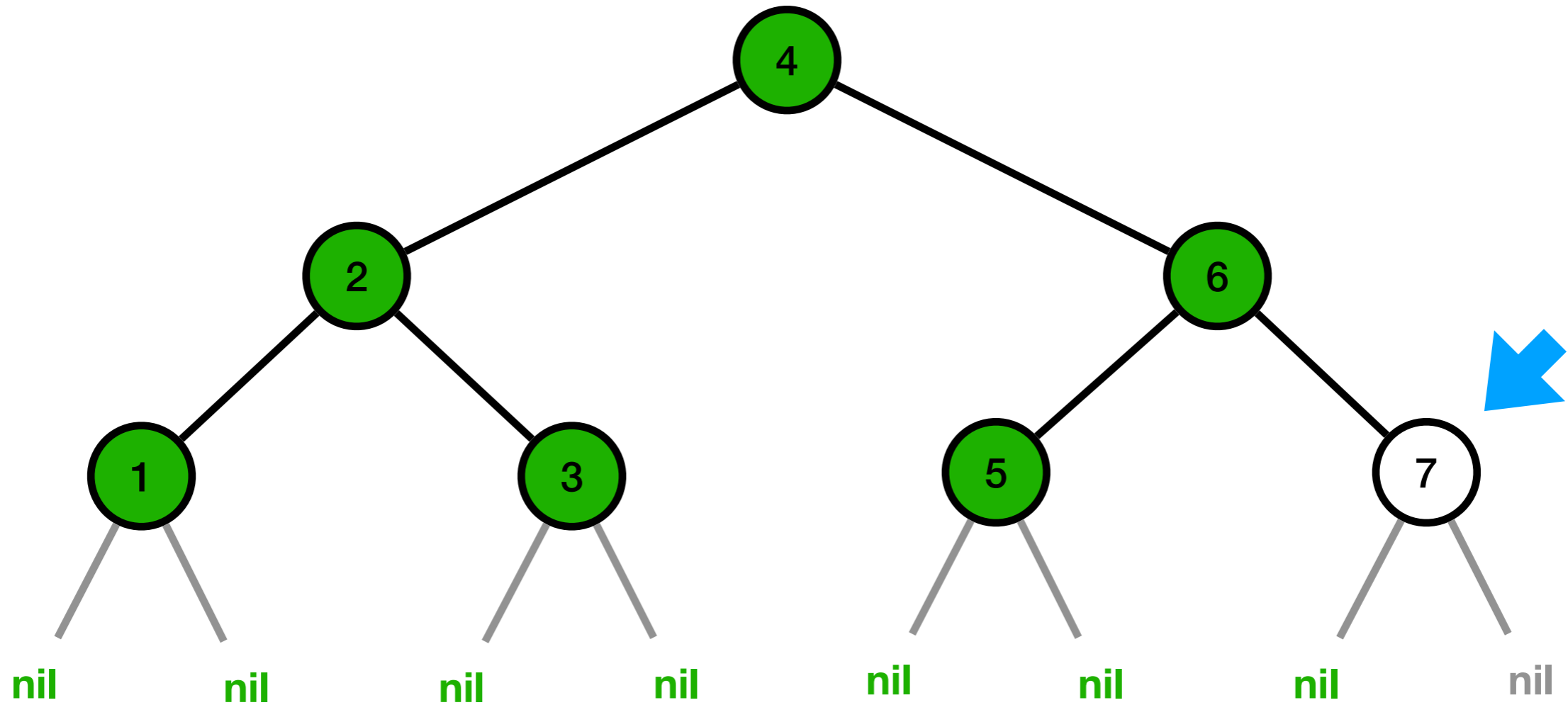




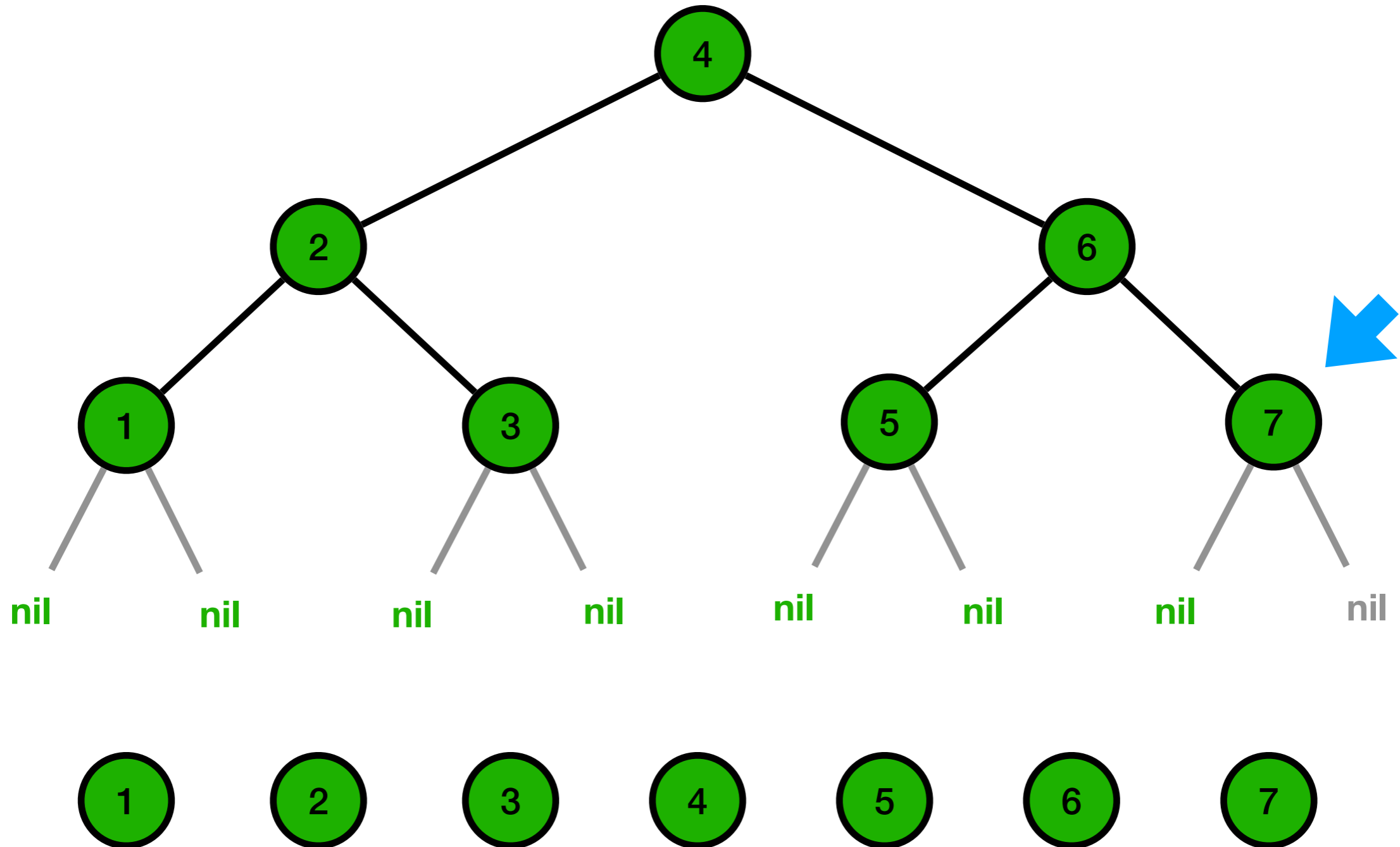
# Parcours infixe d'ABR



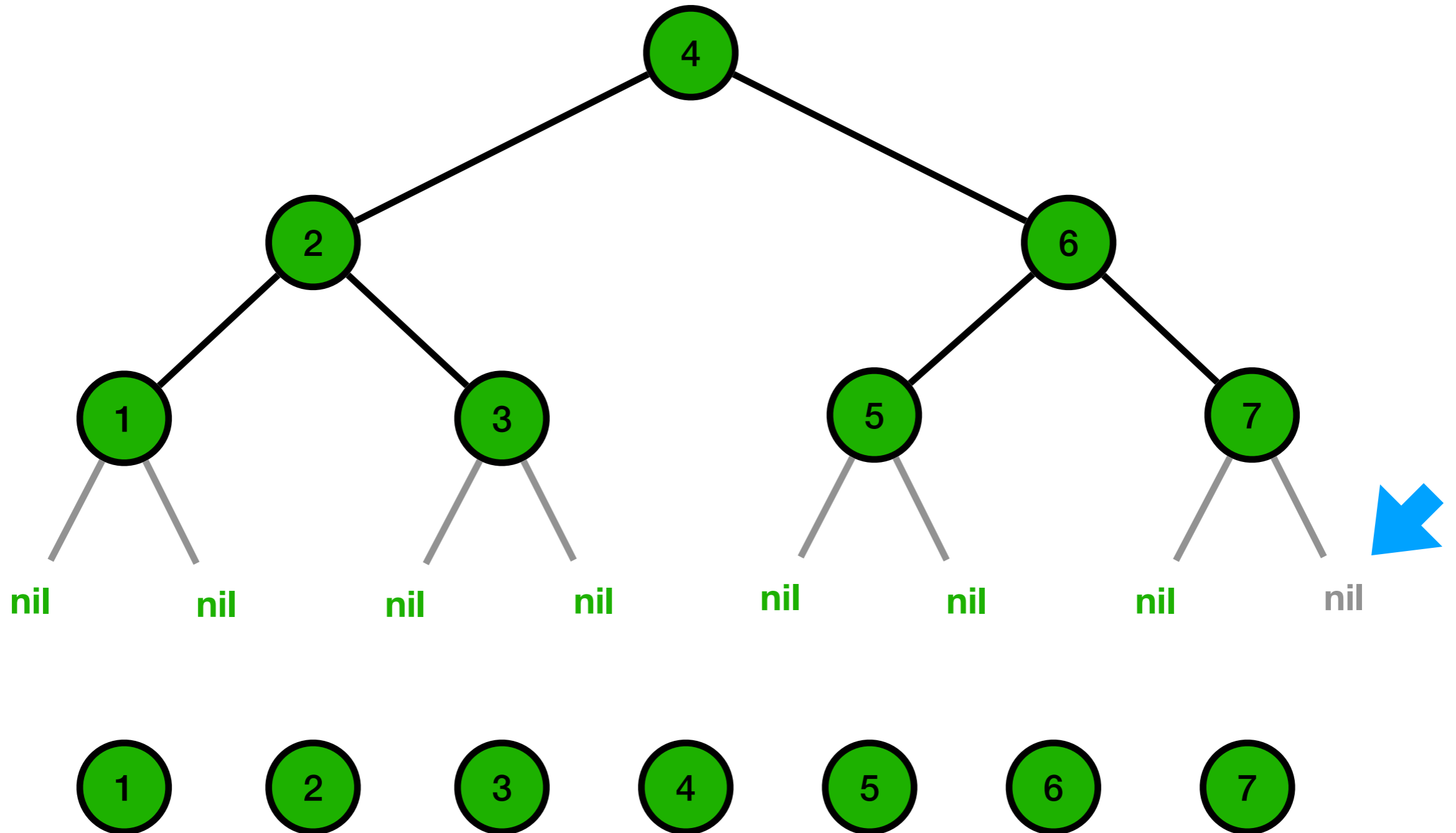
# Parcours infixe d'ABR



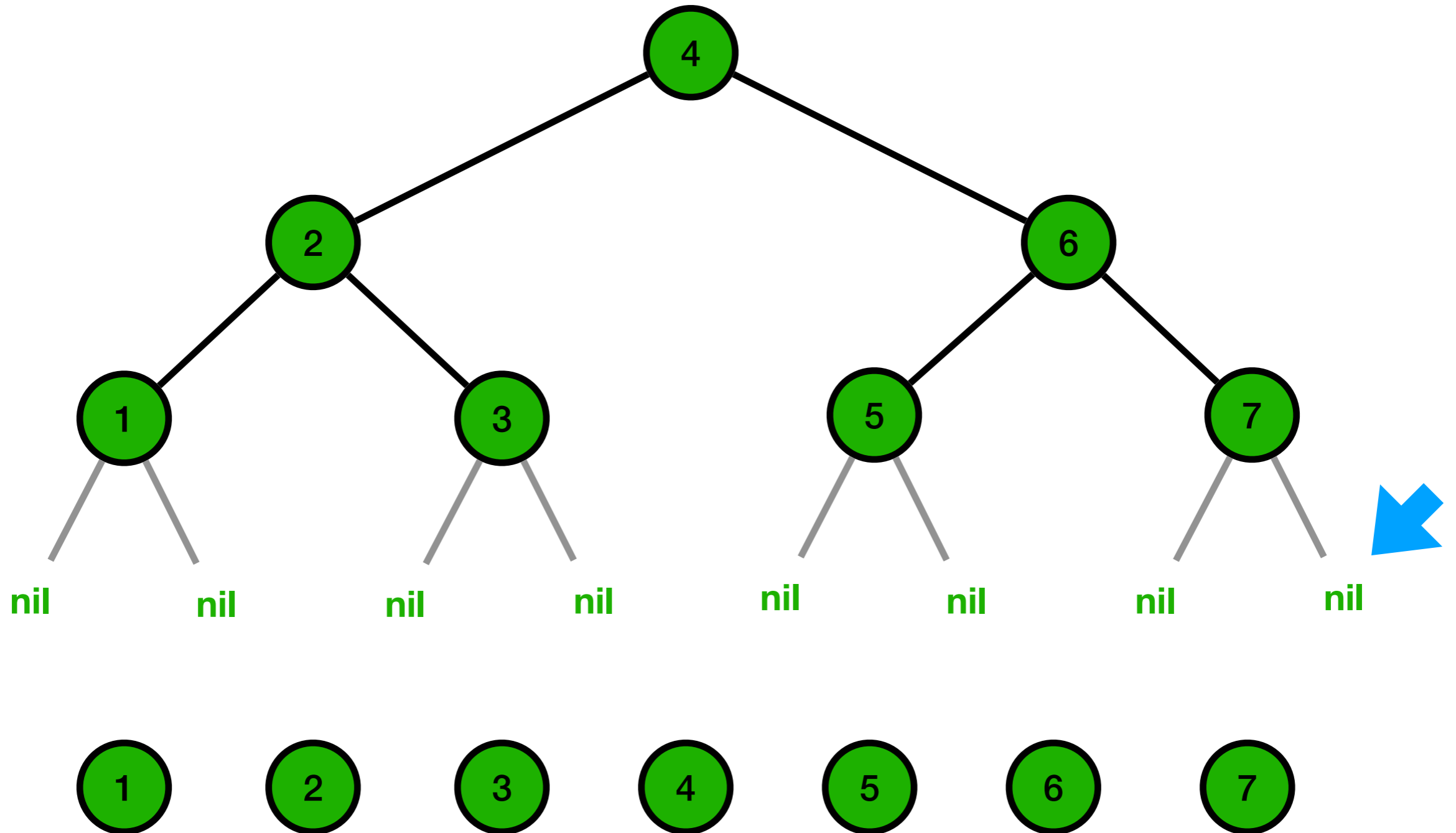
# Parcours infixe d'ABR



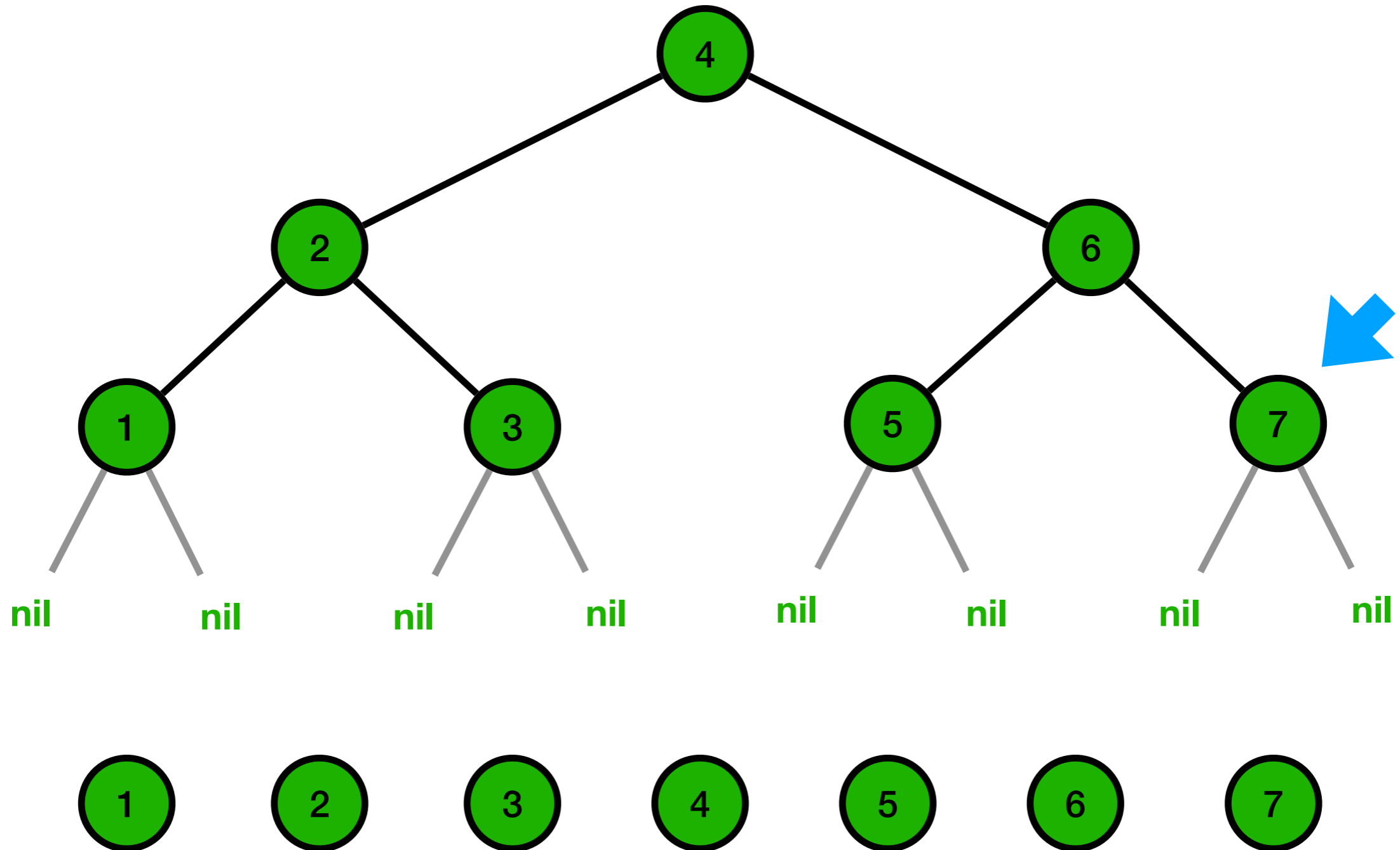
# Parcours infixe d'ABR



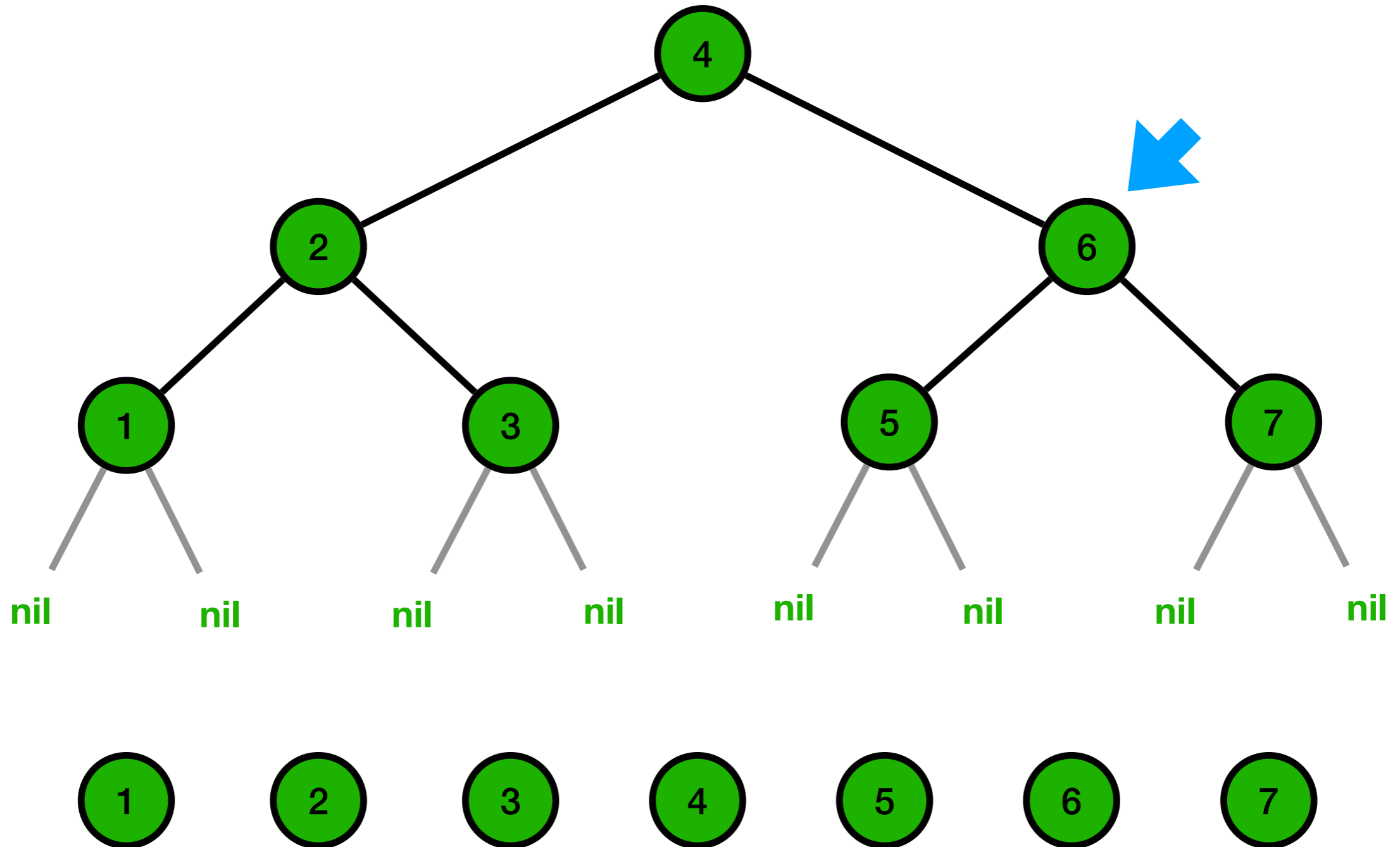
# Parcours infixe d'ABR



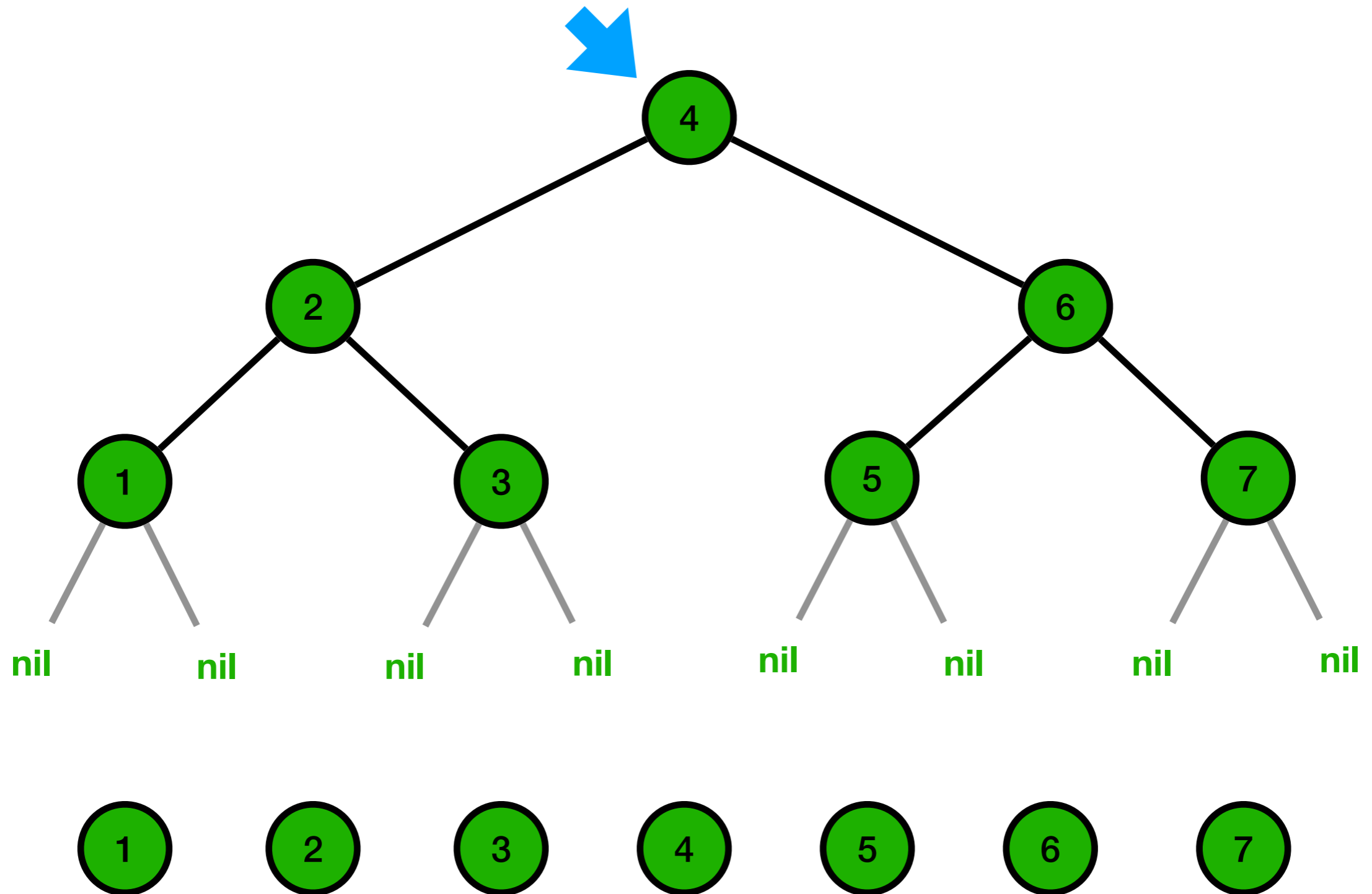
# Parcours infixe d'ABR



# Parcours infixe d'ABR

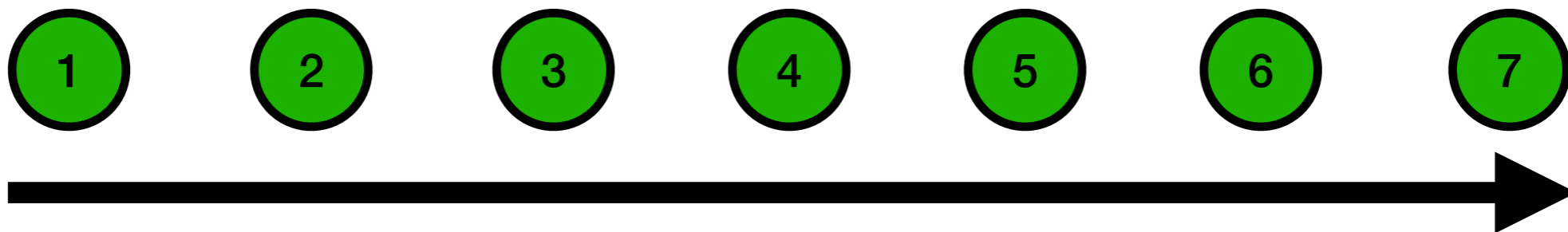
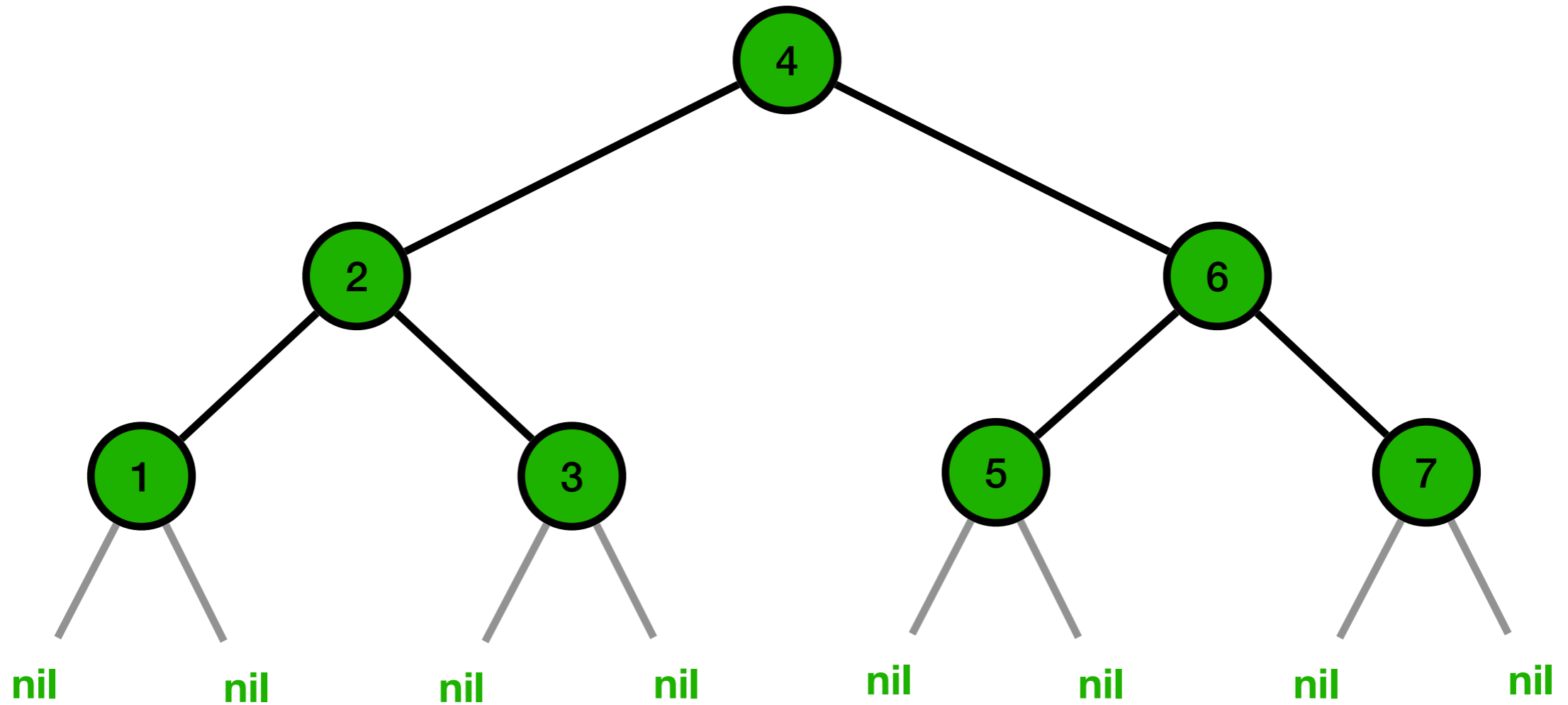


# Parcours infixe d'ABR





# Parcours infixe d'ABR



# Arbres de décision



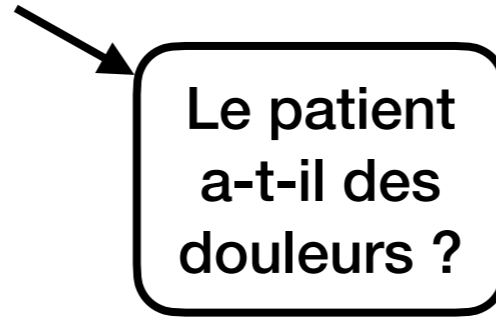
# Arbres de décision



# Arbres de décision

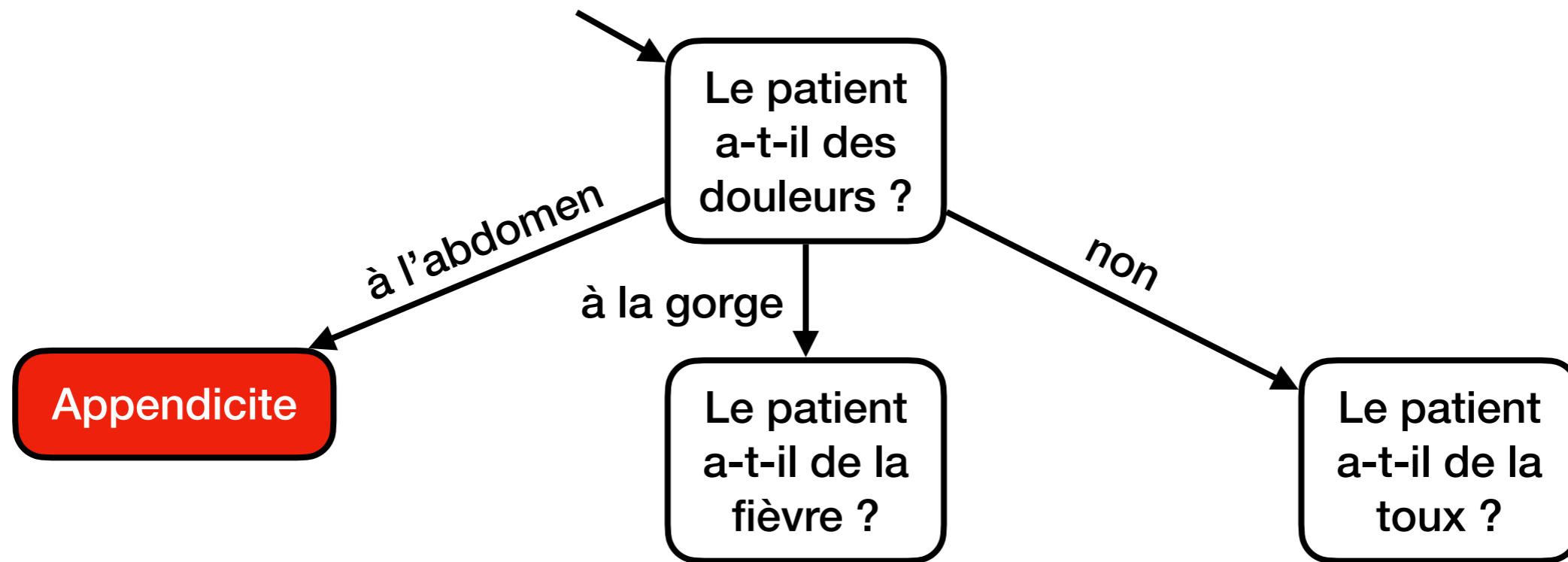


# Diagnostic de maladies

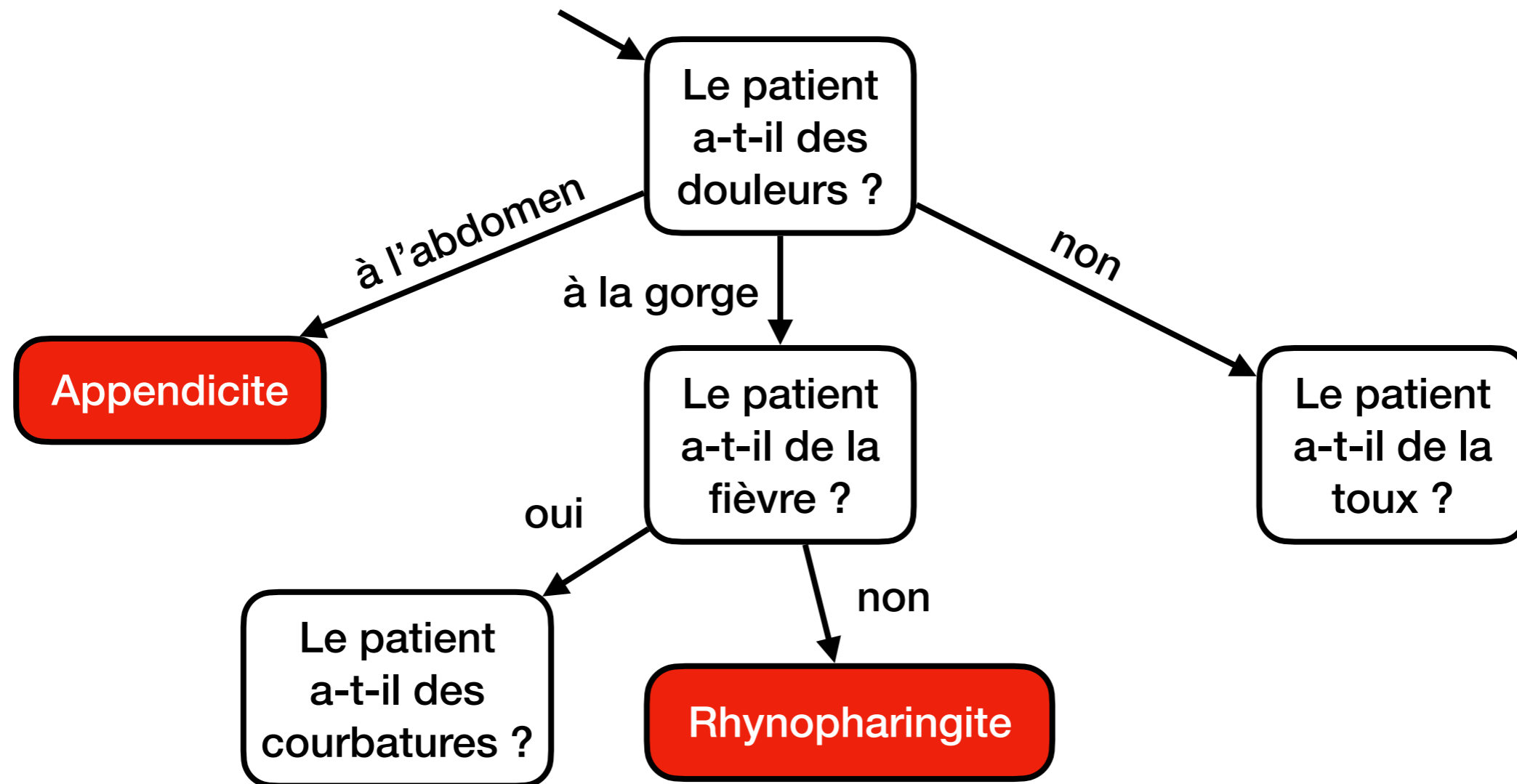


Le patient  
a-t-il des  
douleurs ?

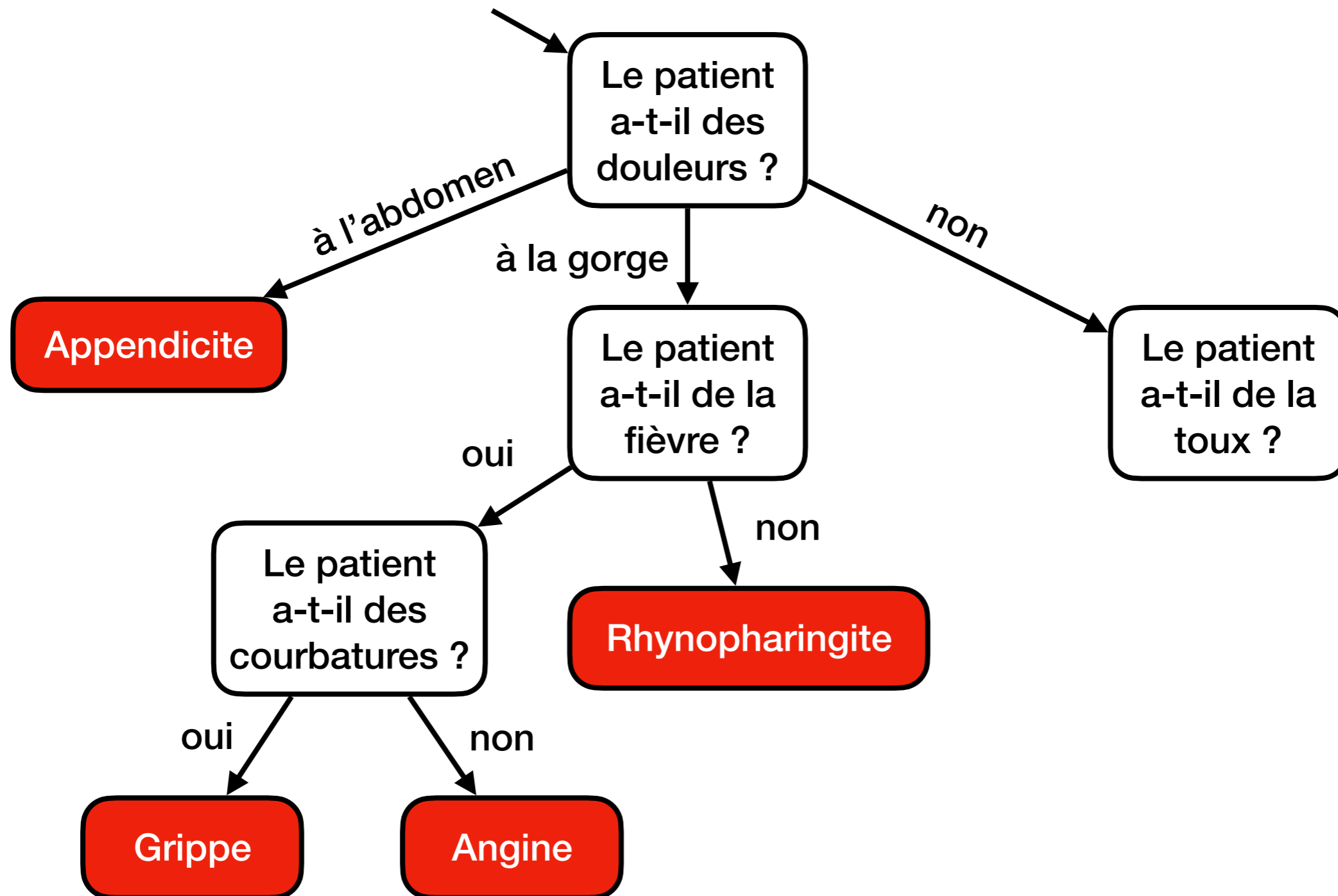
# Diagnostic de maladies



# Diagnostic de maladies

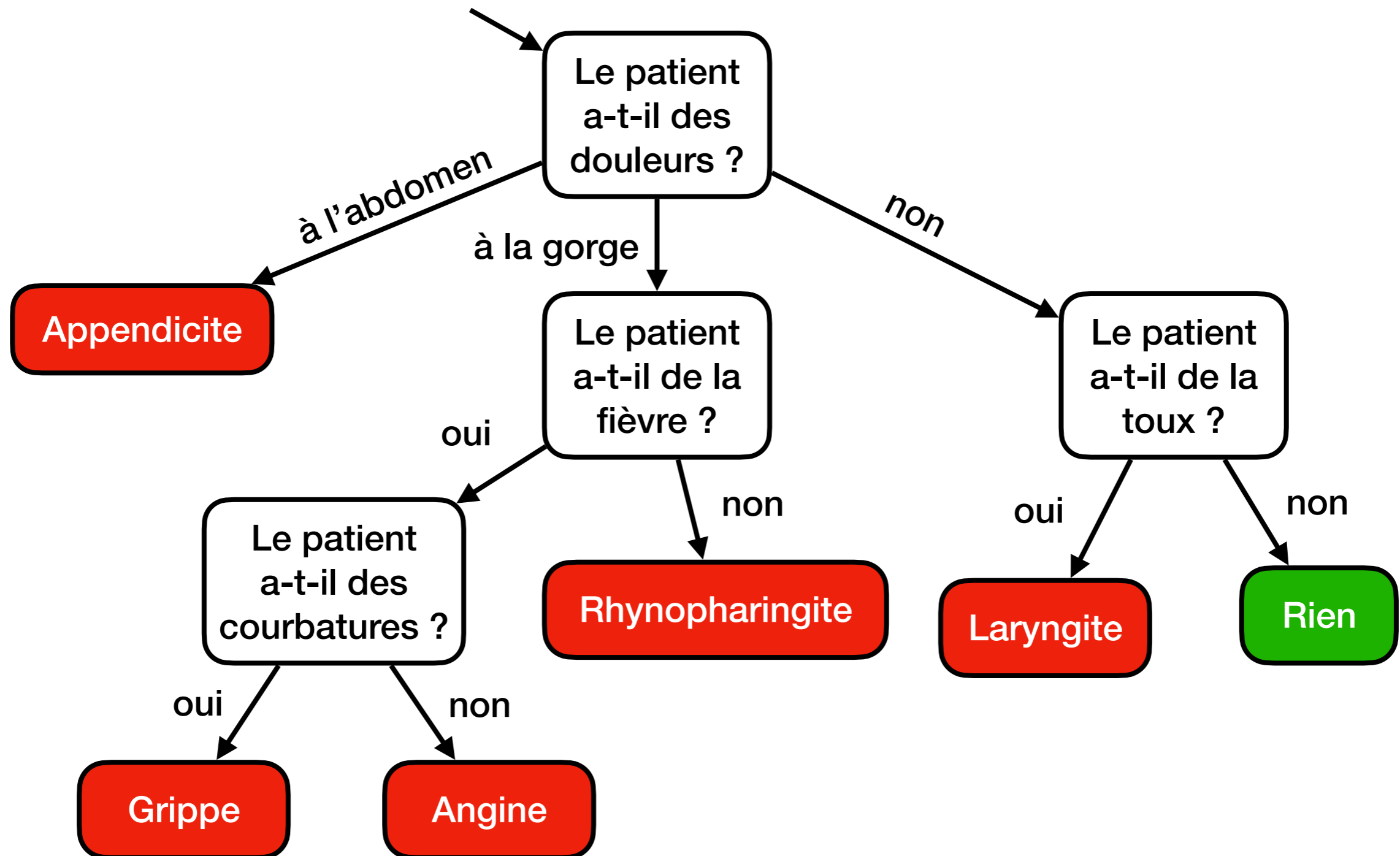


# Diagnostic de maladies





# Diagnostic de maladies



# Qui est-ce ?

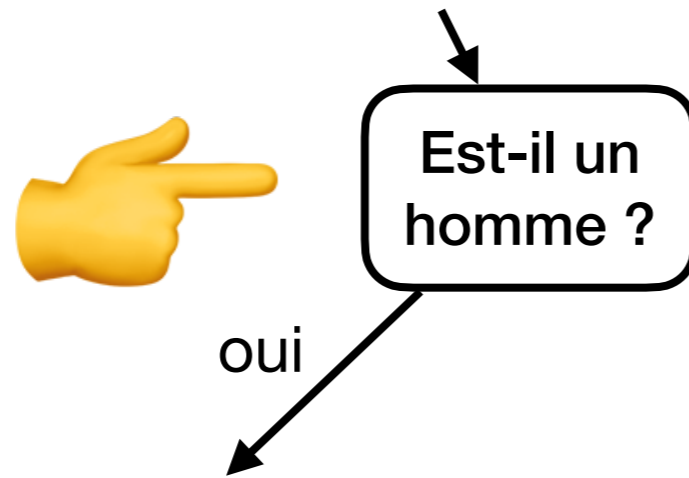


# Qui est-ce ?



↓  
Est-il un  
homme ?

# Qui est-ce ?





# Qui est-ce ?

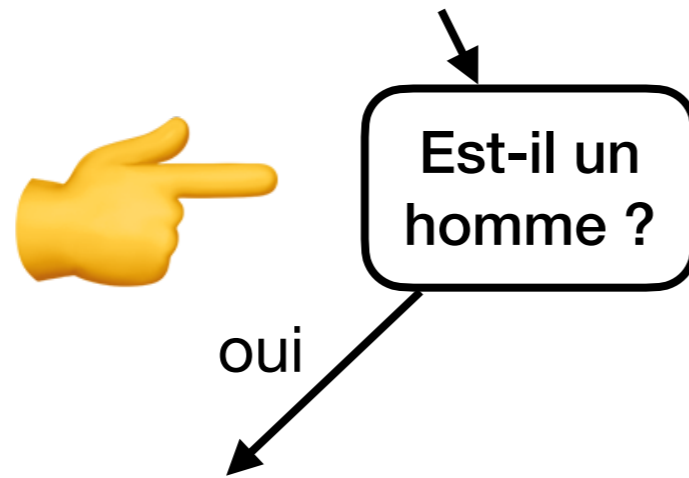




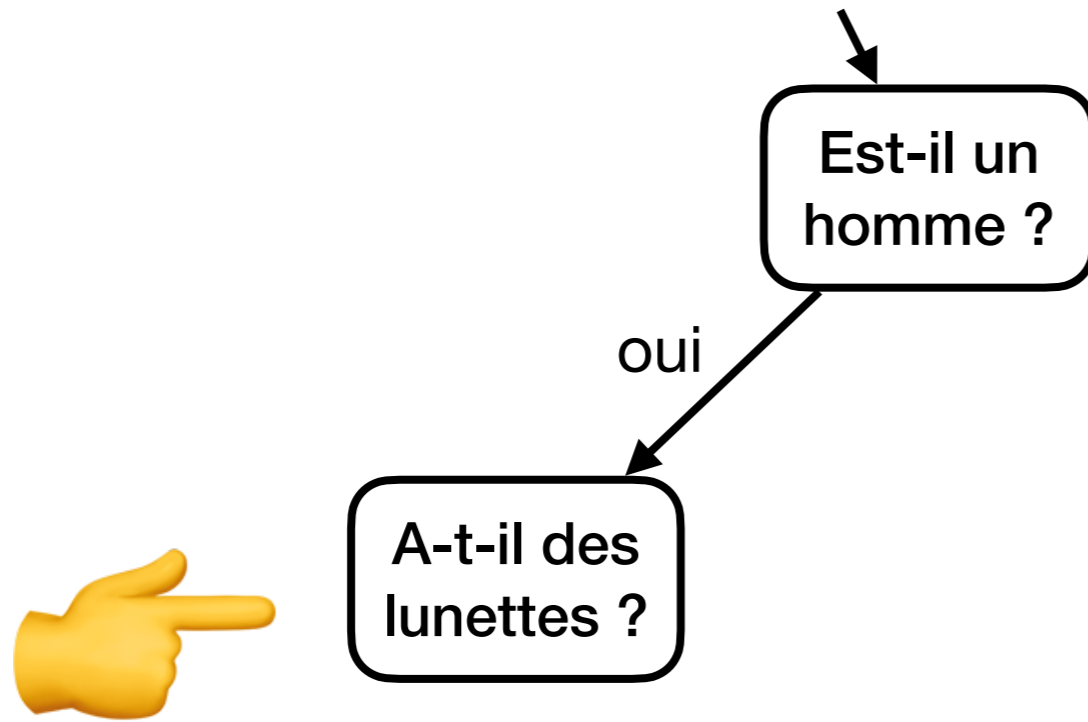
# Qui est-ce ?



# Qui est-ce ?

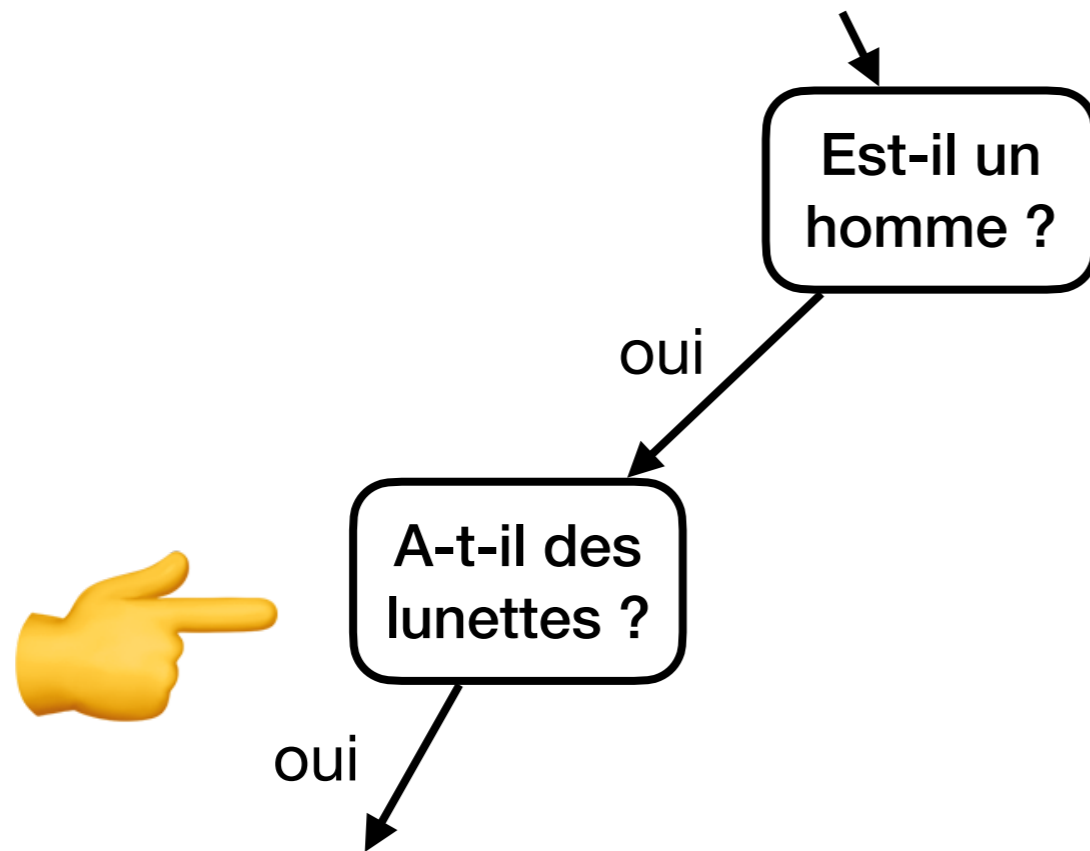


# Qui est-ce ?





# Qui est-ce ?



# Qui est-ce ?





# Qui est-ce ?



Alex



Alfred



Anita



Anne



Bernard



Bill



Charles



Claire



David



Eric



Frans



George



Herman



Joe



Maria



Max



Paul



Peter



Philip



Richard



Robert



Sam

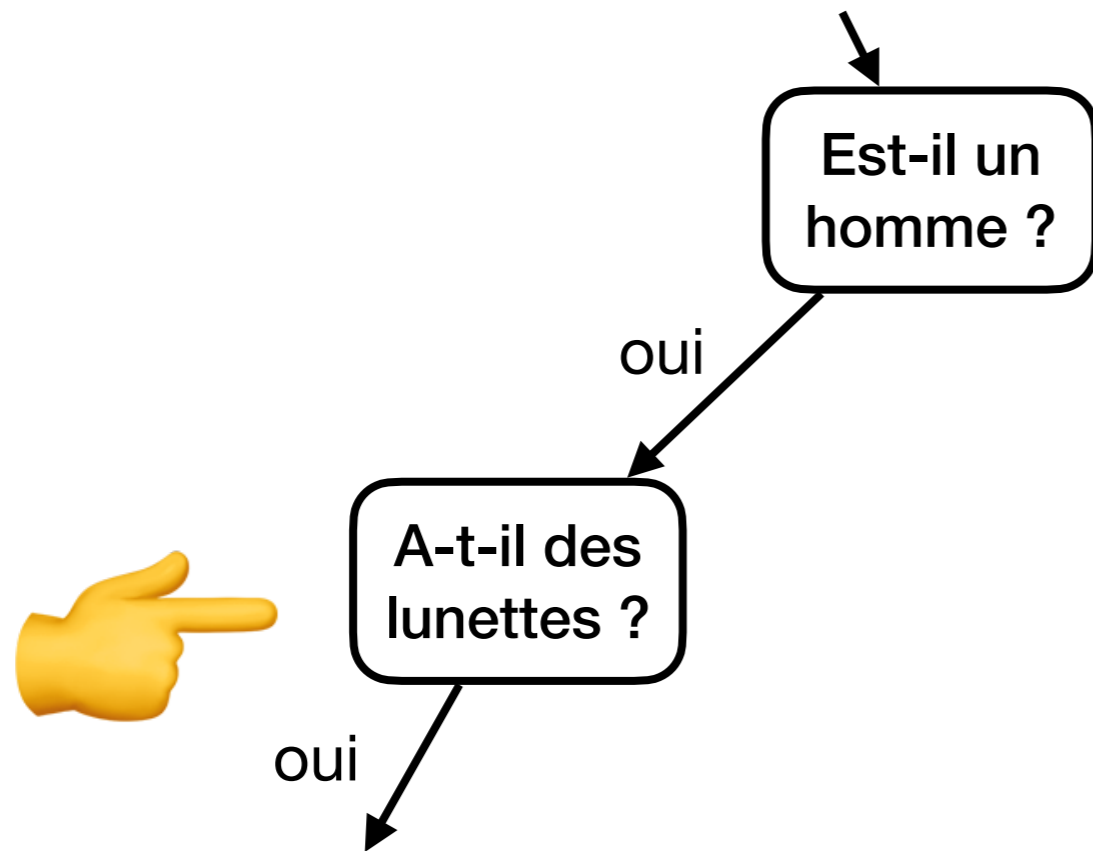


Susan

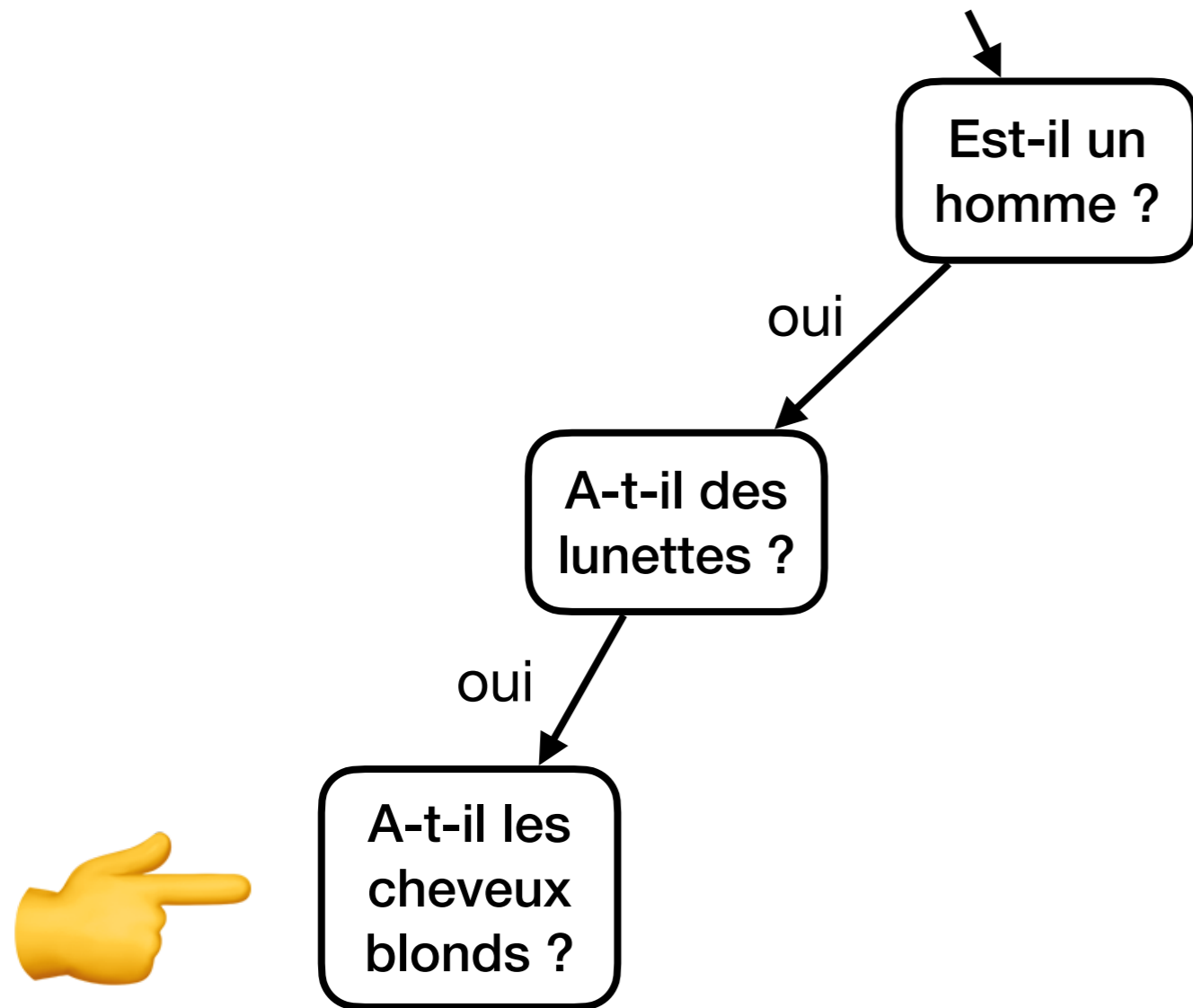


Tom

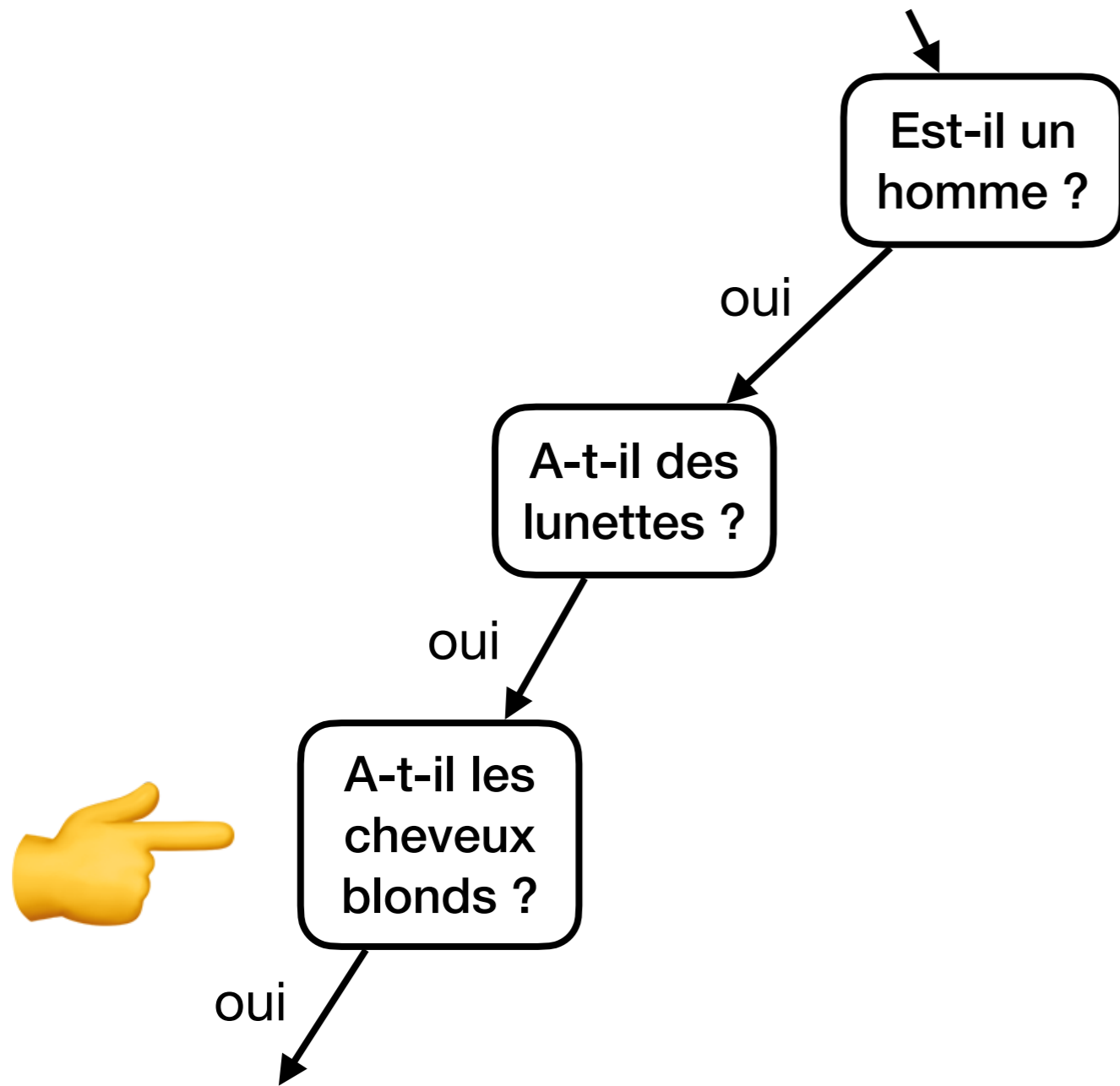
# Qui est-ce ?



# Qui est-ce ?



# Qui est-ce ?





# Qui est-ce ?



Alex



Alfred



Anita



Anne



Bernard



Bill



Charles



Claire



David



Eric



Frans



George



Herman



Joe



Maria



Max



Paul



Peter



Philip



Richard



Robert



Sam



Susan



Tom

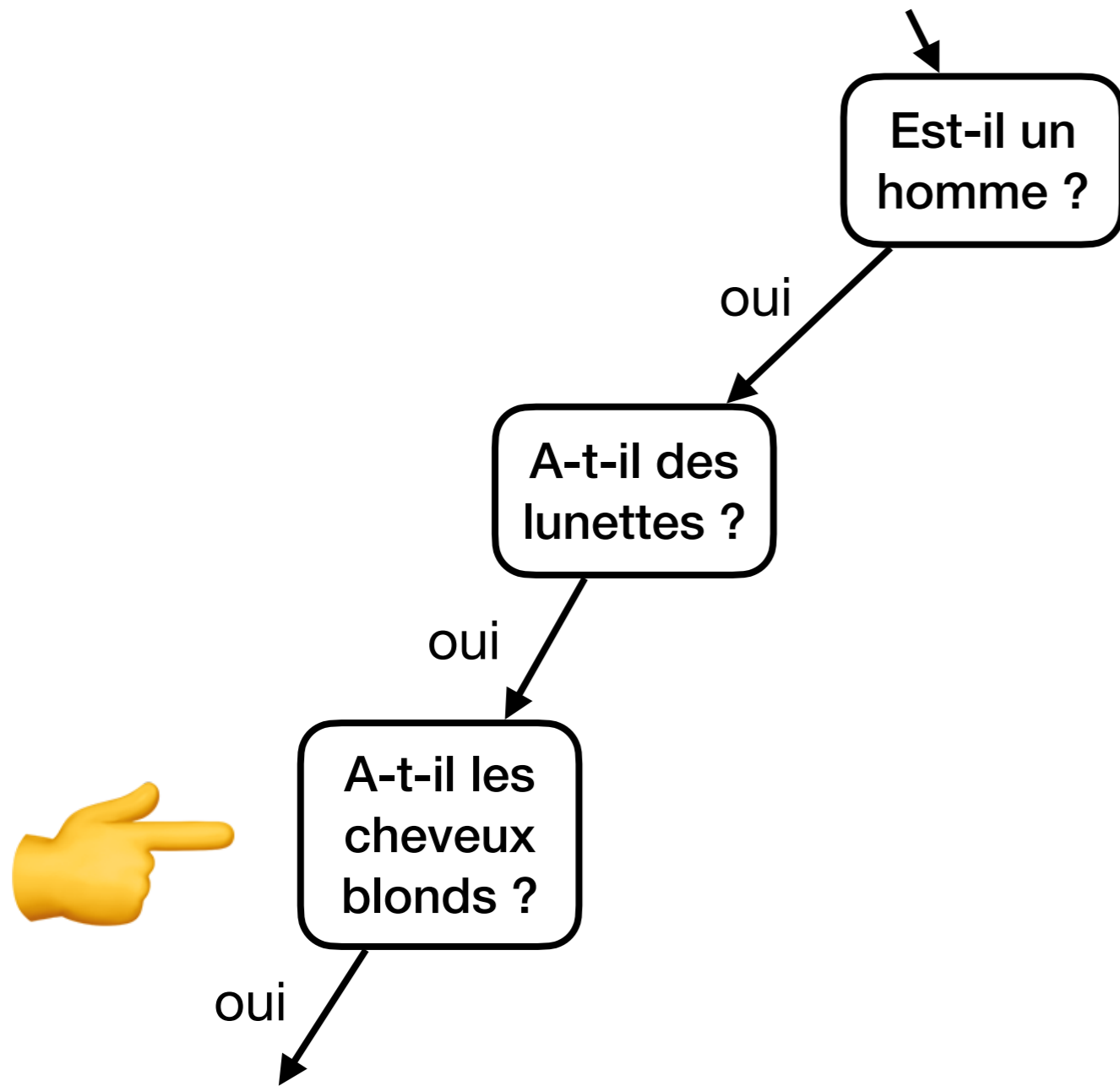


# Qui est-ce ?

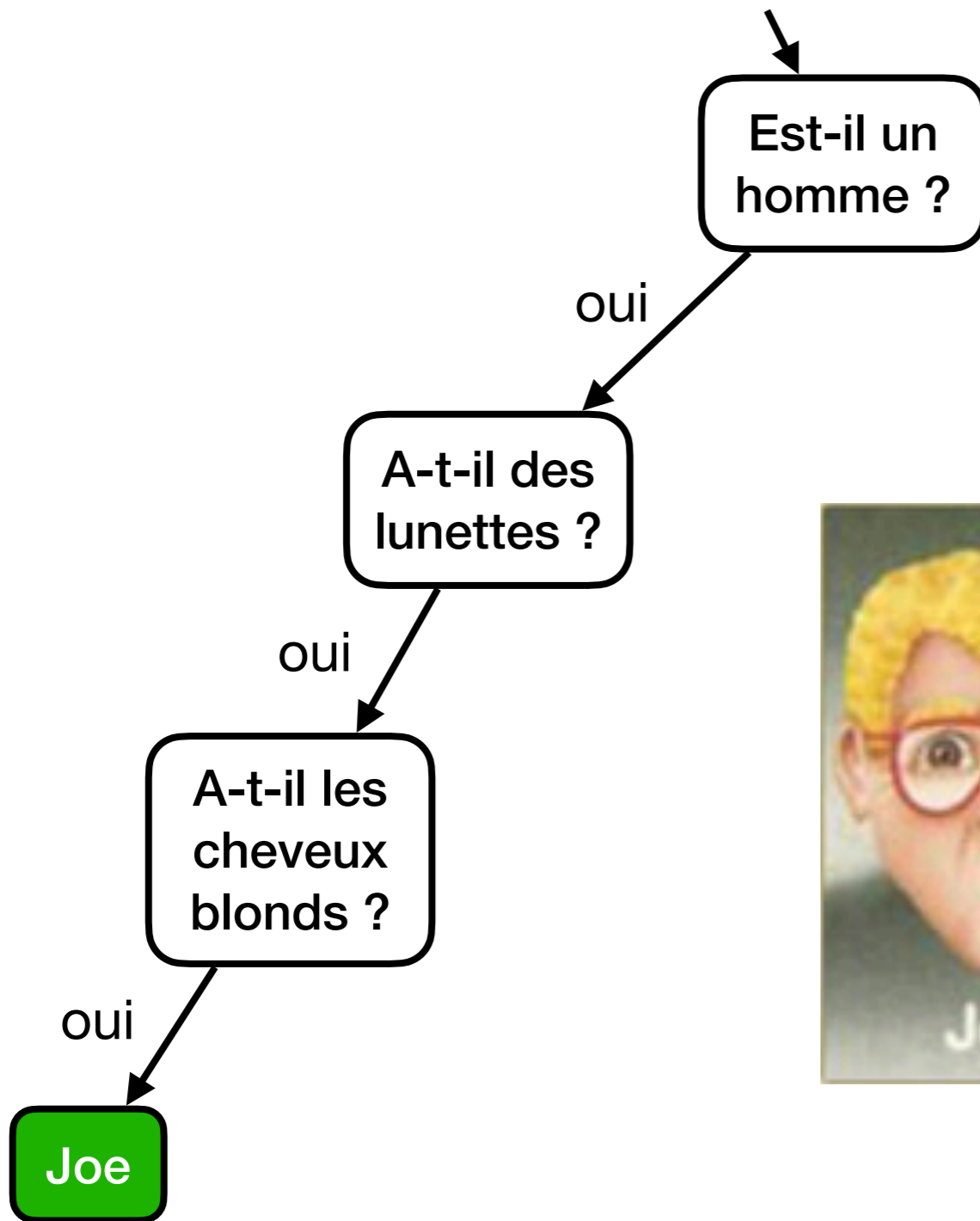




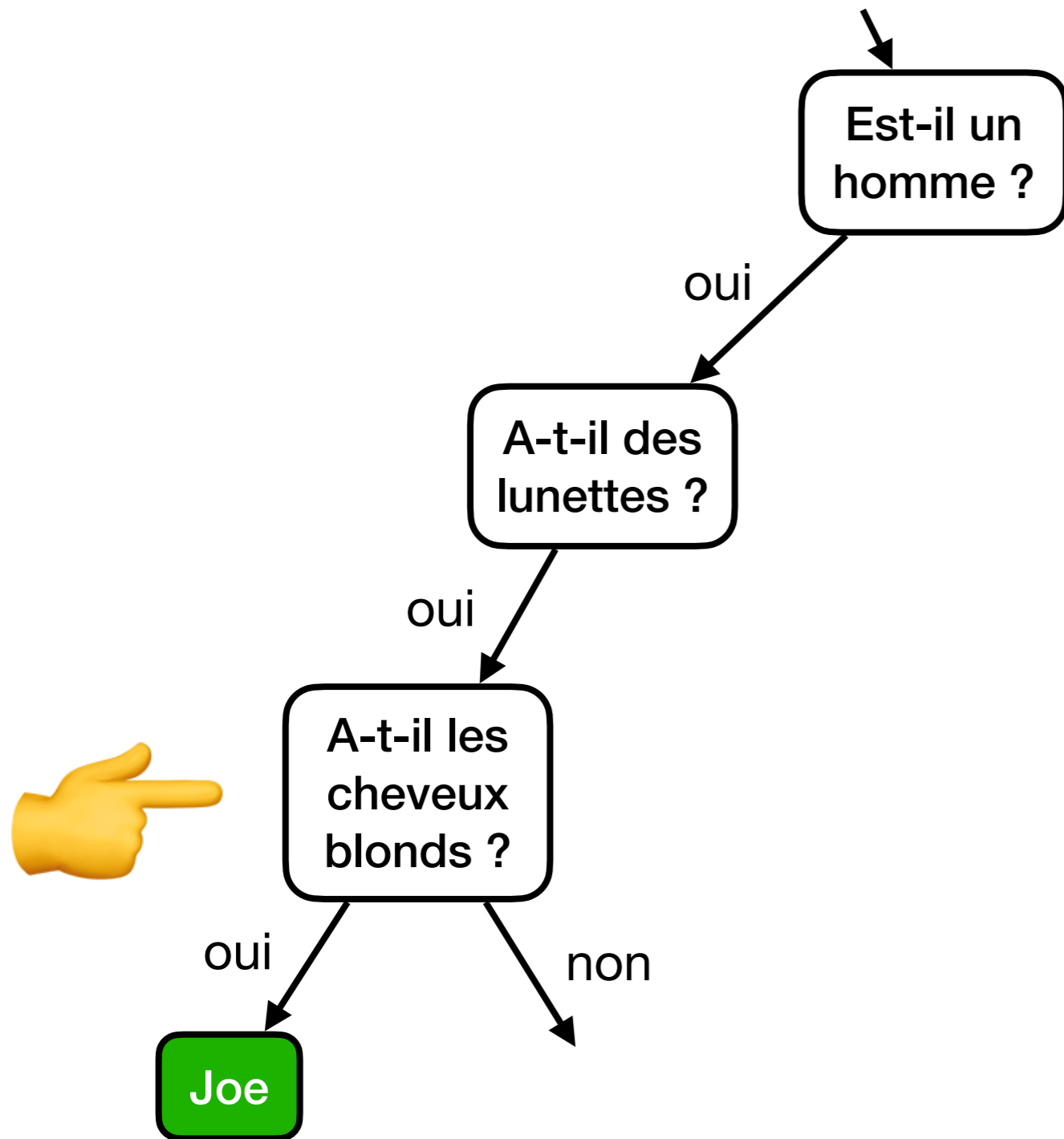
# Qui est-ce ?



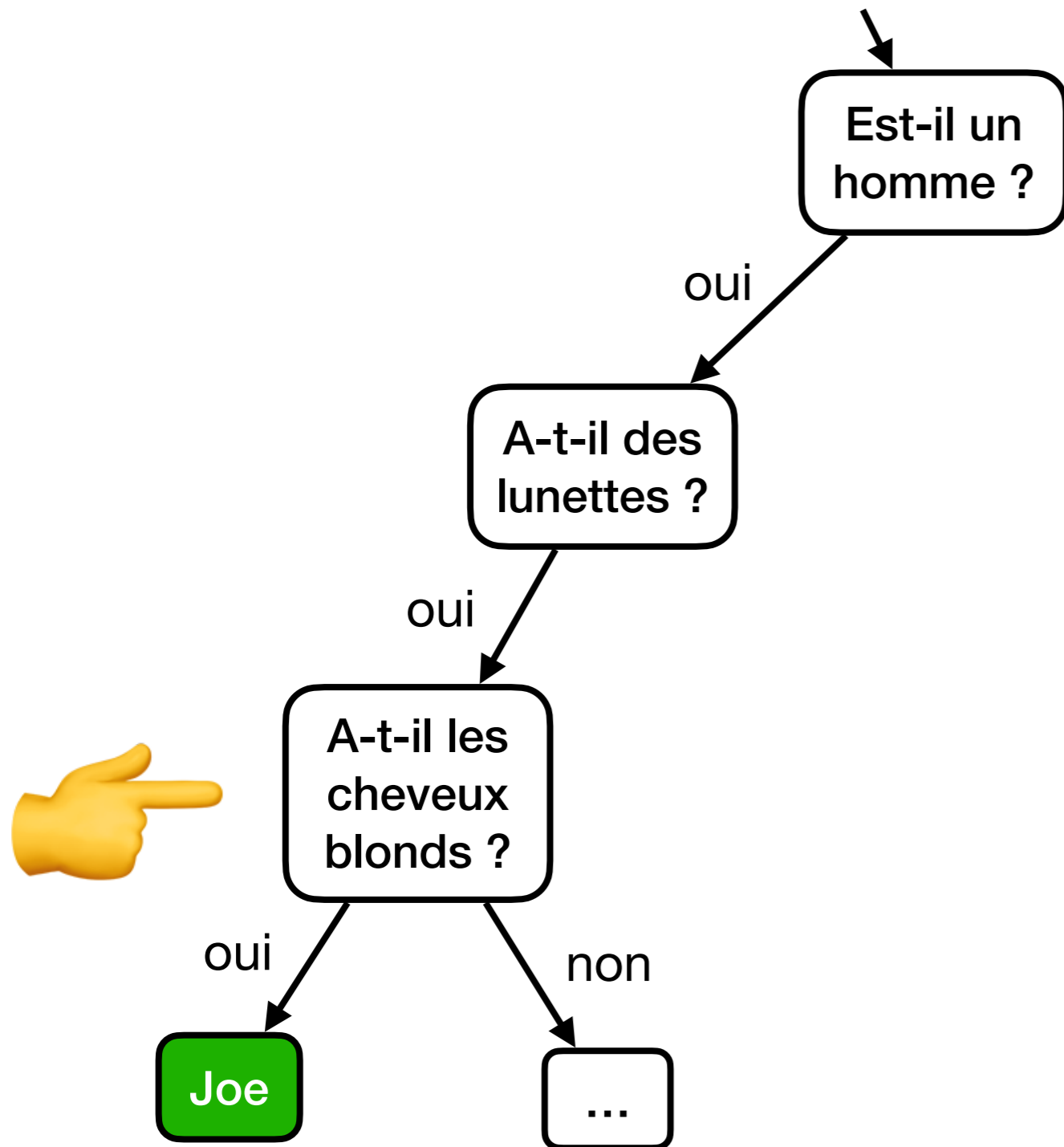
# Qui est-ce ?



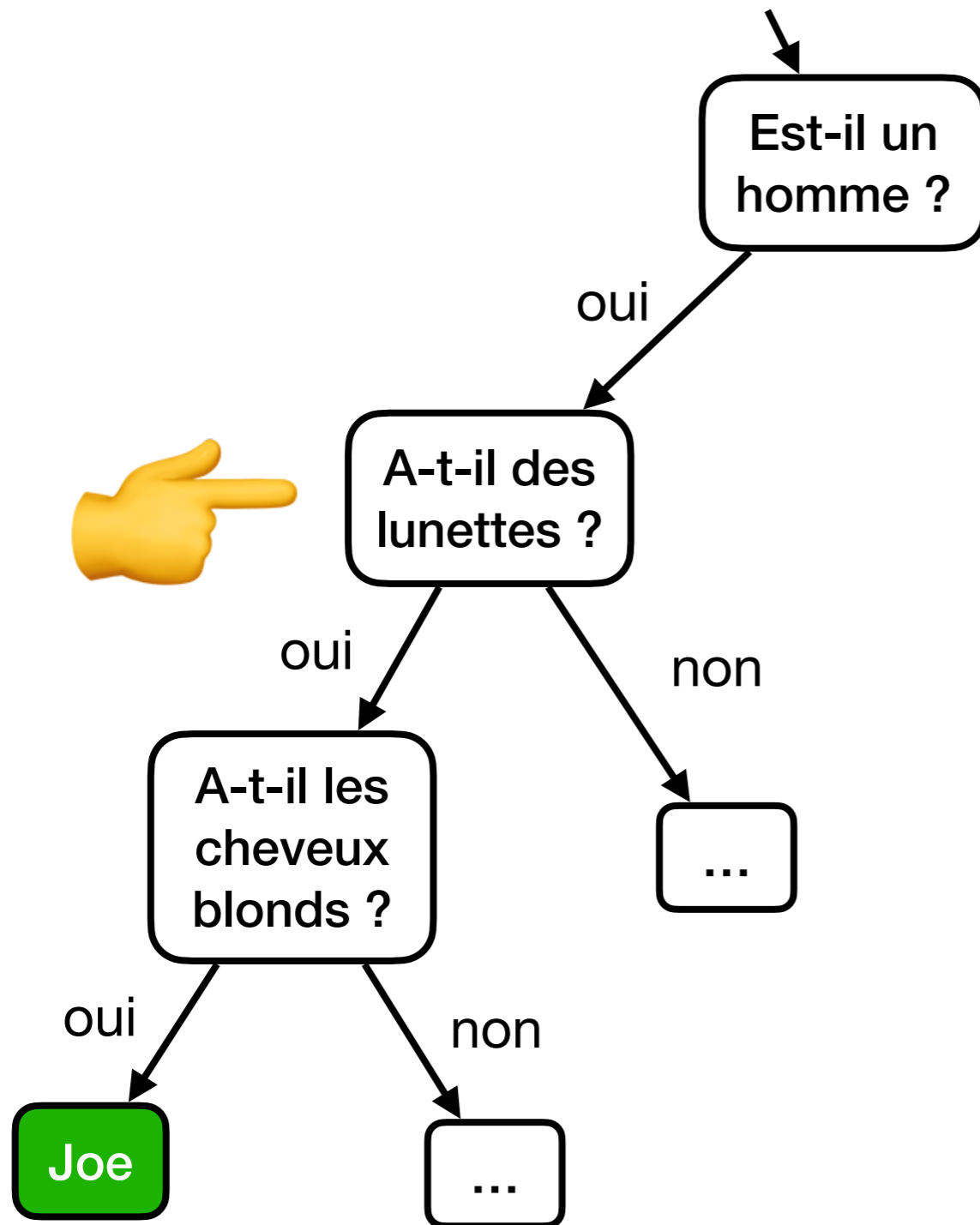
# Qui est-ce ?



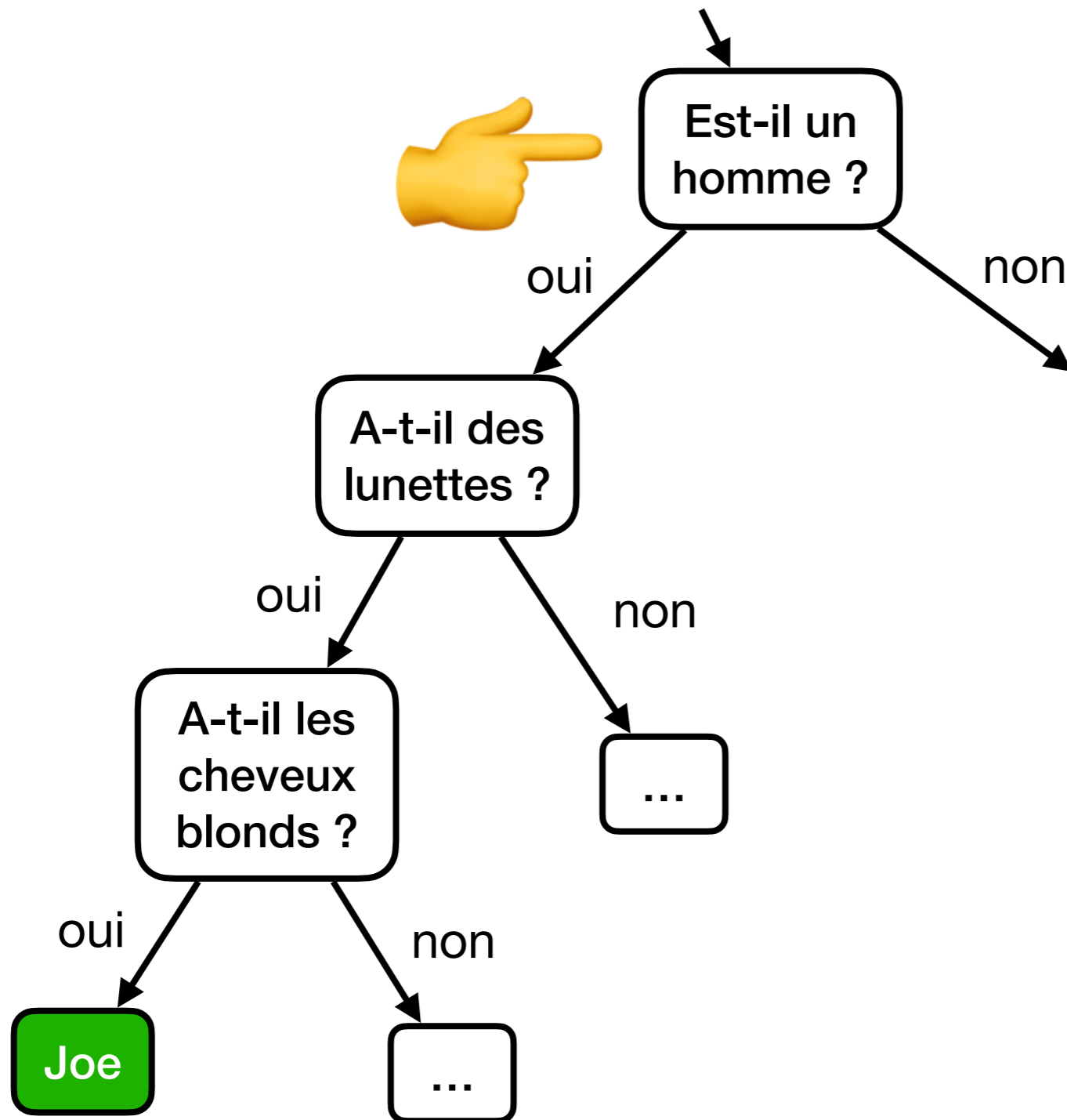
# Qui est-ce ?



# Qui est-ce ?



# Qui est-ce ?





# Qui est-ce ?



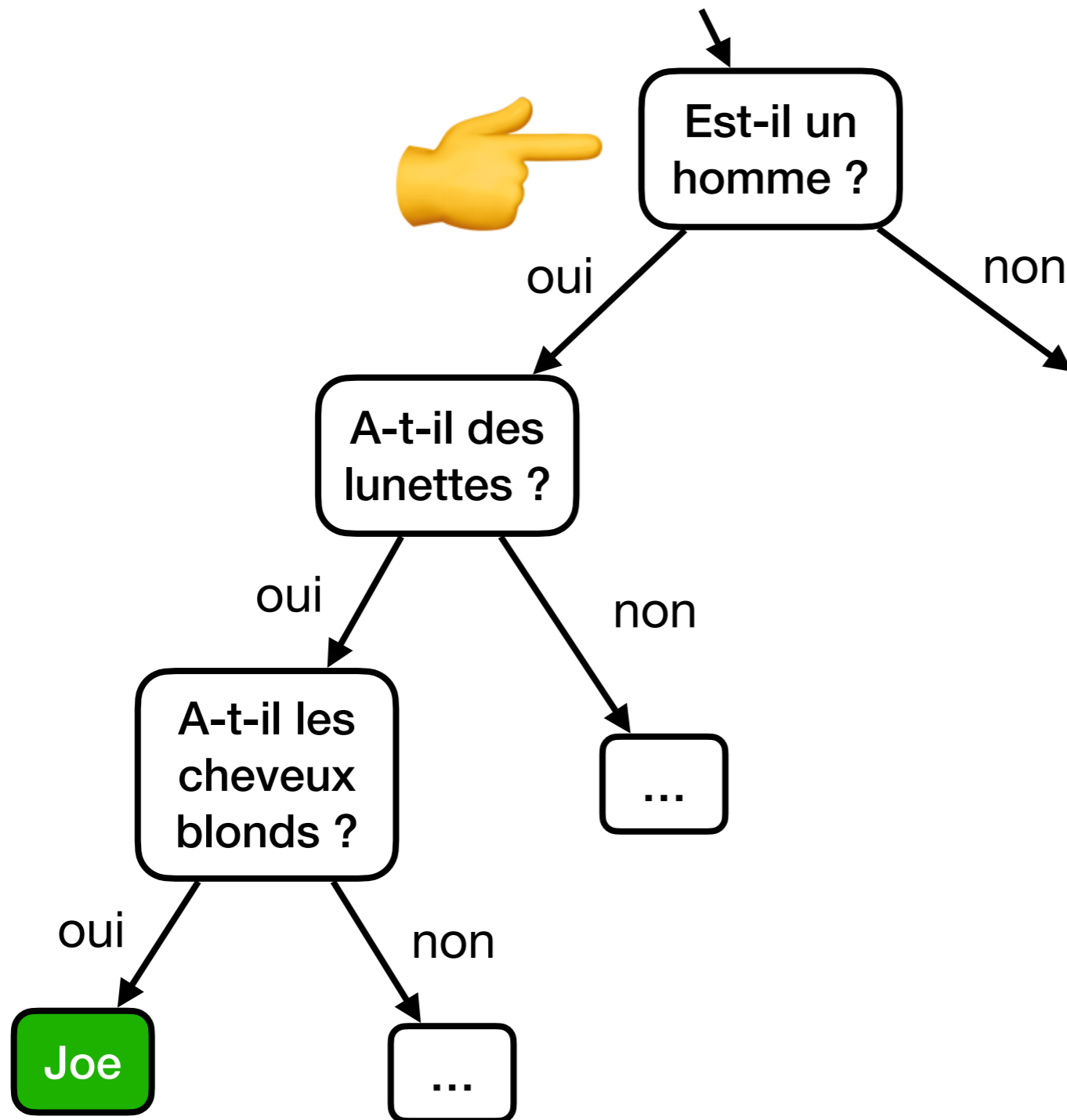


# Qui est-ce ?

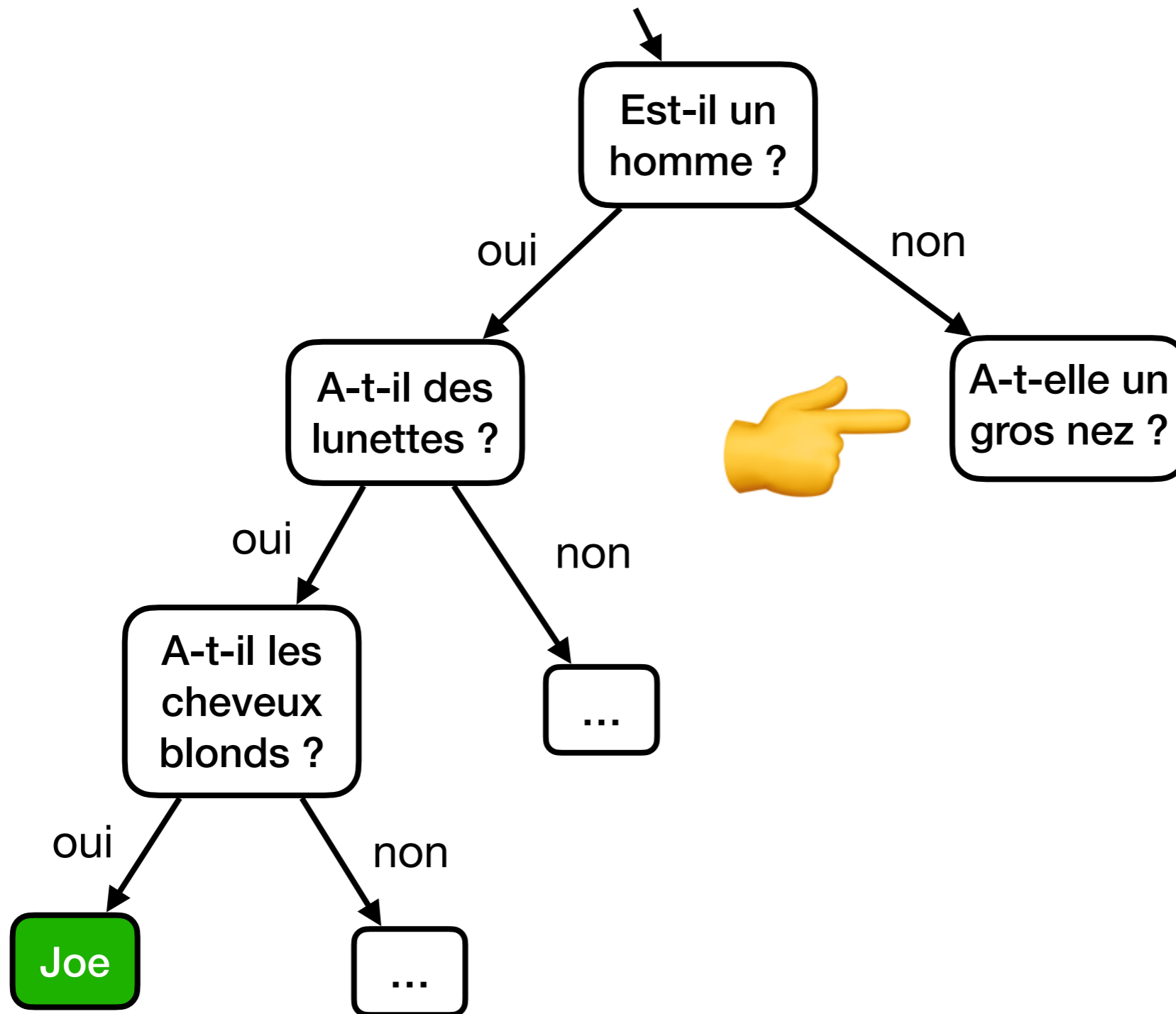




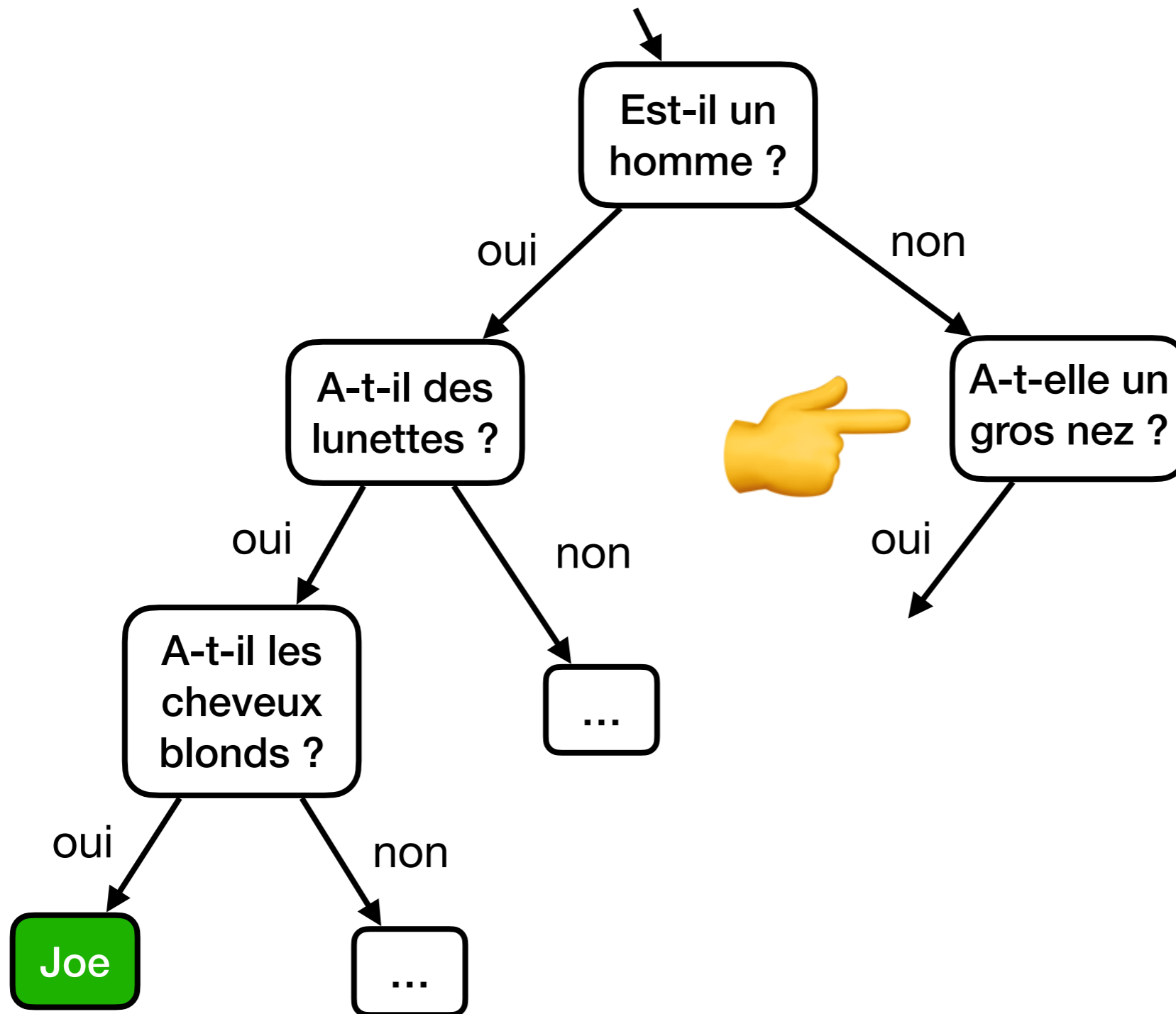
# Qui est-ce ?



# Qui est-ce ?



# Qui est-ce ?



# Qui est-ce ?



Alex



Alfred



Anita



Anne



Bernard



Bill



Charles



Claire



David



Eric



Frans



George



Herman



Joe



Maria



Max



Paul



Peter



Philip



Richard



Robert



Sam



Susan



Tom



# Qui est-ce ?



Alex



Alfred



Anita



Anne



Bernard



Bill



Charles



Claire



David



Eric



Frans



George



Herman



Joe



Maria



Max



Paul



Peter



Philip



Richard



Robert



Sam

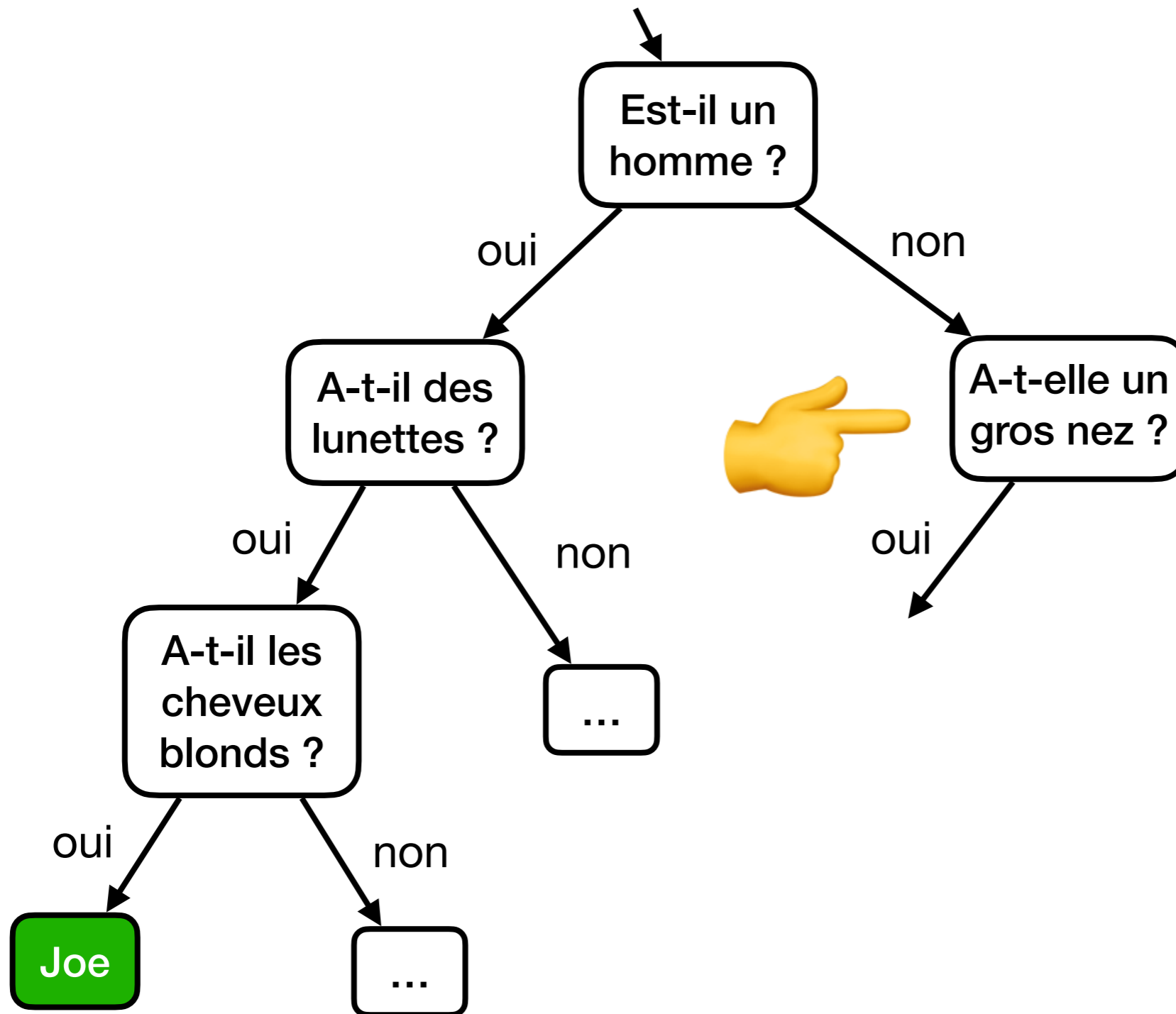


Susan

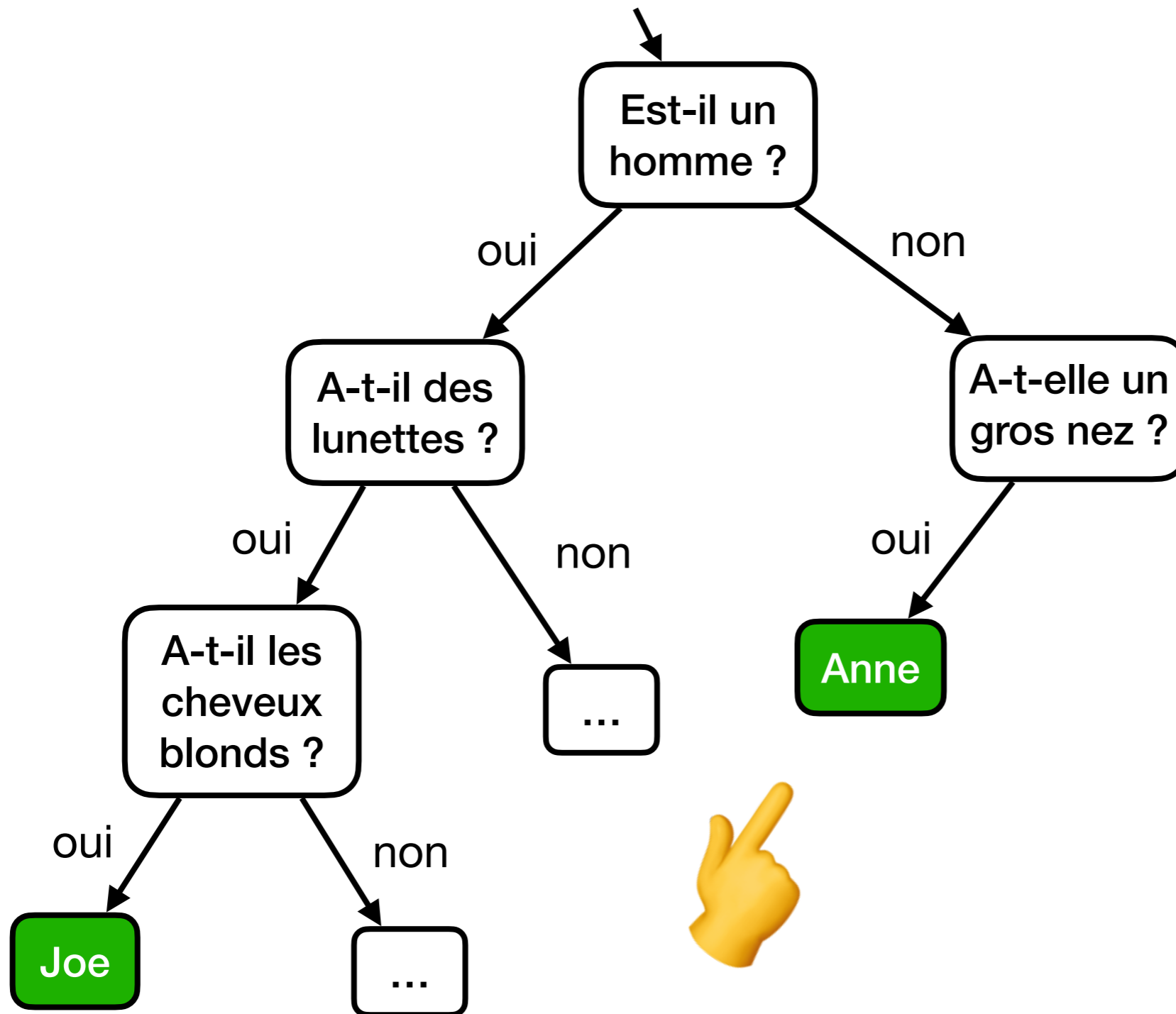


Tom

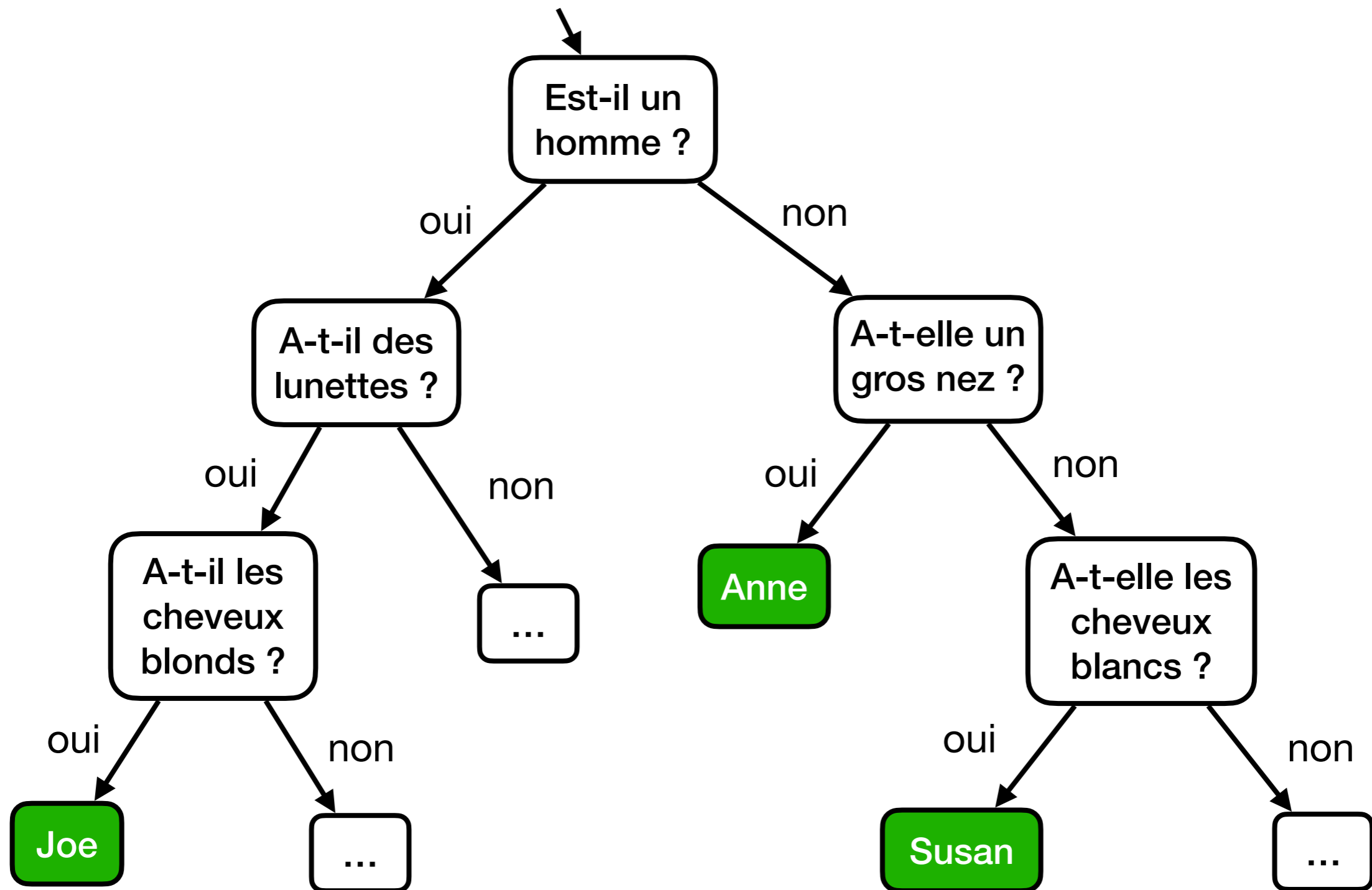
# Qui est-ce ?



# Qui est-ce ?



# Qui est-ce ?





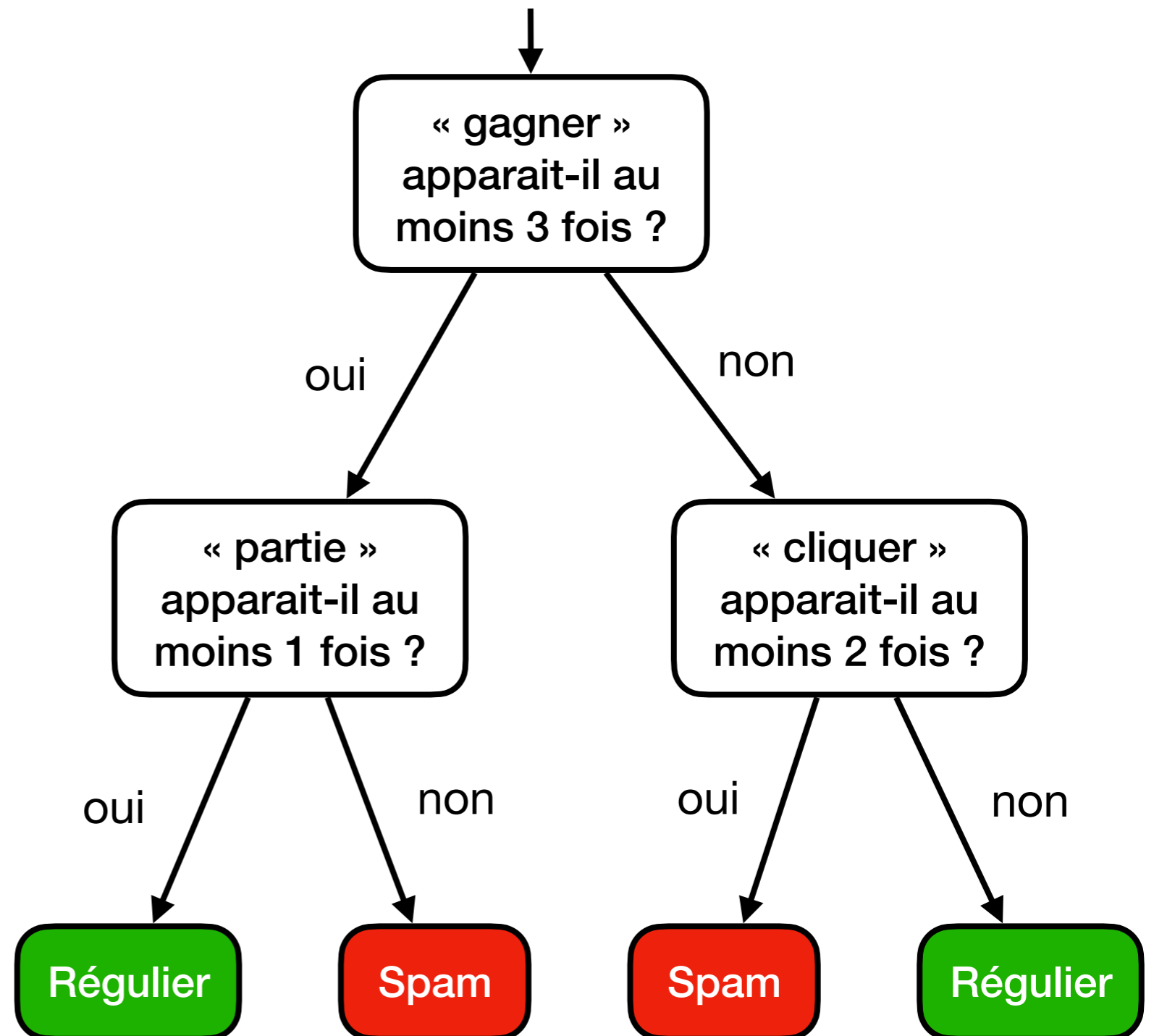
# Détection de spam

## Message 1

Bravo,  
Vous venez de gagner à notre grand tirage internet.  
Cliquez ici pour recevoir 1.000.000 de dollars !!!  
Et pour gagner d'autres cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me qualifie directement. Si je ne gagne pas demain mais que je gagne la suivante, je monte en pool 3 l'année prochaine !  
Bises, Sandra



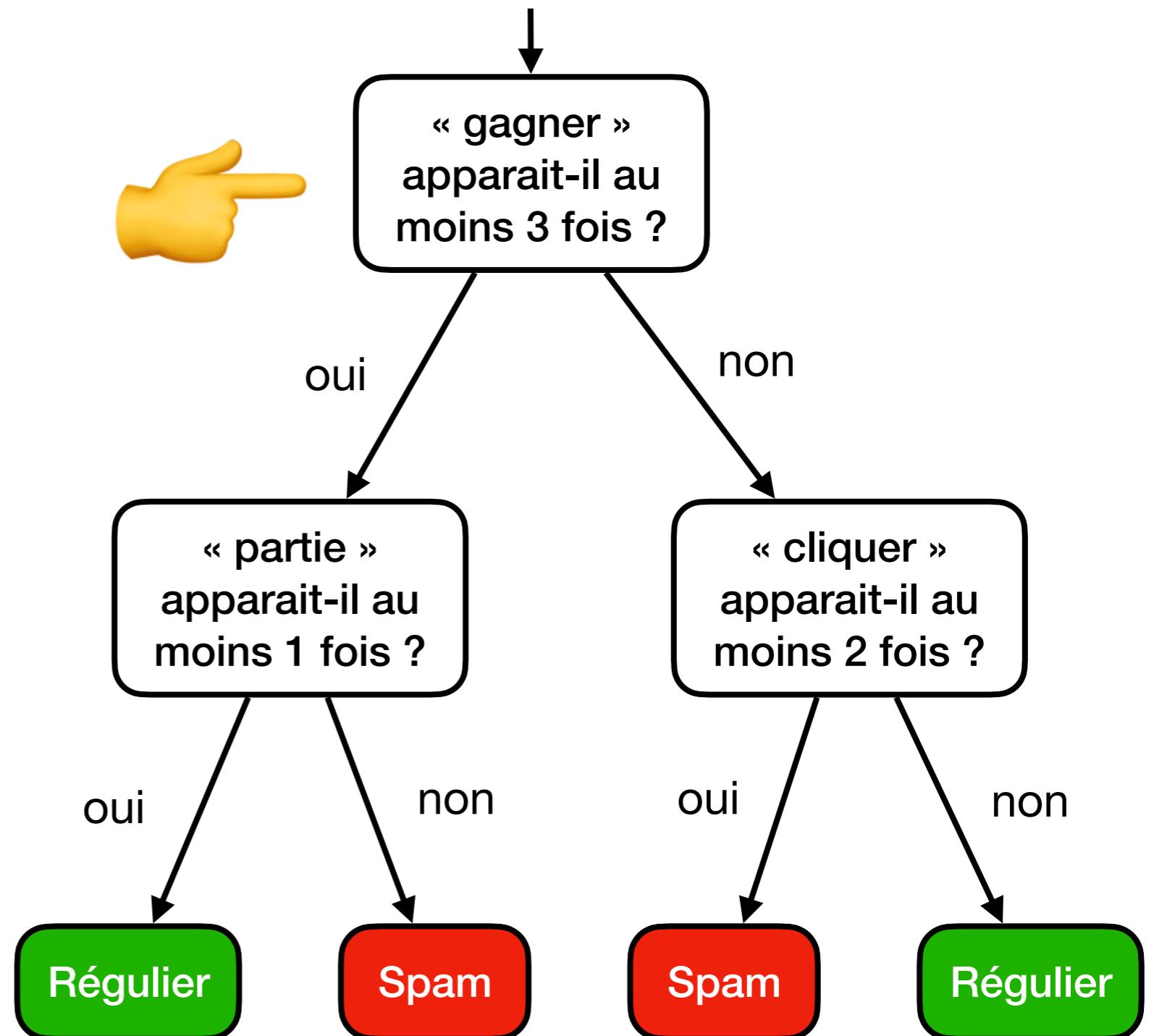
# Détection de spam

## Message 1

Bravo,  
Vous venez de gagner à notre grand tirage internet.  
Cliquez ici pour recevoir 1.000.000 de dollars !!!  
Et pour gagner d'autres cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me qualifie directement. Si je ne gagne pas demain mais que je gagne la suivante, je monte en pool 3 l'année prochaine !  
Bises, Sandra



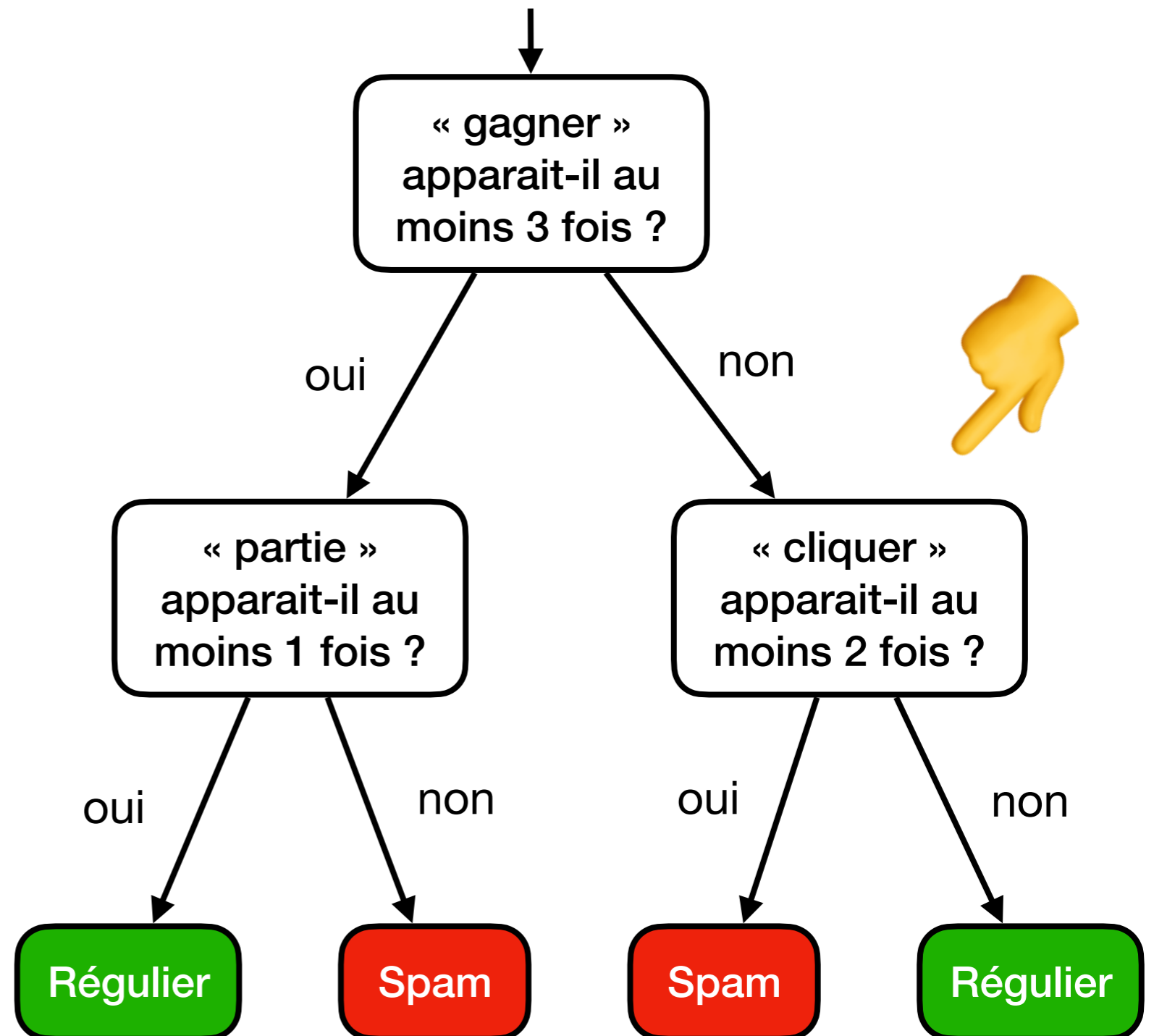
# Détection de spam

## Message 1

Bravo,  
Vous venez de gagner à notre grand tirage internet.  
Cliquez ici pour recevoir 1.000.000 de dollars !!!  
Et pour gagner d'autres cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me qualifie directement. Si je ne gagne pas demain mais que je gagne la suivante, je monte en pool 3 l'année prochaine !  
Bises, Sandra



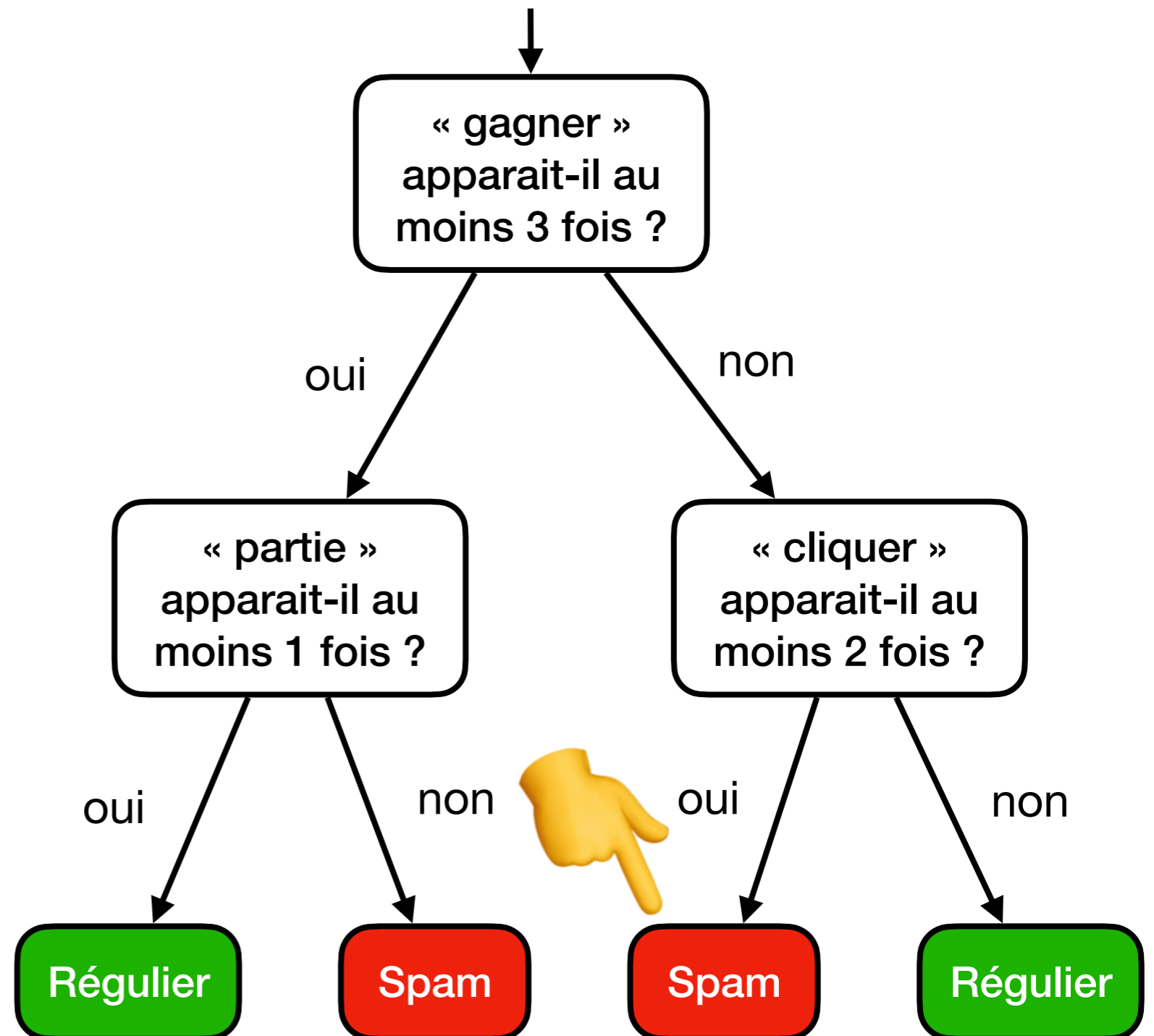
# Détection de spam

## Message 1

Bravo,  
Vous venez de gagner à notre grand tirage internet.  
Cliquez ici pour recevoir 1.000.000 de dollars !!!  
Et pour gagner d'autres cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me qualifie directement. Si je ne gagne pas demain mais que je gagne la suivante, je monte en pool 3 l'année prochaine !  
Bises, Sandra



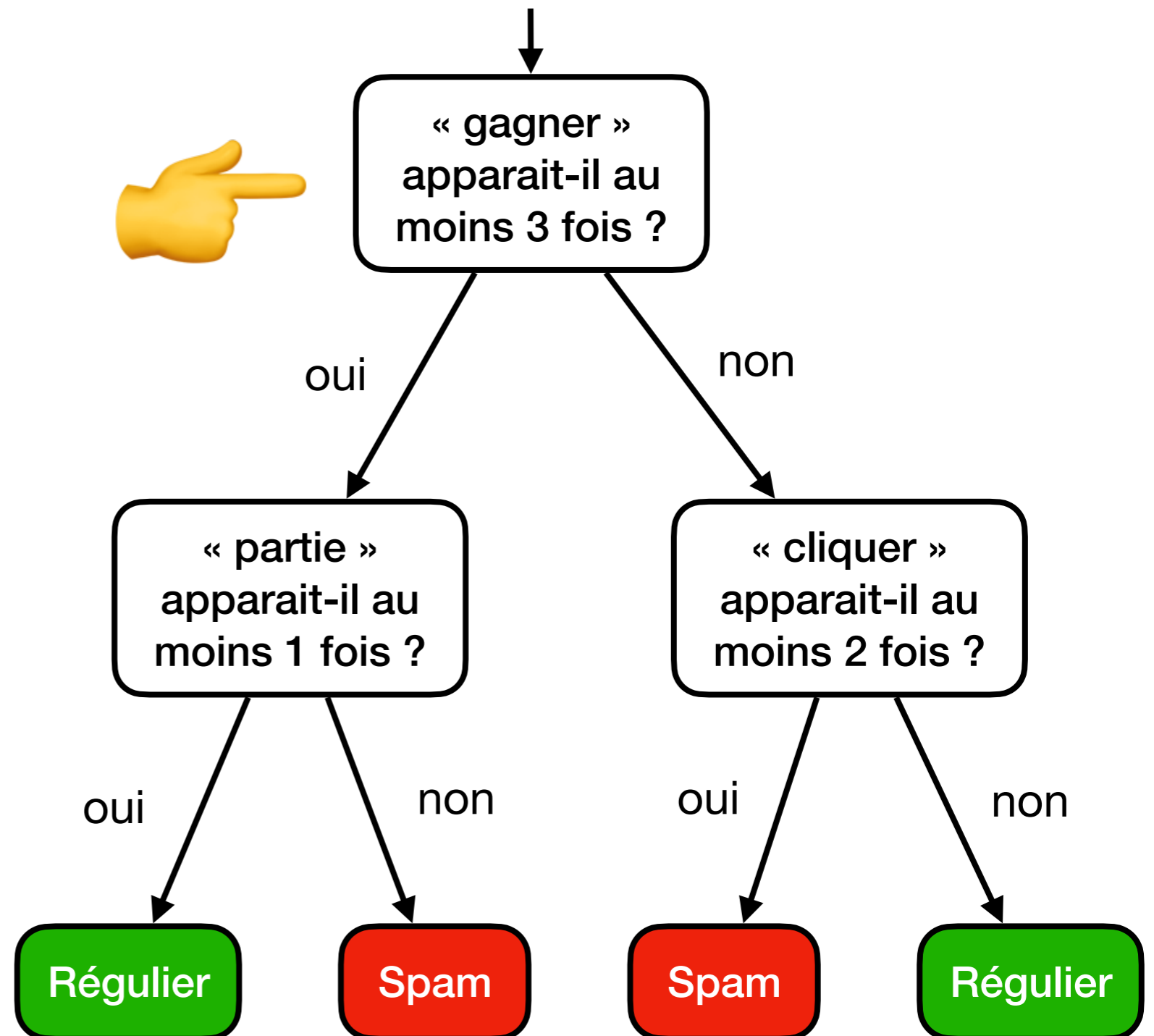
# Détection de spam

## Message 1

Bravo  
Vous venez de gagner à notre  
grand tirage. Cliquez ici pour recevoir  
1.000.000 de francs !!!  
Et pour gagner de plus  
cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me  
qualifie directement. Si je ne  
gagne pas demain mais que  
je gagne la suivante, je monte  
en pool 3 l'année prochaine !  
Bises, Sandra



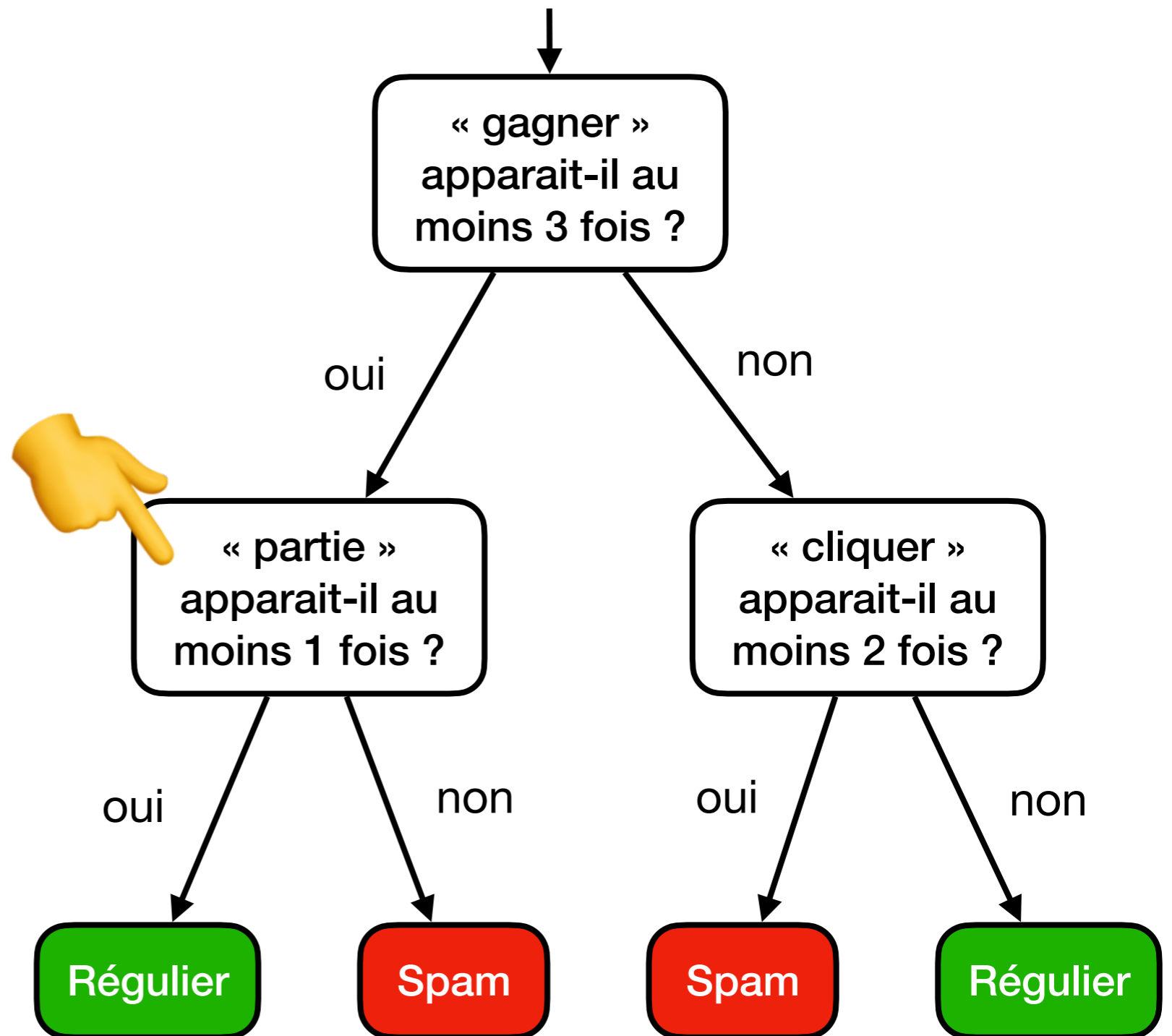
# Détection de spam

## Message 1

Bravo  
Vous venez de gagner à notre  
grand tirage. Cliquez ici pour recevoir  
1.000.000 de francs !!!  
Et pour gagner de plus  
cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me  
qualifie directement. Si je ne  
gagne pas demain mais que  
je gagne la suivante, je monte  
en pool 3 l'année prochaine !  
Bises, Sandra





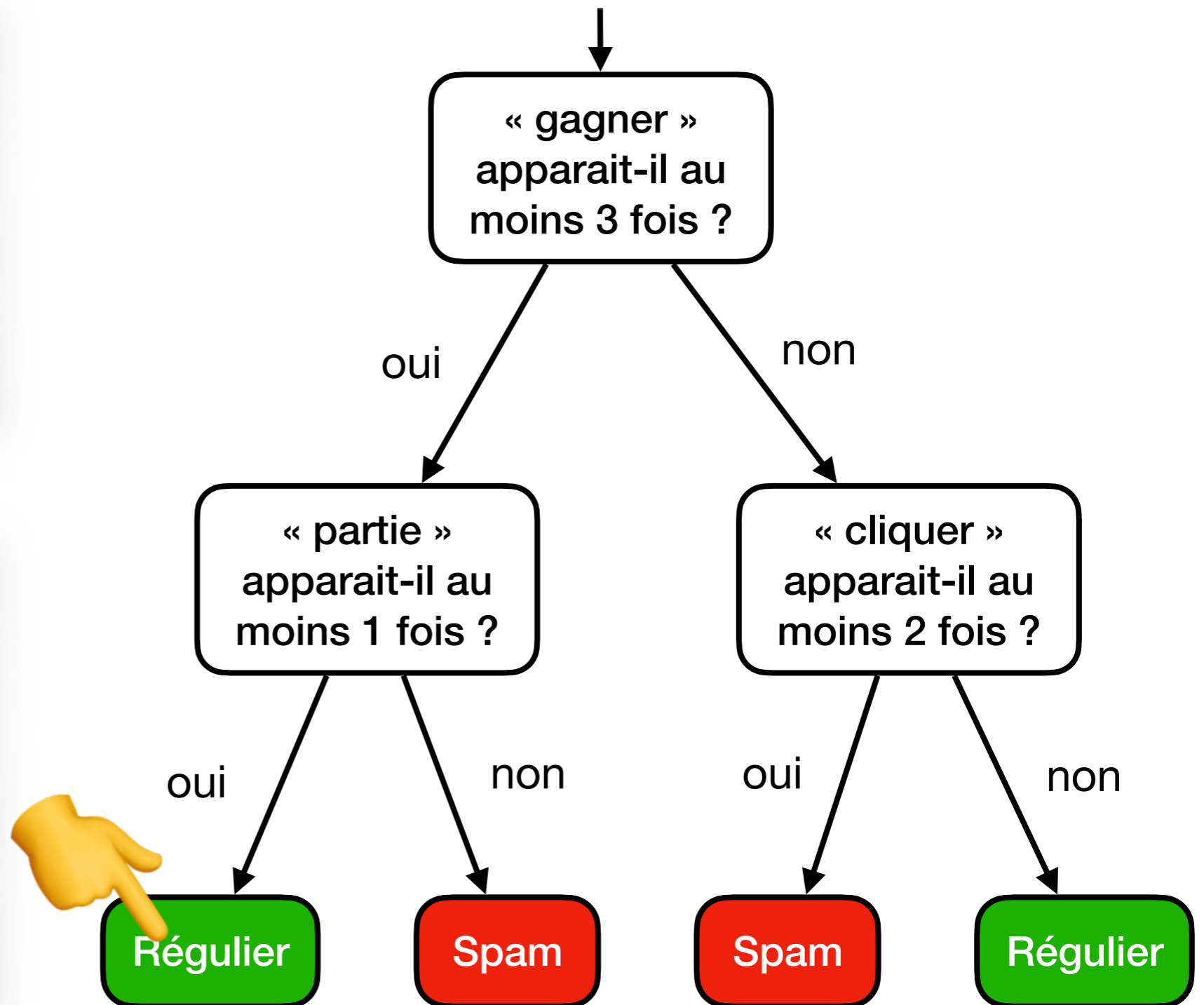
# Détection de spam

## Message 1

Bravo  
Vous venez de gagner à notre  
grand tirage. Cliquez ici pour recevoir  
1.000.000 de francs !!!  
Et pour gagner de plus  
cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me  
qualifie directement. Si je ne  
gagne pas demain mais que  
je gagne la suivante, je monte  
en pool 3 l'année prochaine !  
Bises, Sandra



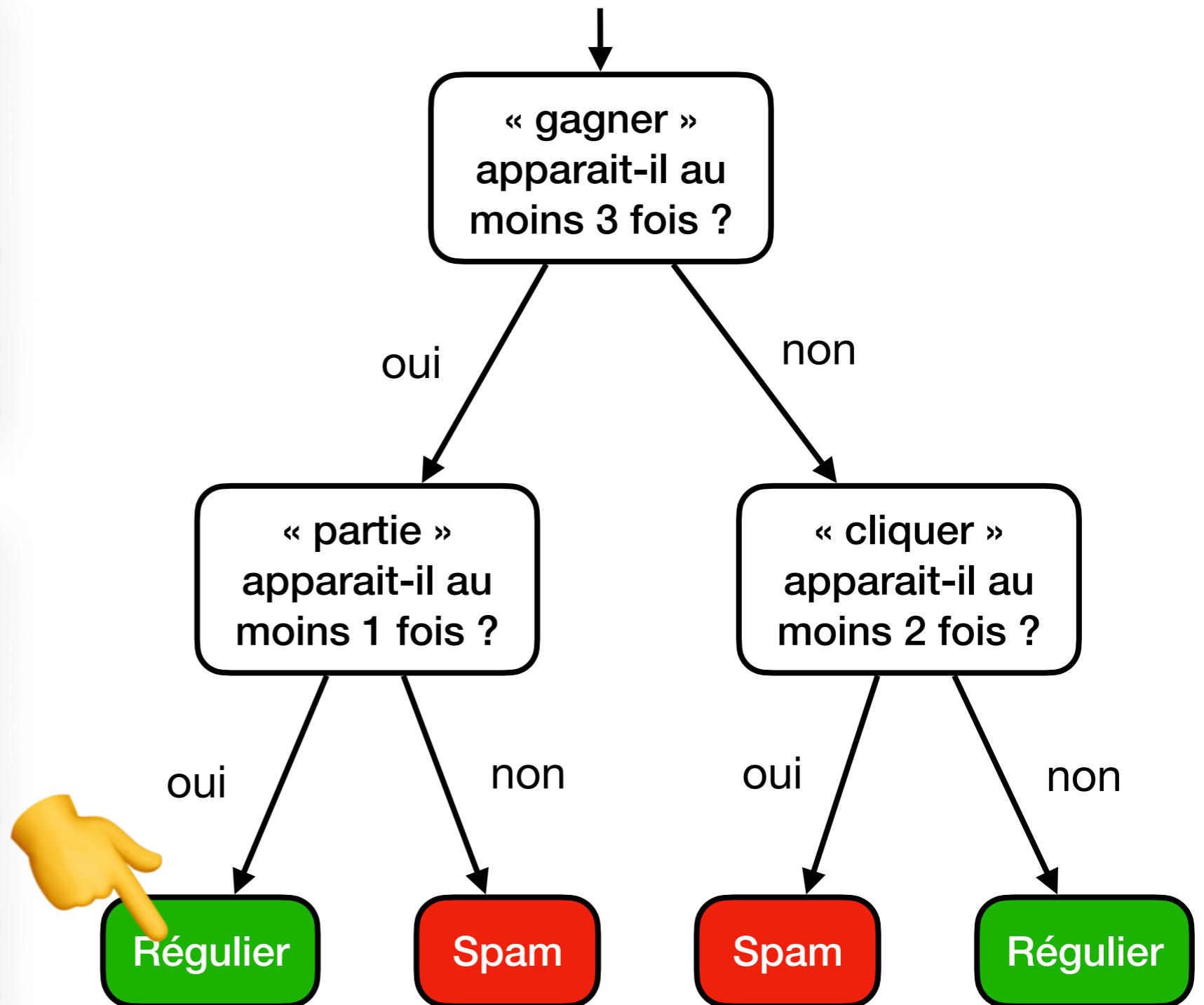
# Détection de spam

## Message 1

Bravo  
Vous venez de gagner à notre  
grand tirage. Cliquez ici pour recevoir  
1.000.000 de francs !!!  
Et pour gagner de plus gros  
cadeaux, cliquez ici.

## Message 2

Salut Benoît,  
Demain je joue contre Bruno.  
Si je gagne la partie, je me  
qualifie directement. Si je ne  
gagne pas, je monte en pouce  
en pouce la chaîne !  
Bises, Sandra





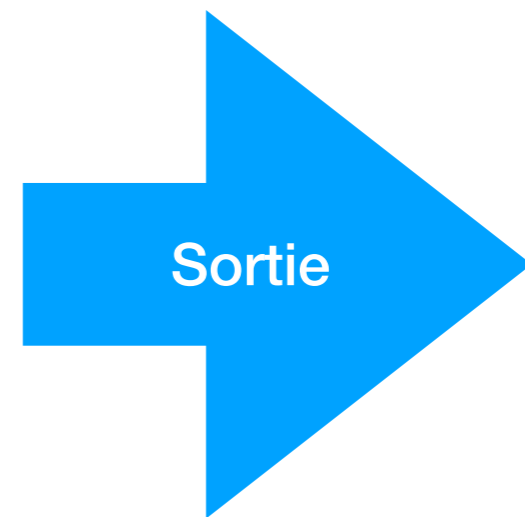
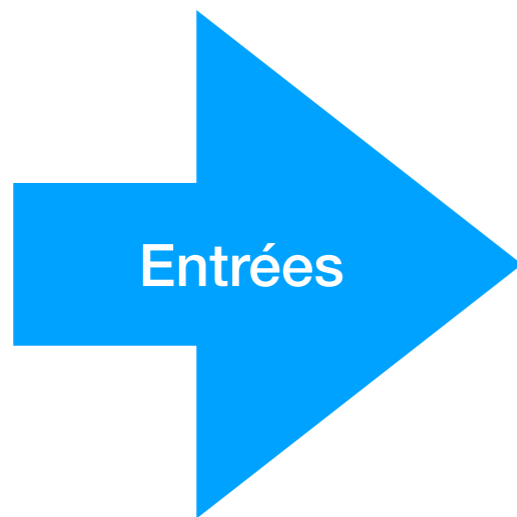
# Taille des arbres en pratique



# Taille des arbres en pratique



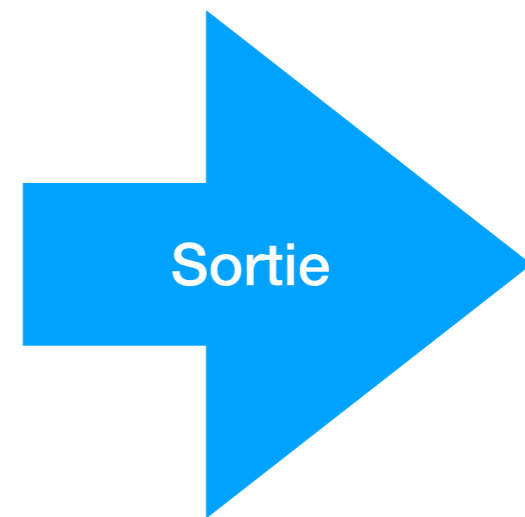
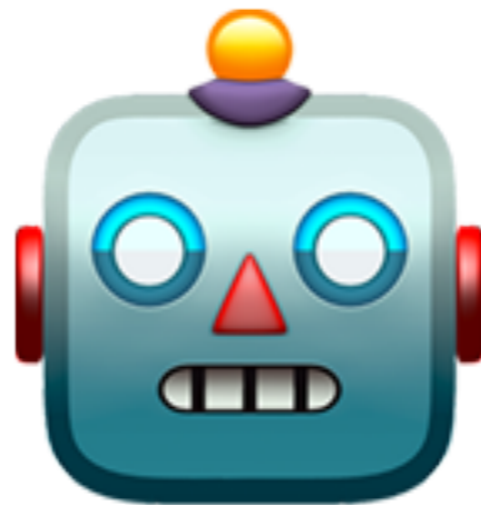
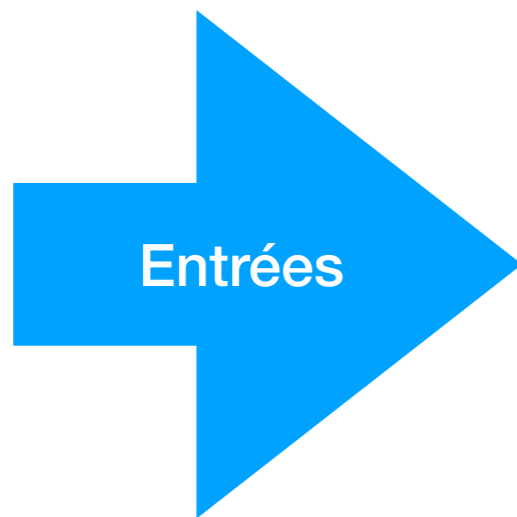
# Que peut-on calculer avec un arbre de décision ?



**Réponses aux questions**

**Décision**

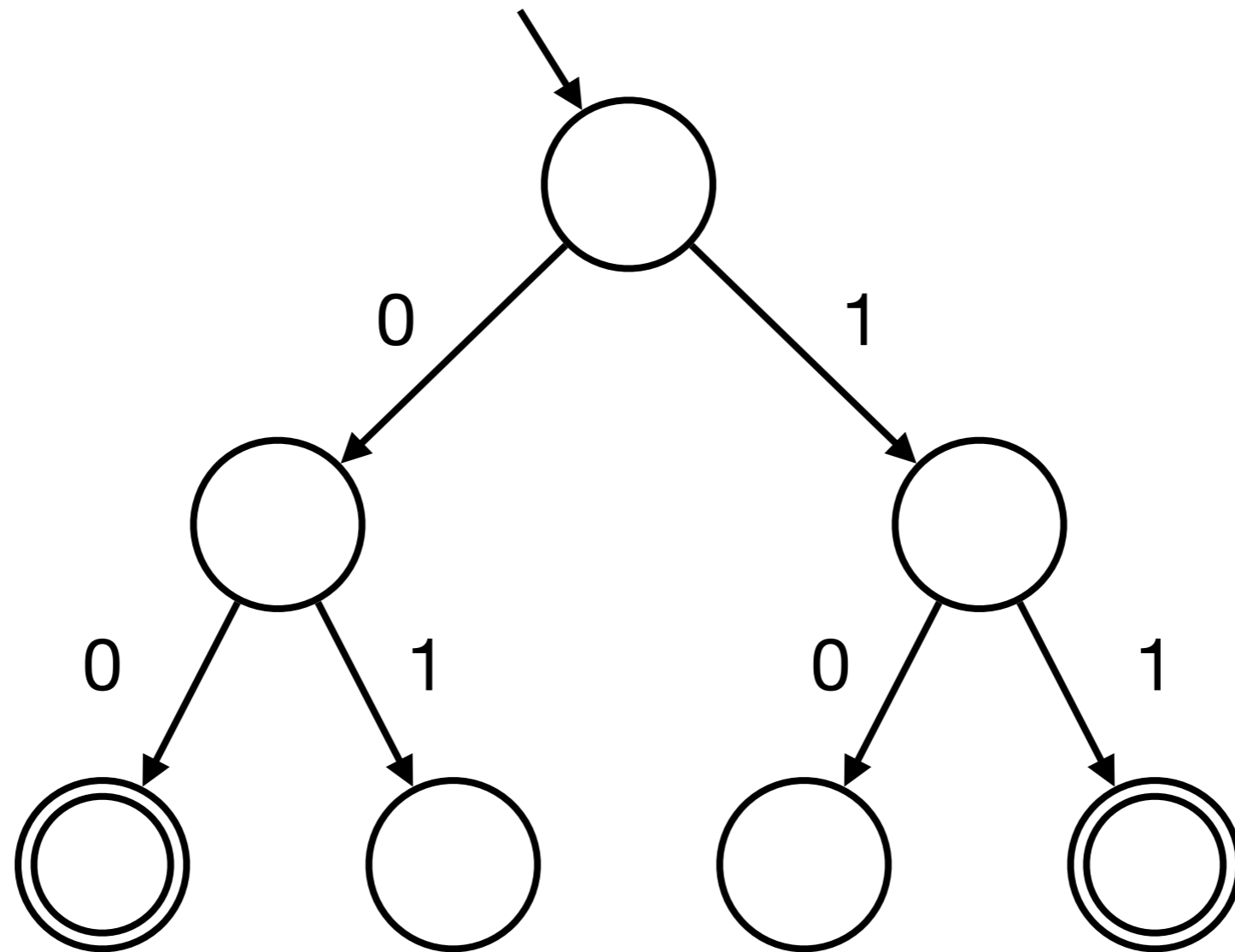
# Automates finis



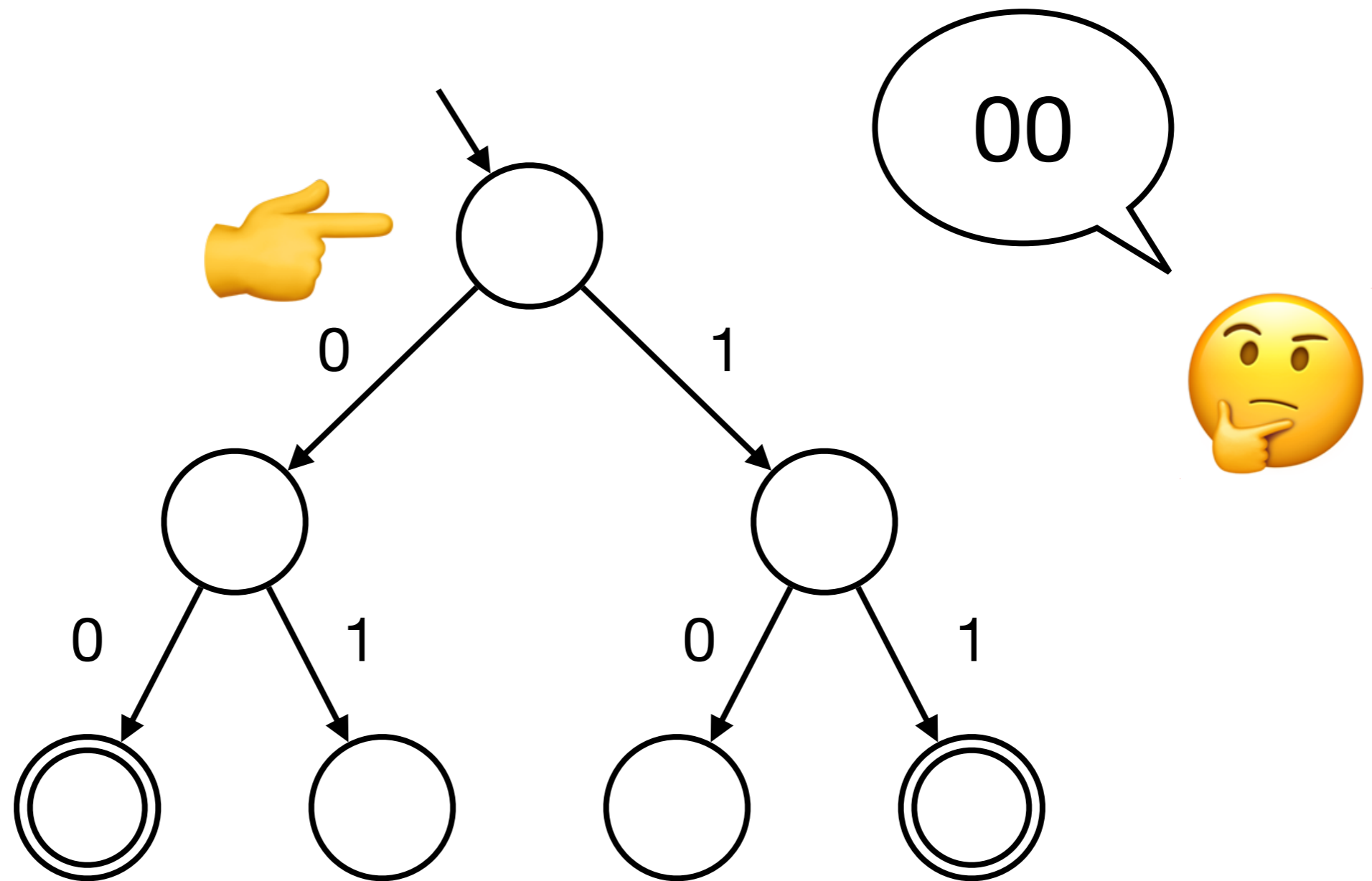
**Séquence de  
données**

**oui / non**

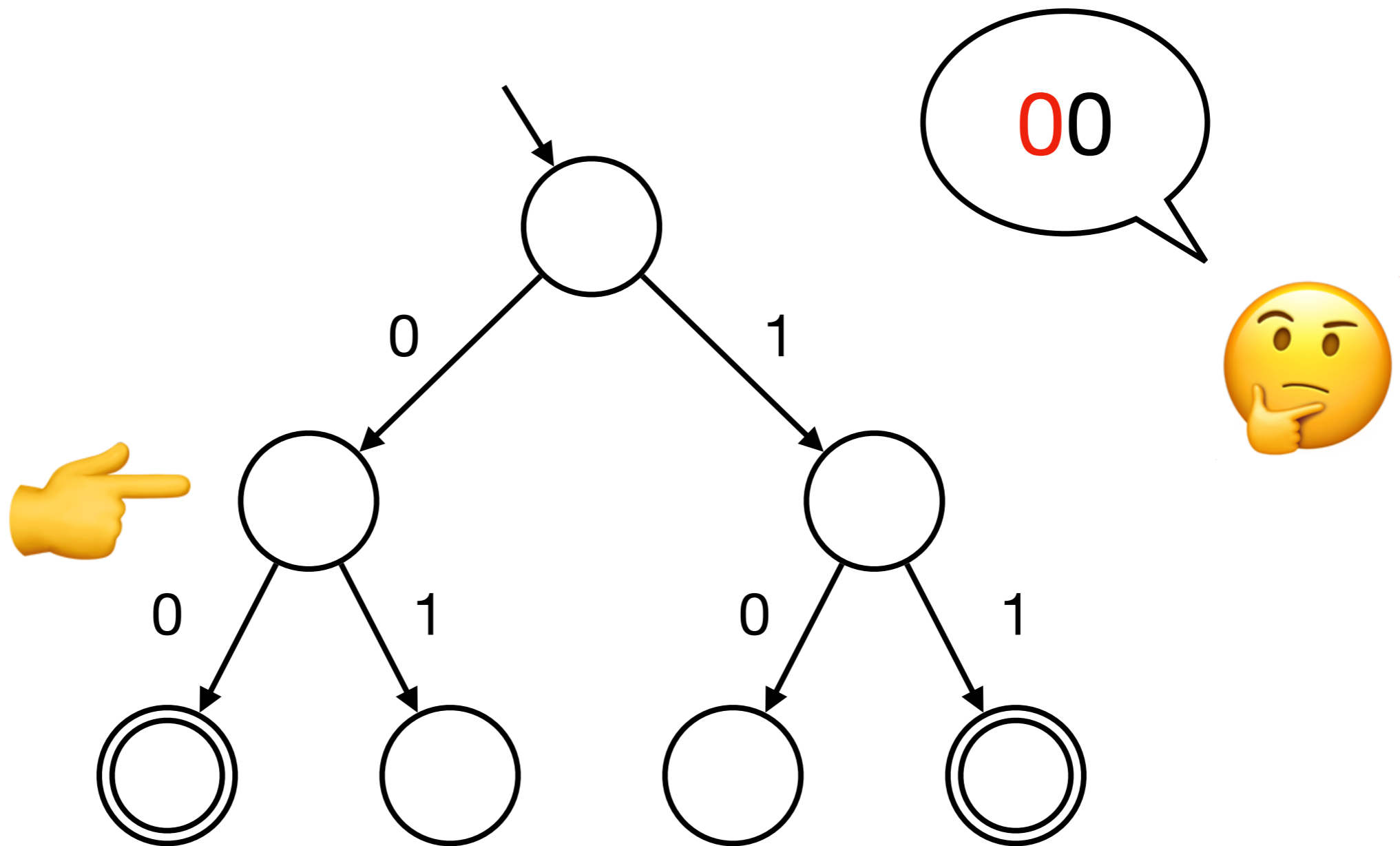
# Un automate fini



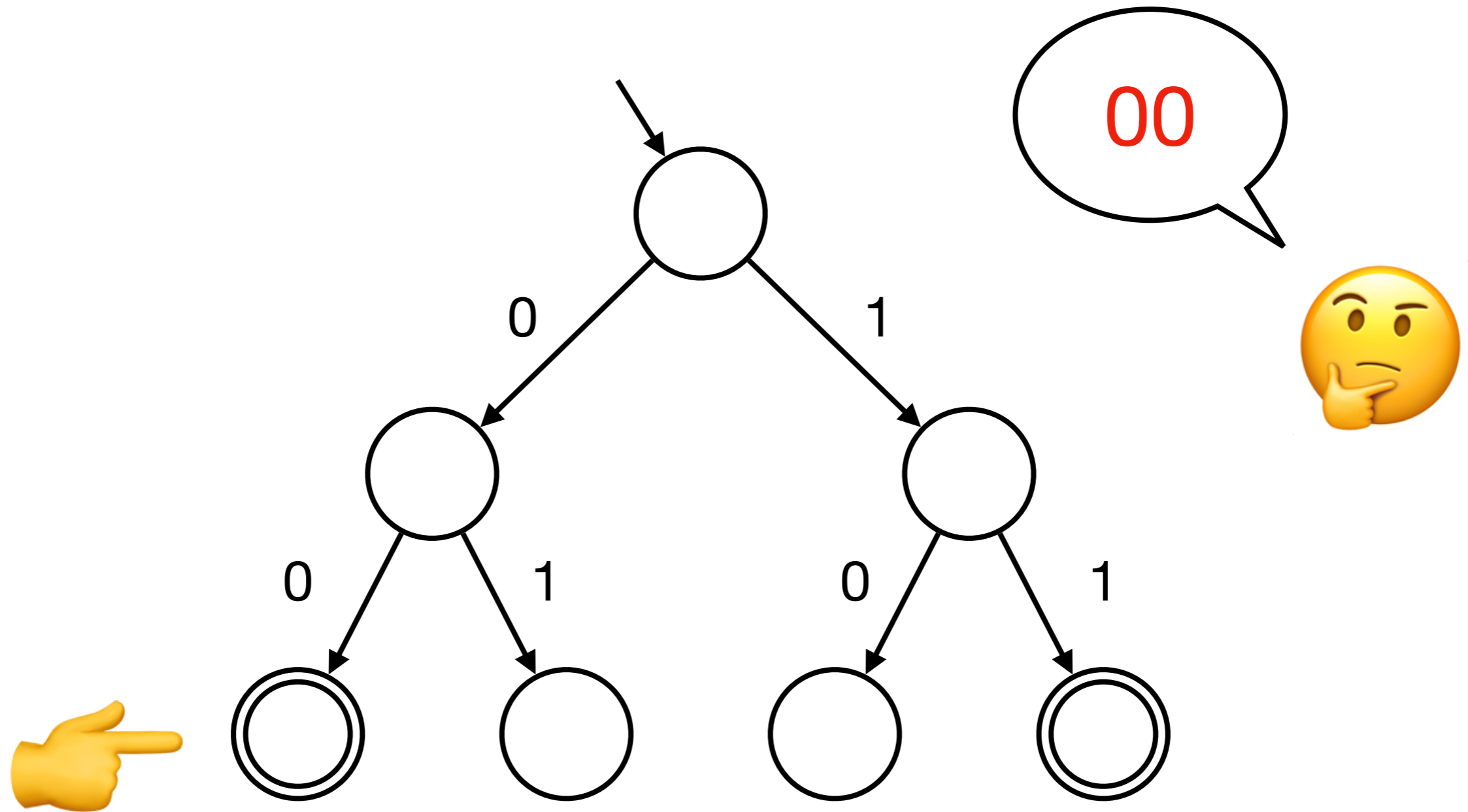
# Un automate fini



# Un automate fini

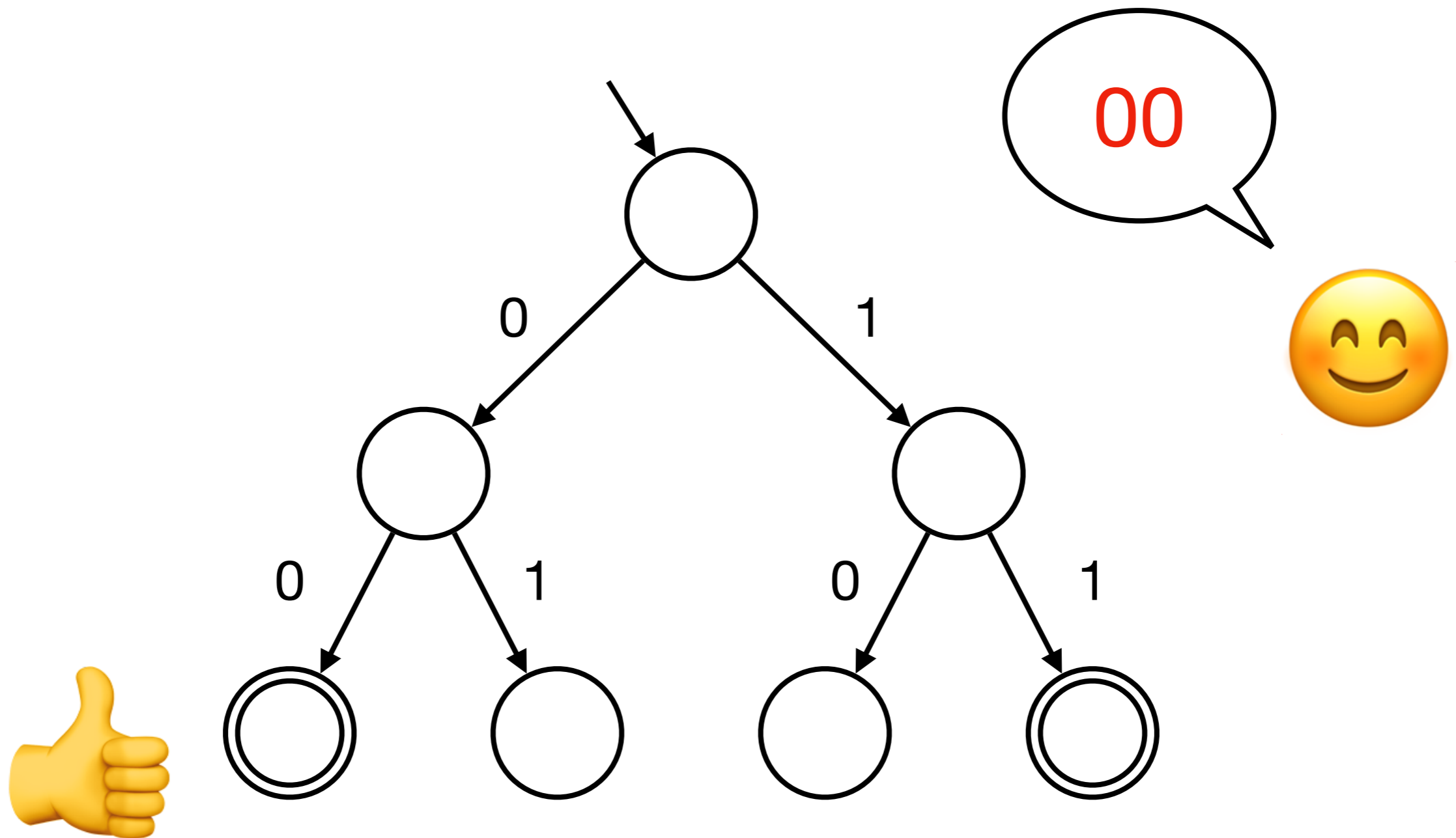


# Un automate fini



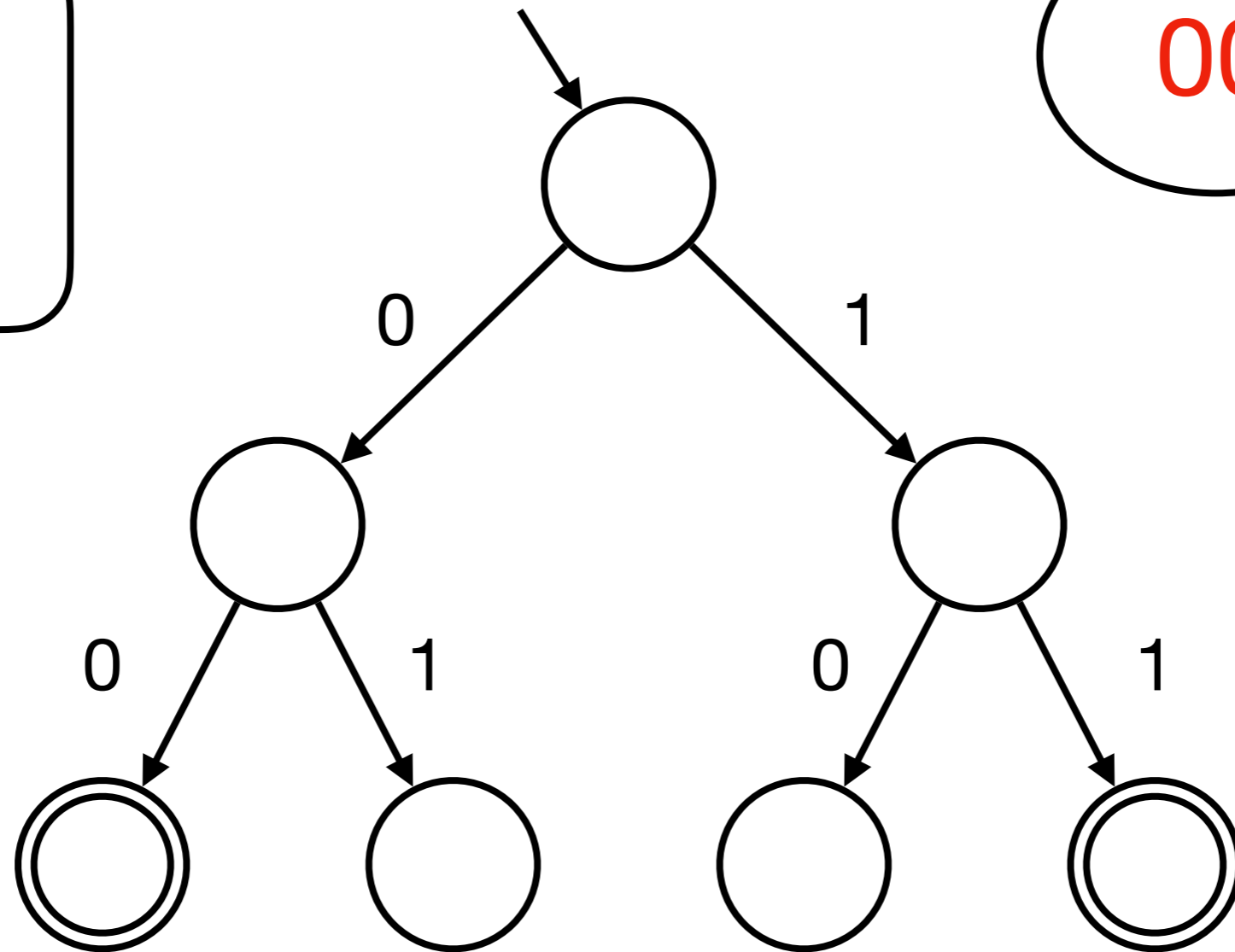
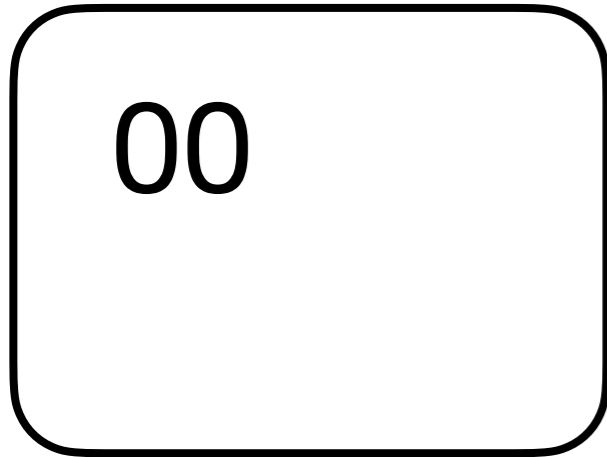


# Un automate fini



# Un automate fini

mots acceptés

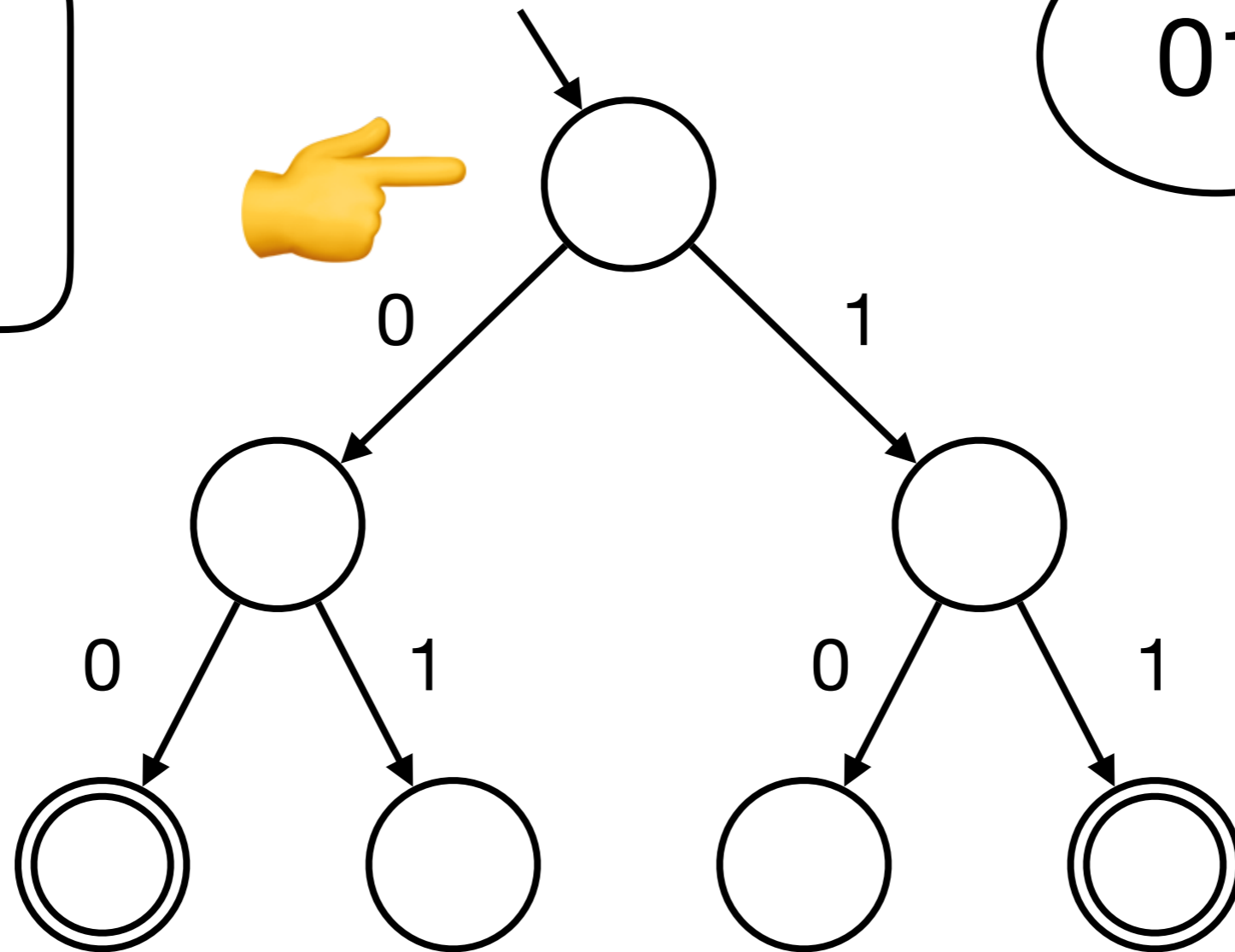


# Un automate fini

mots acceptés

00

01

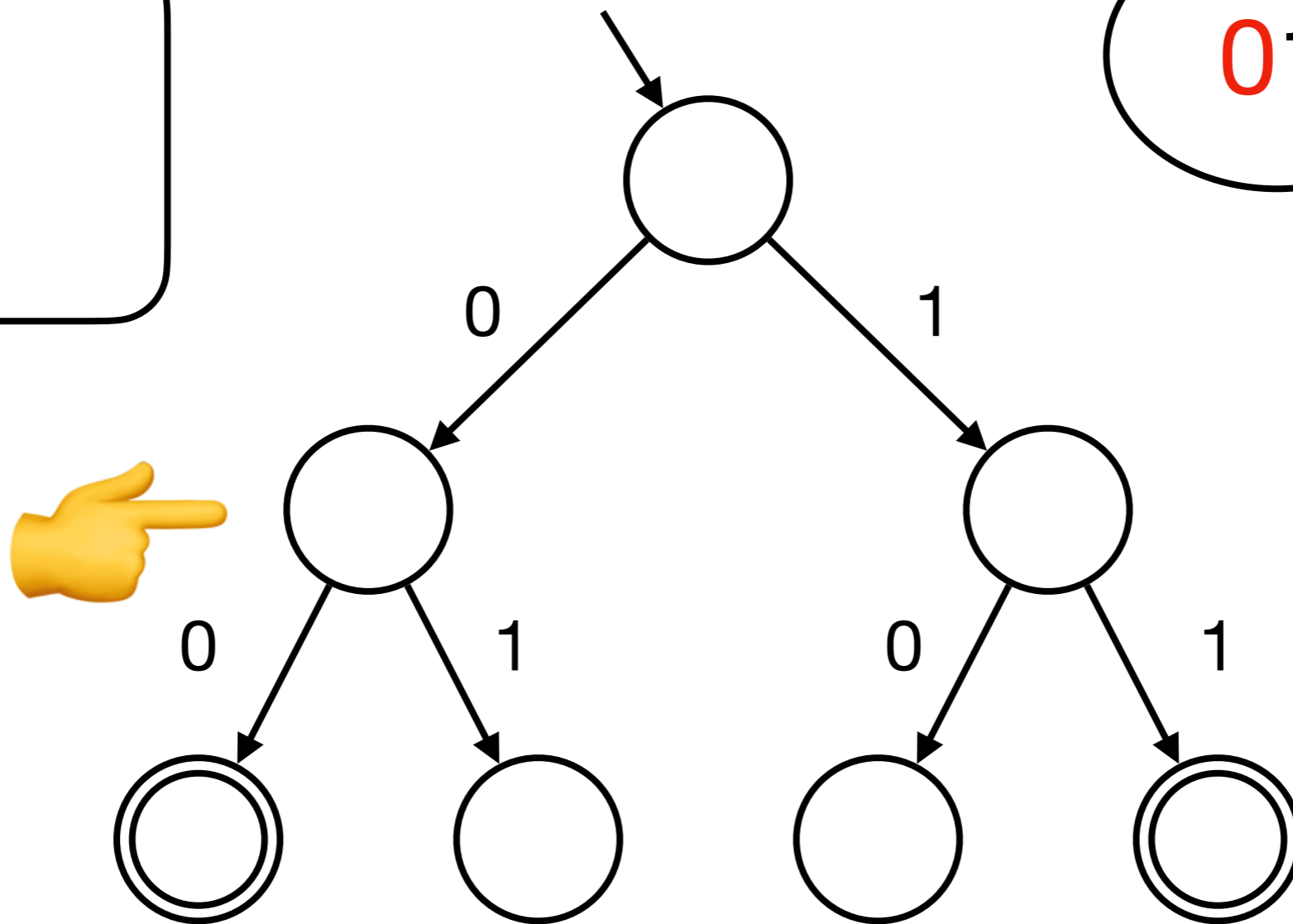


# Un automate fini

mots acceptés

00

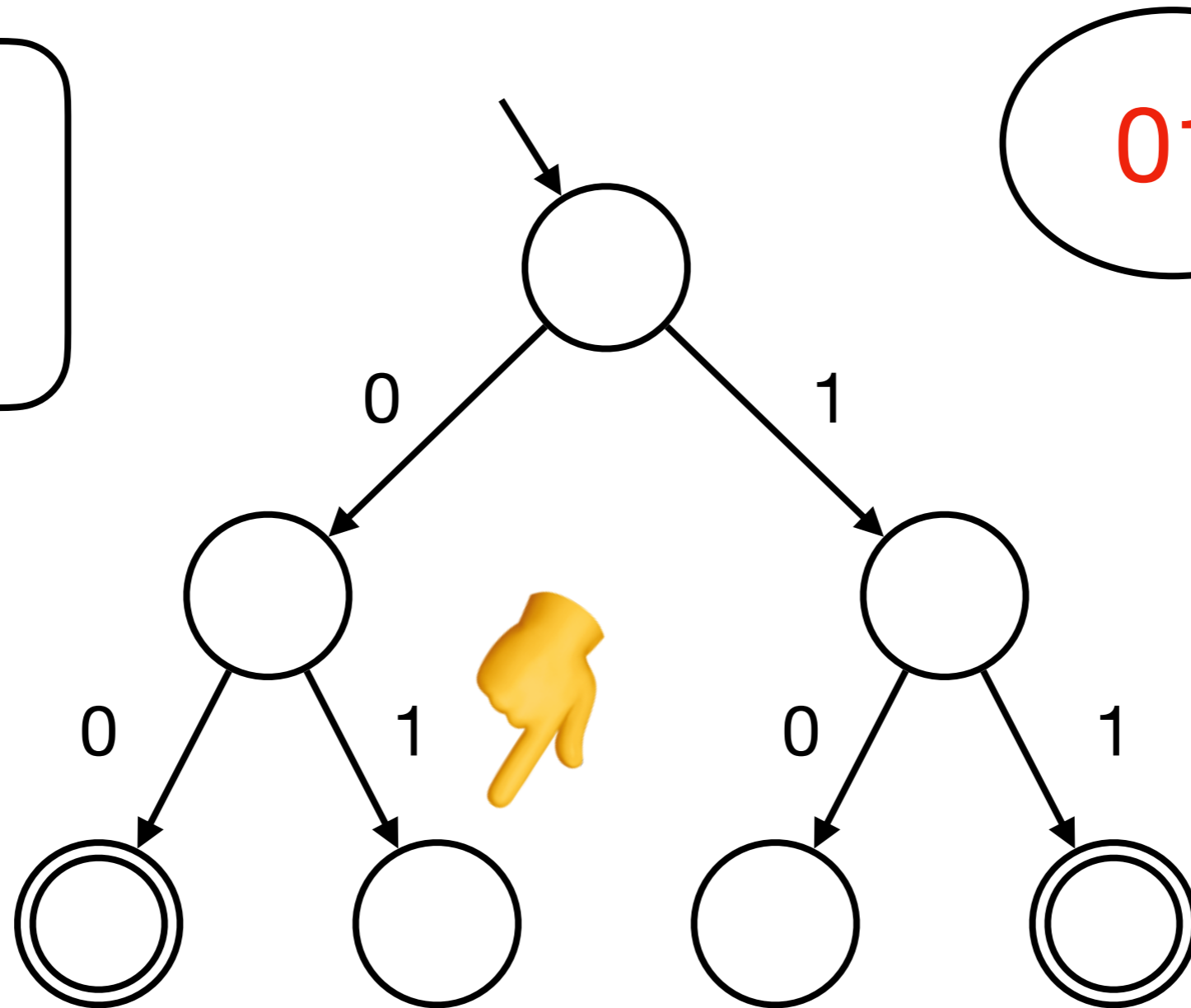
01



# Un automate fini

mots acceptés

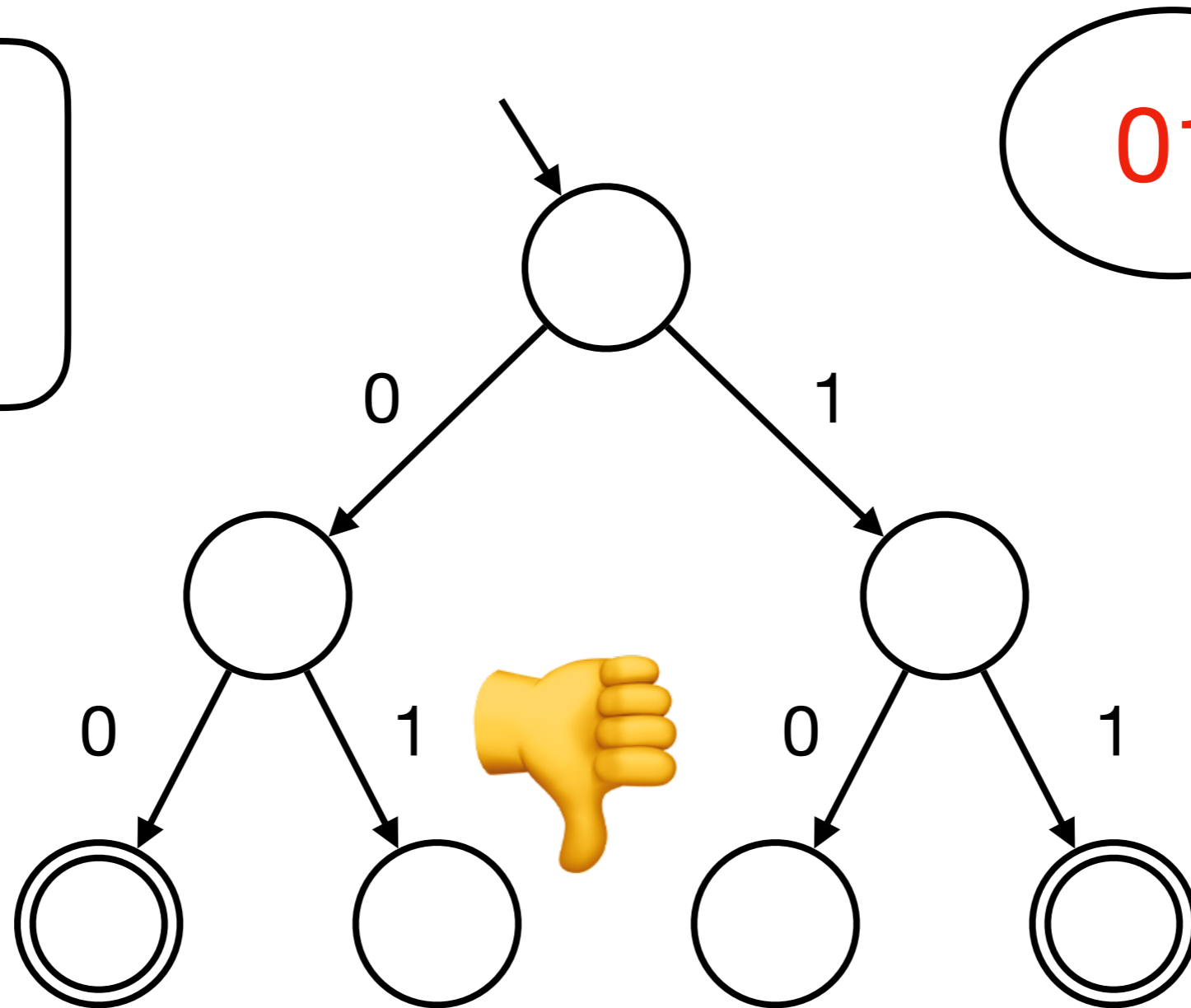
00



# Un automate fini

mots acceptés

00



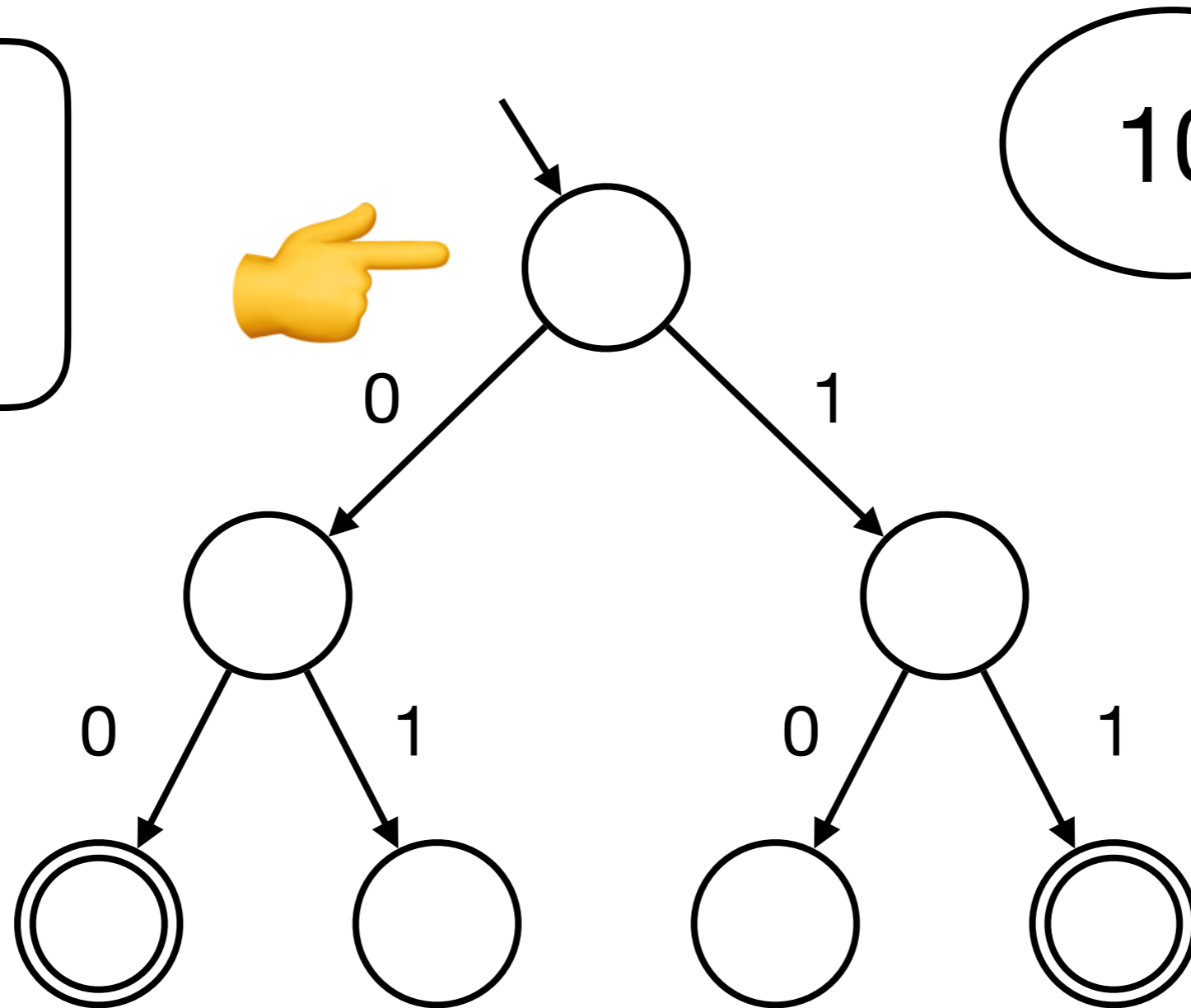
01



# Un automate fini

mots acceptés

00



10

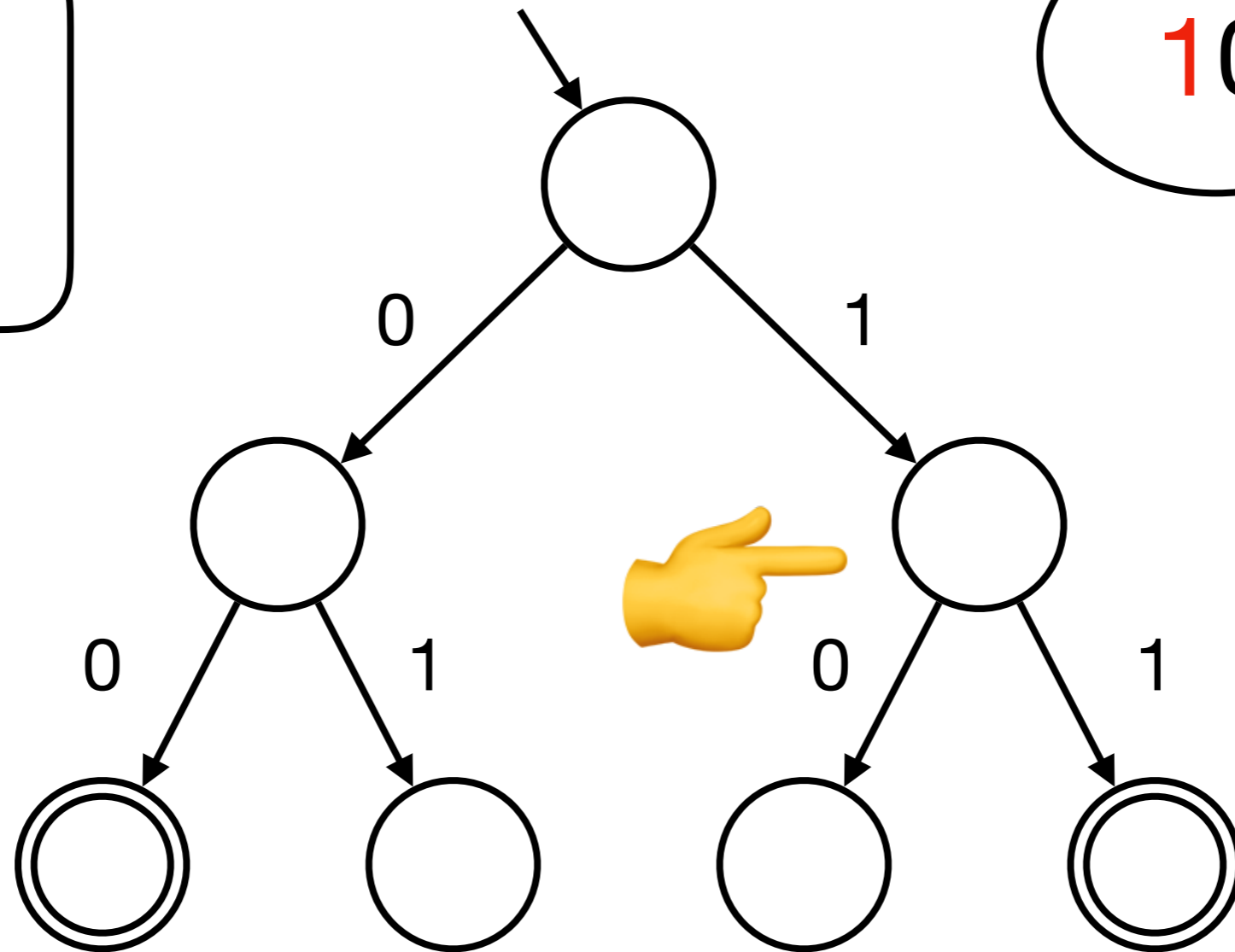


# Un automate fini

mots acceptés

00

10

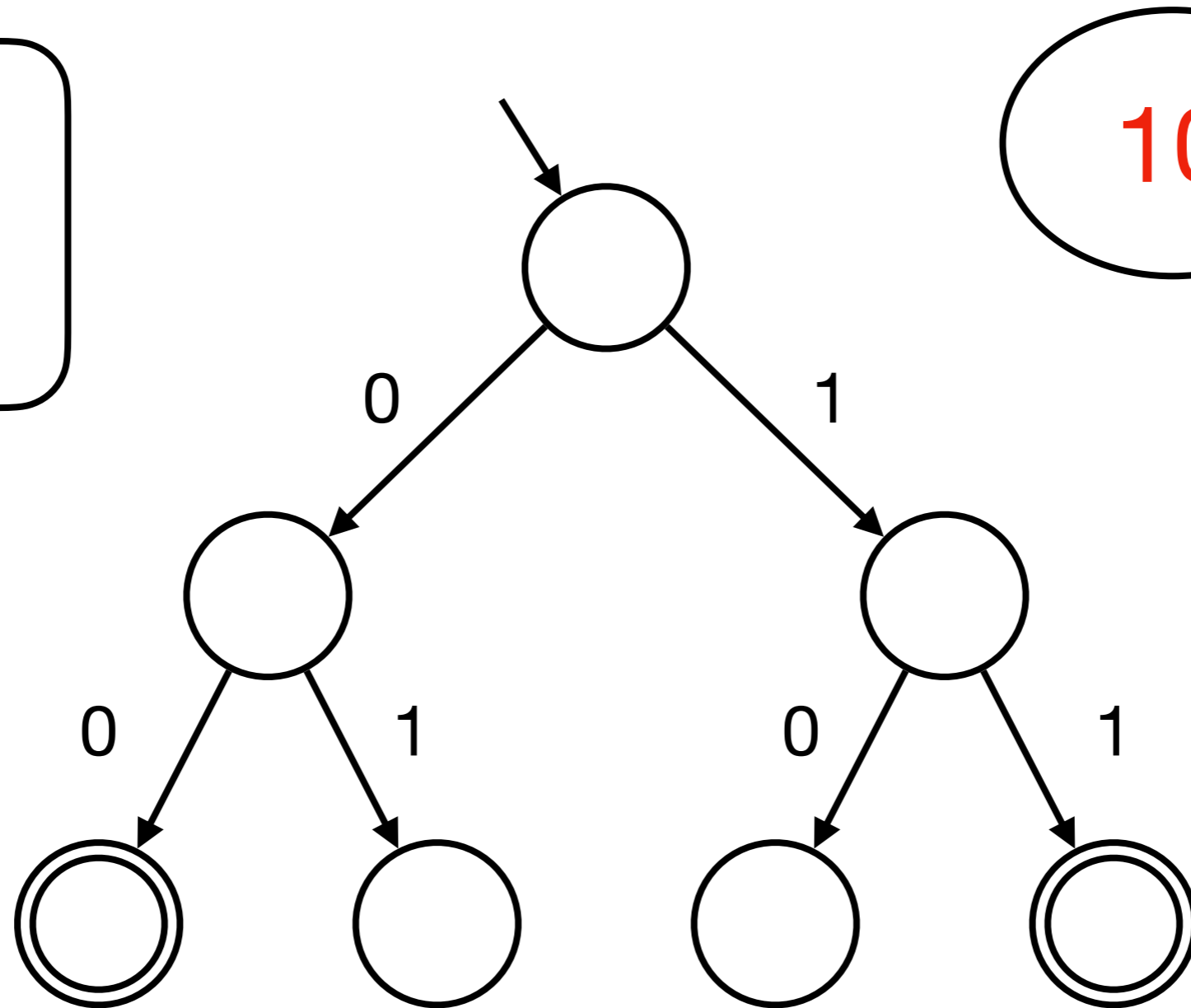




# Un automate fini

mots acceptés

00



10

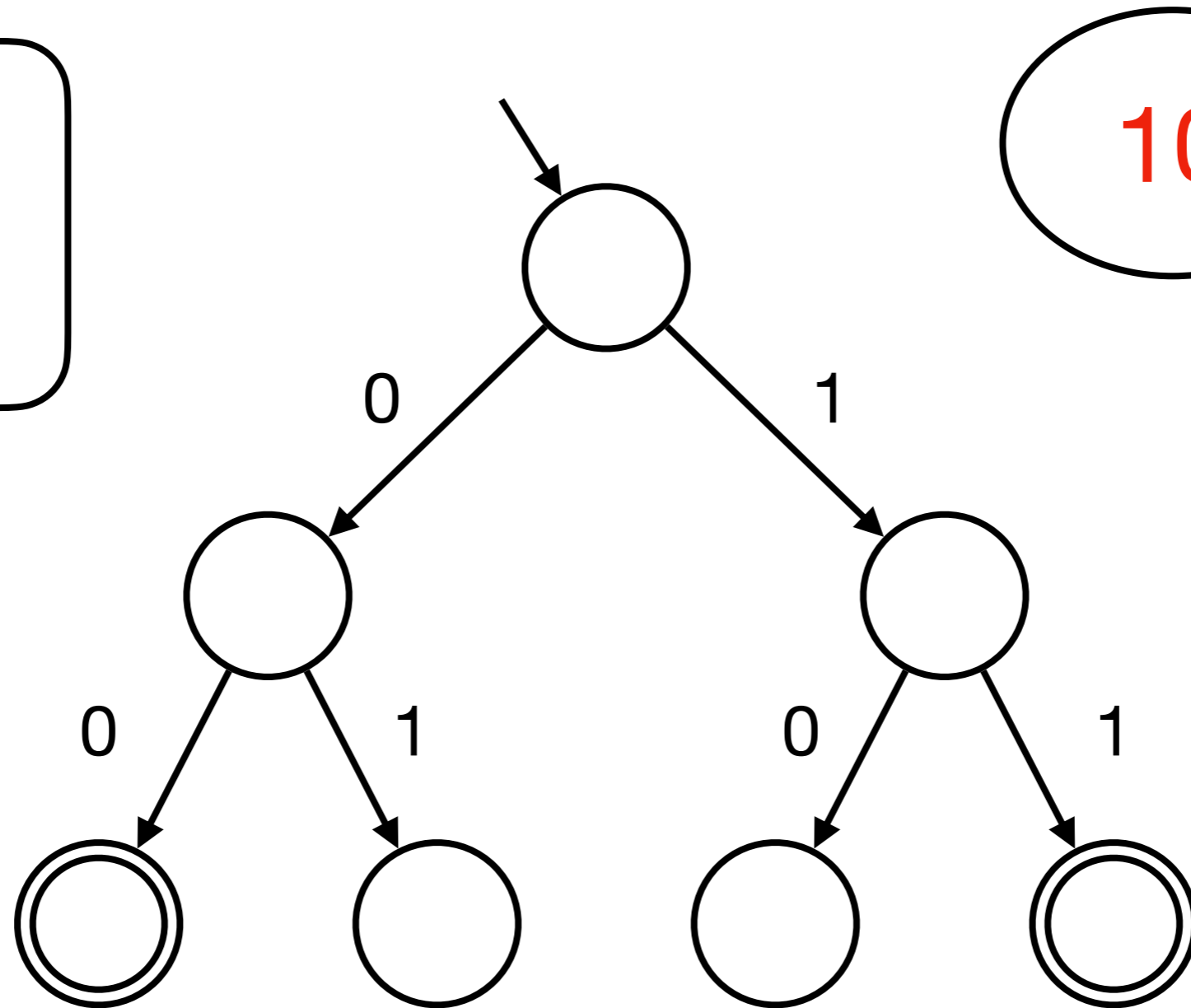


# Un automate fini

mots acceptés

00

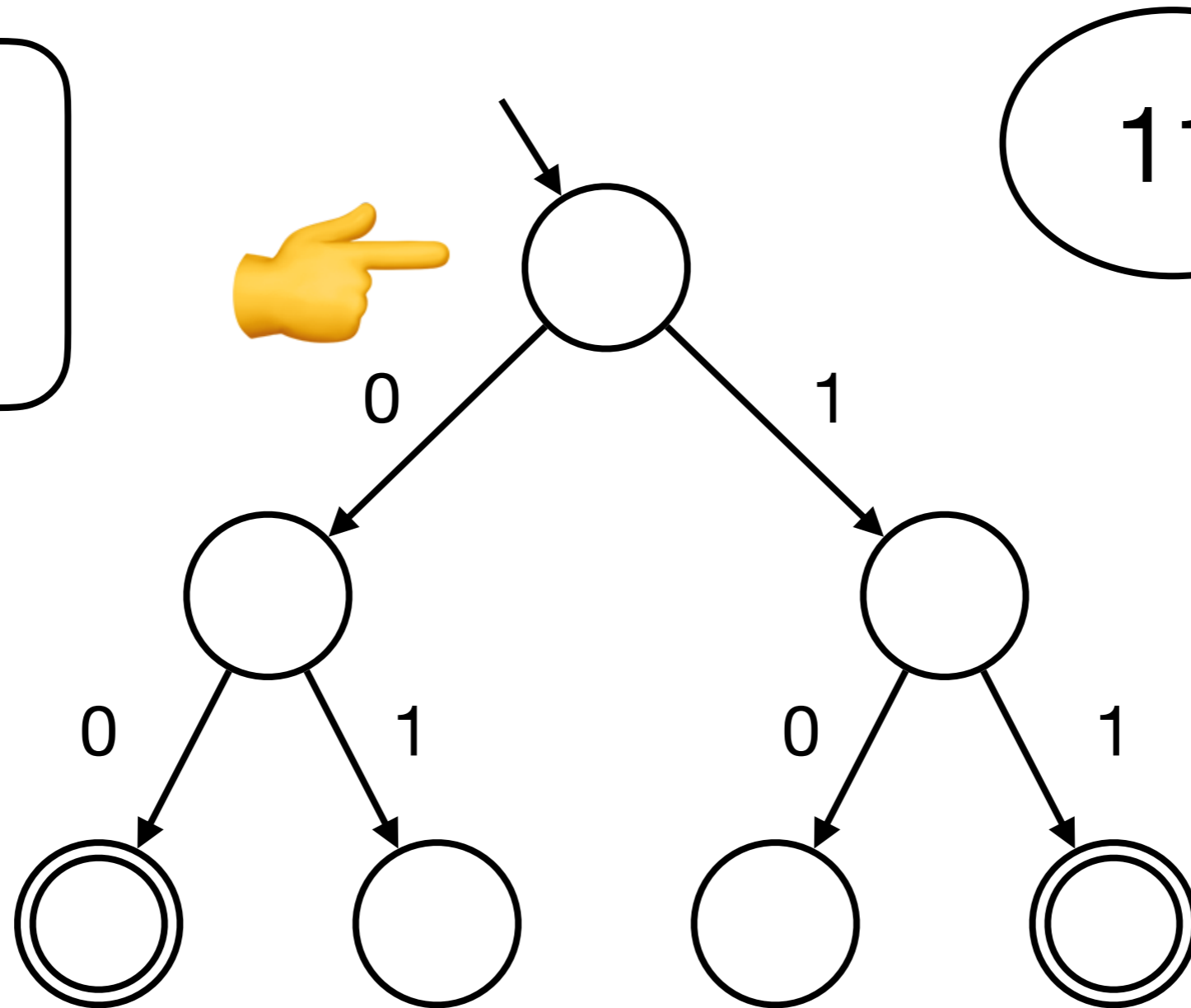
10



# Un automate fini

mots acceptés

00



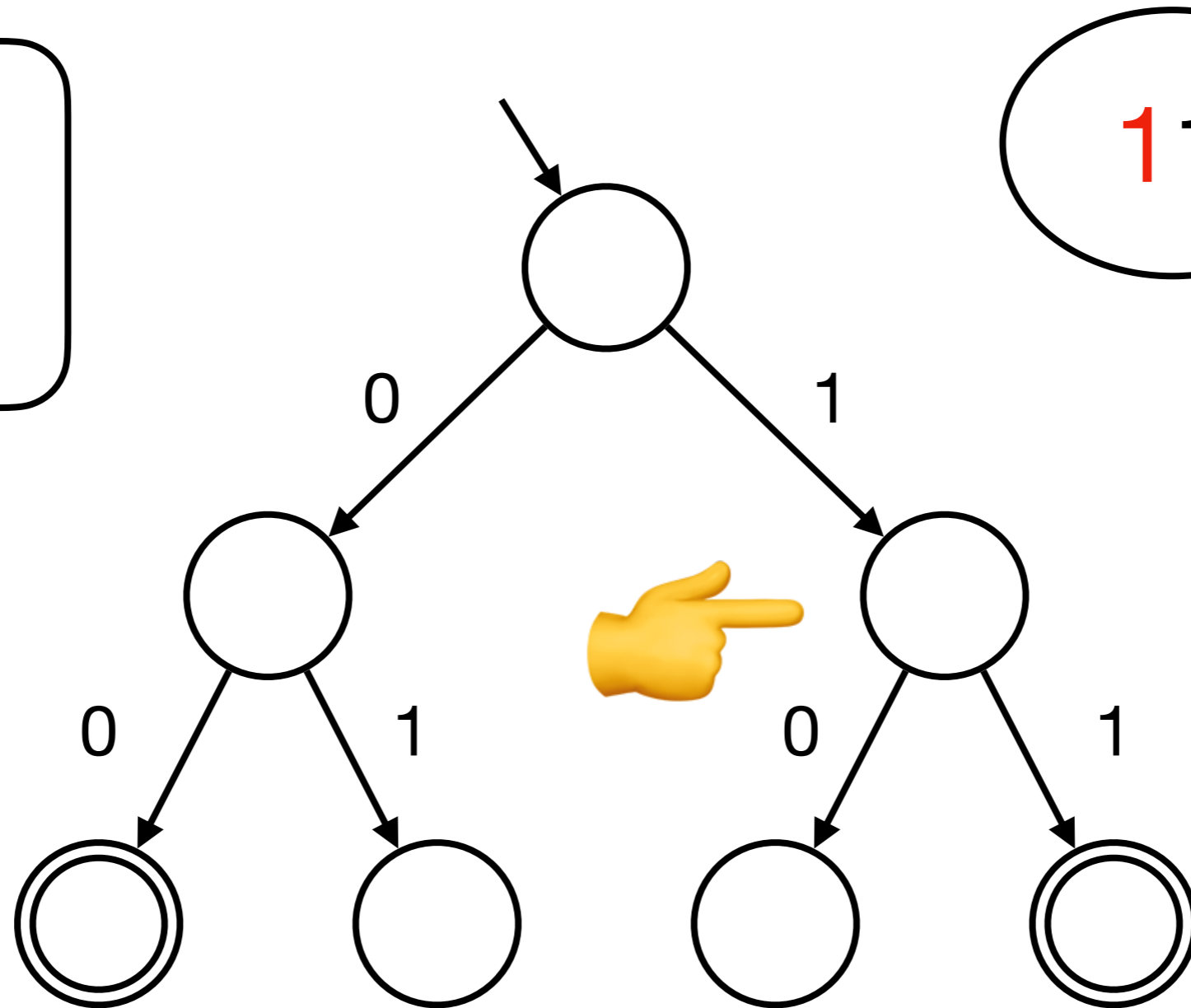
11



# Un automate fini

mots acceptés

00



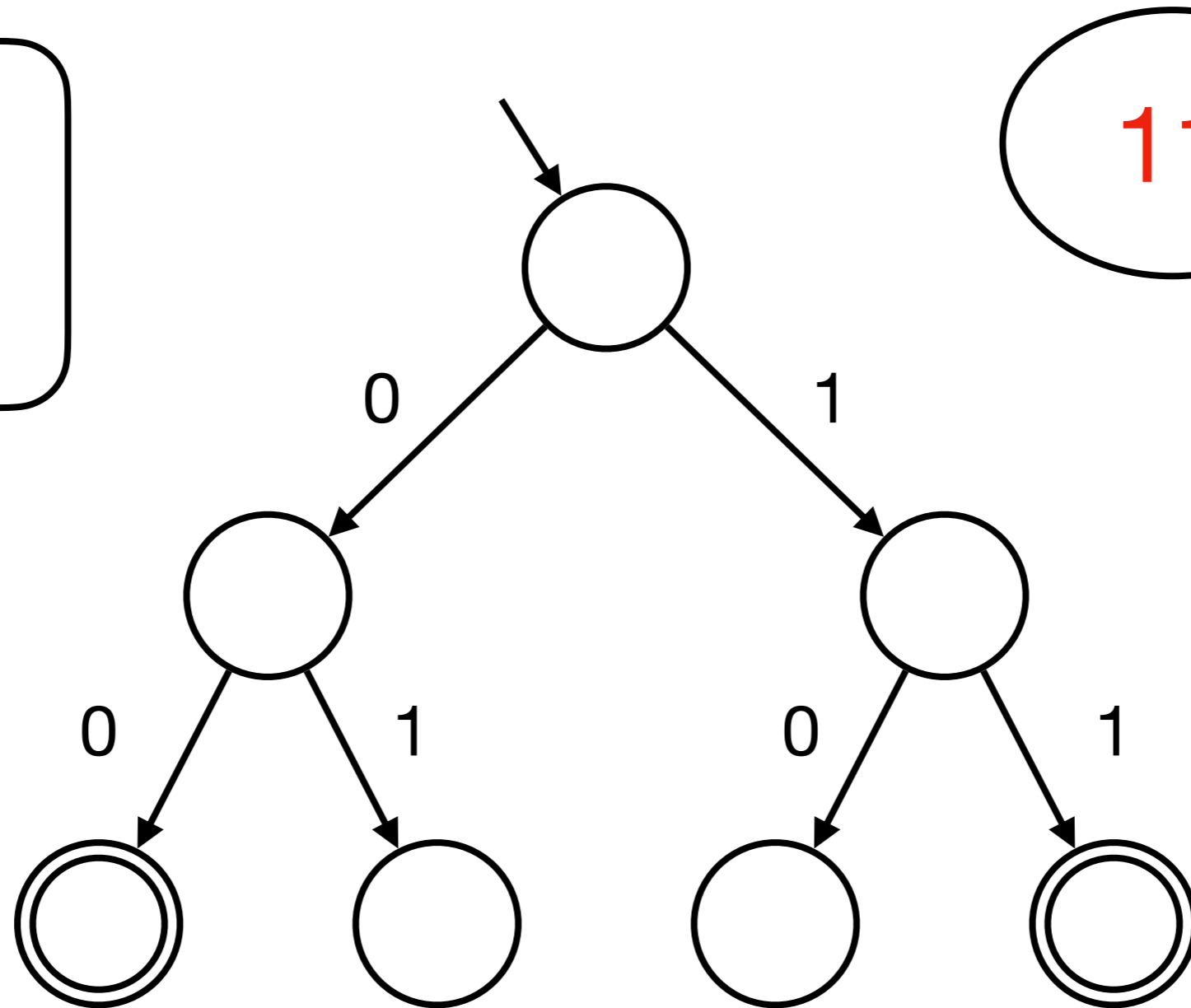
11



# Un automate fini

mots acceptés

00



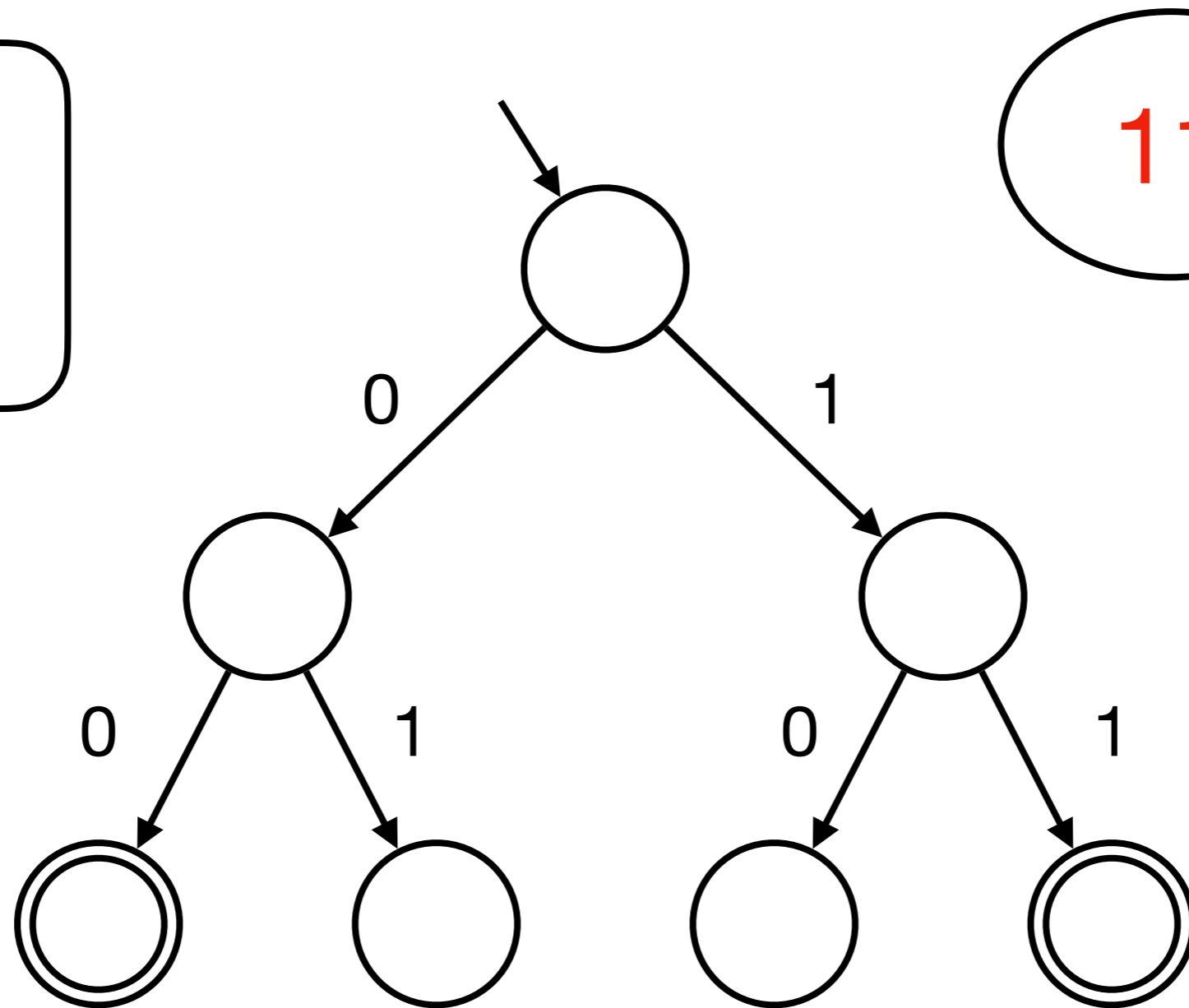
11



# Un automate fini

mots acceptés

00

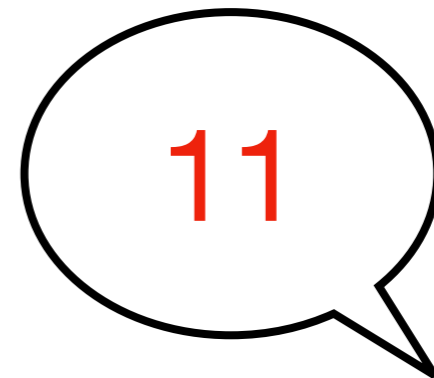
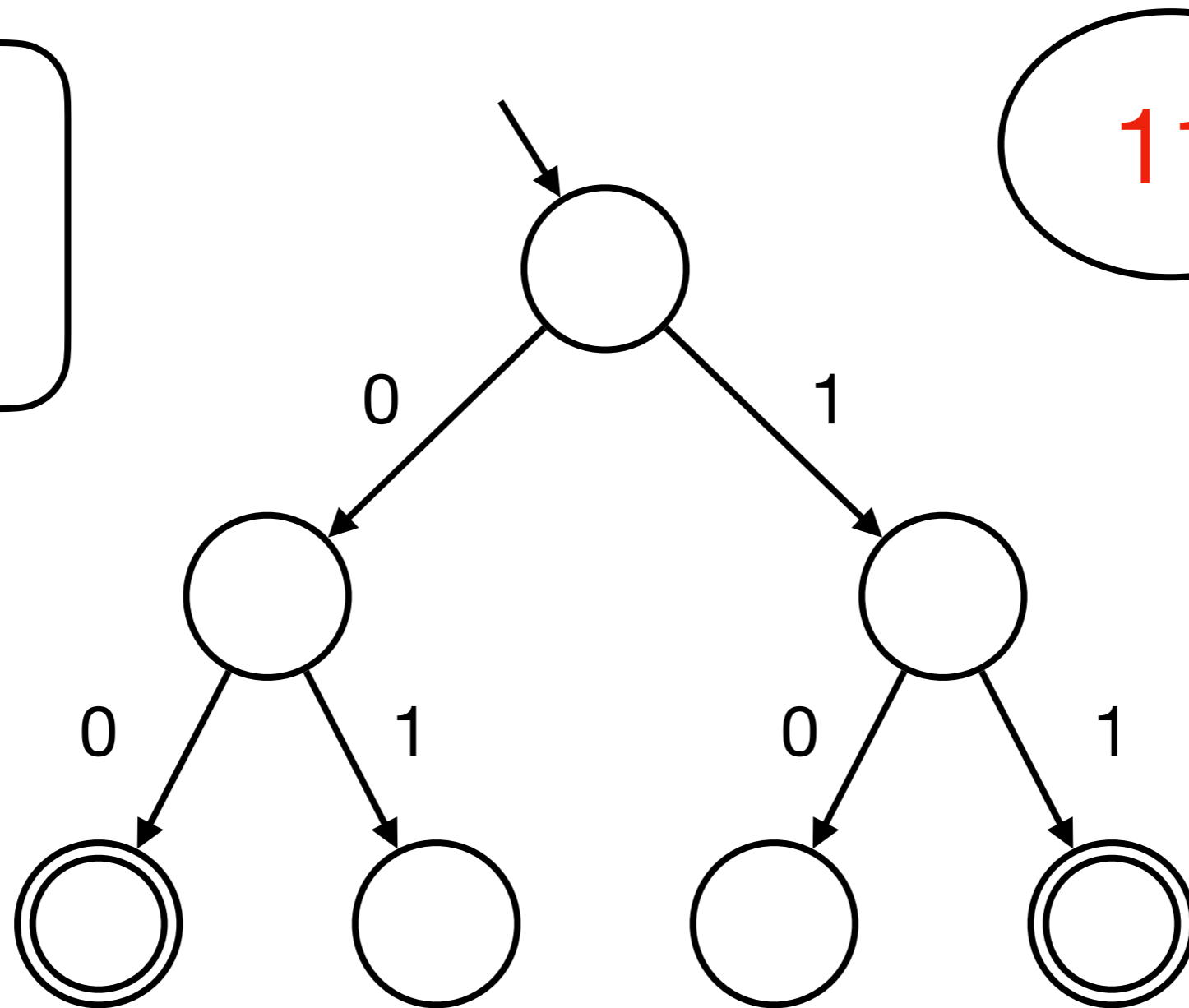
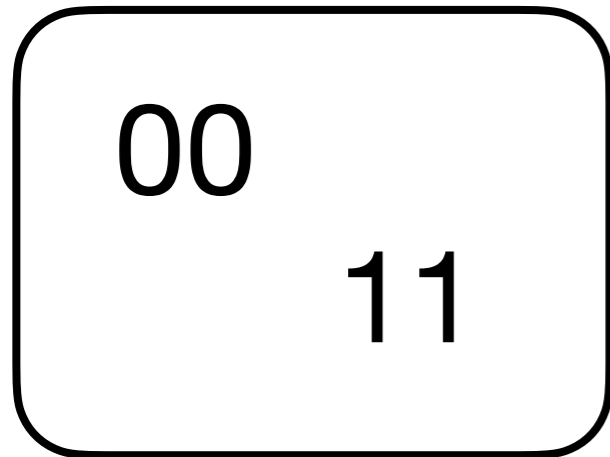


11



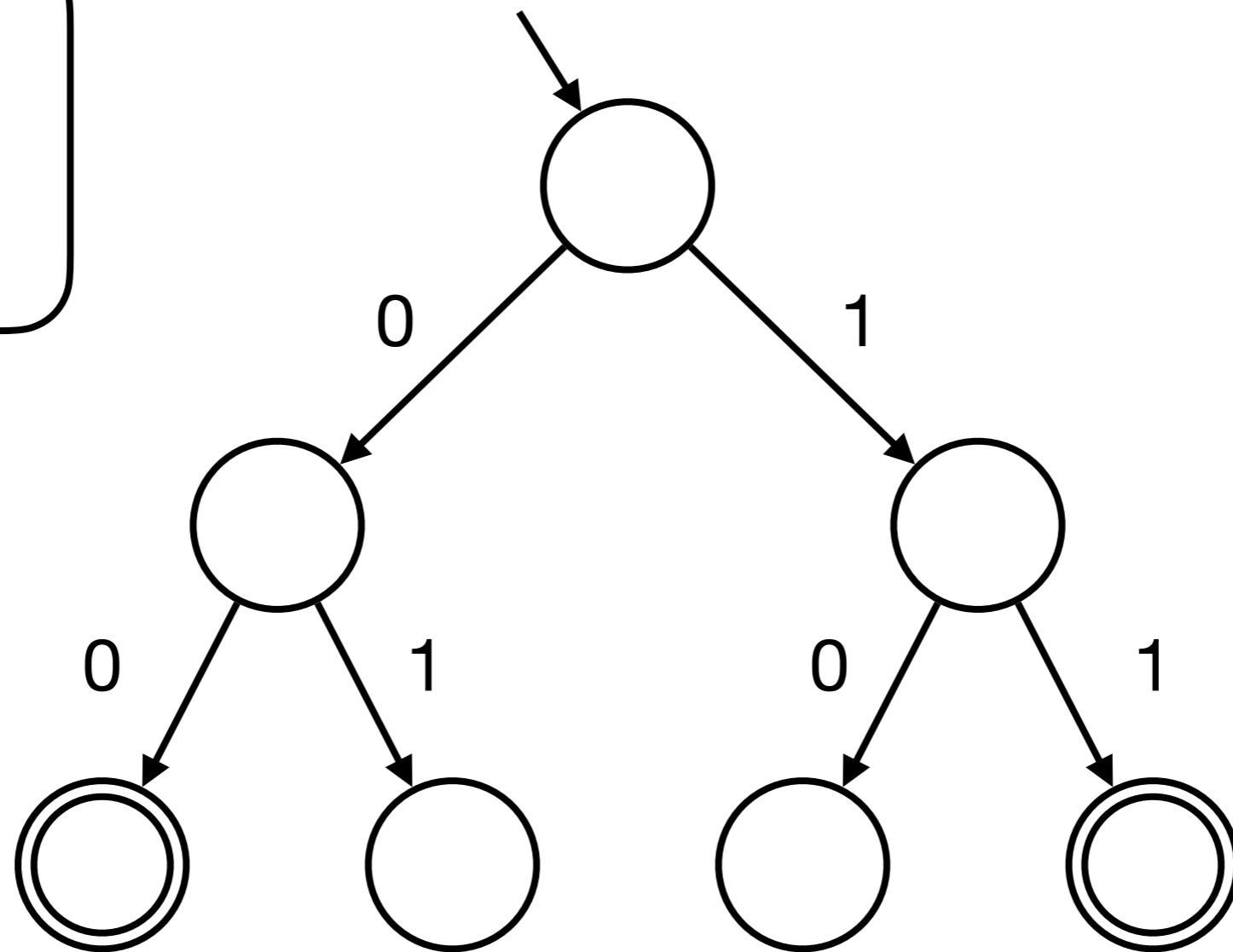
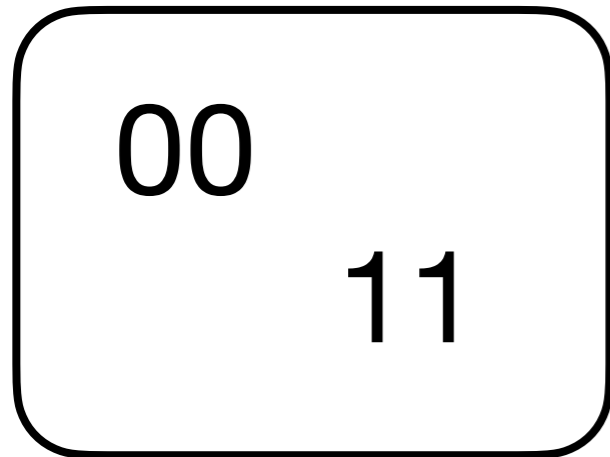
# Un automate fini

mots acceptés



# Un automate fini

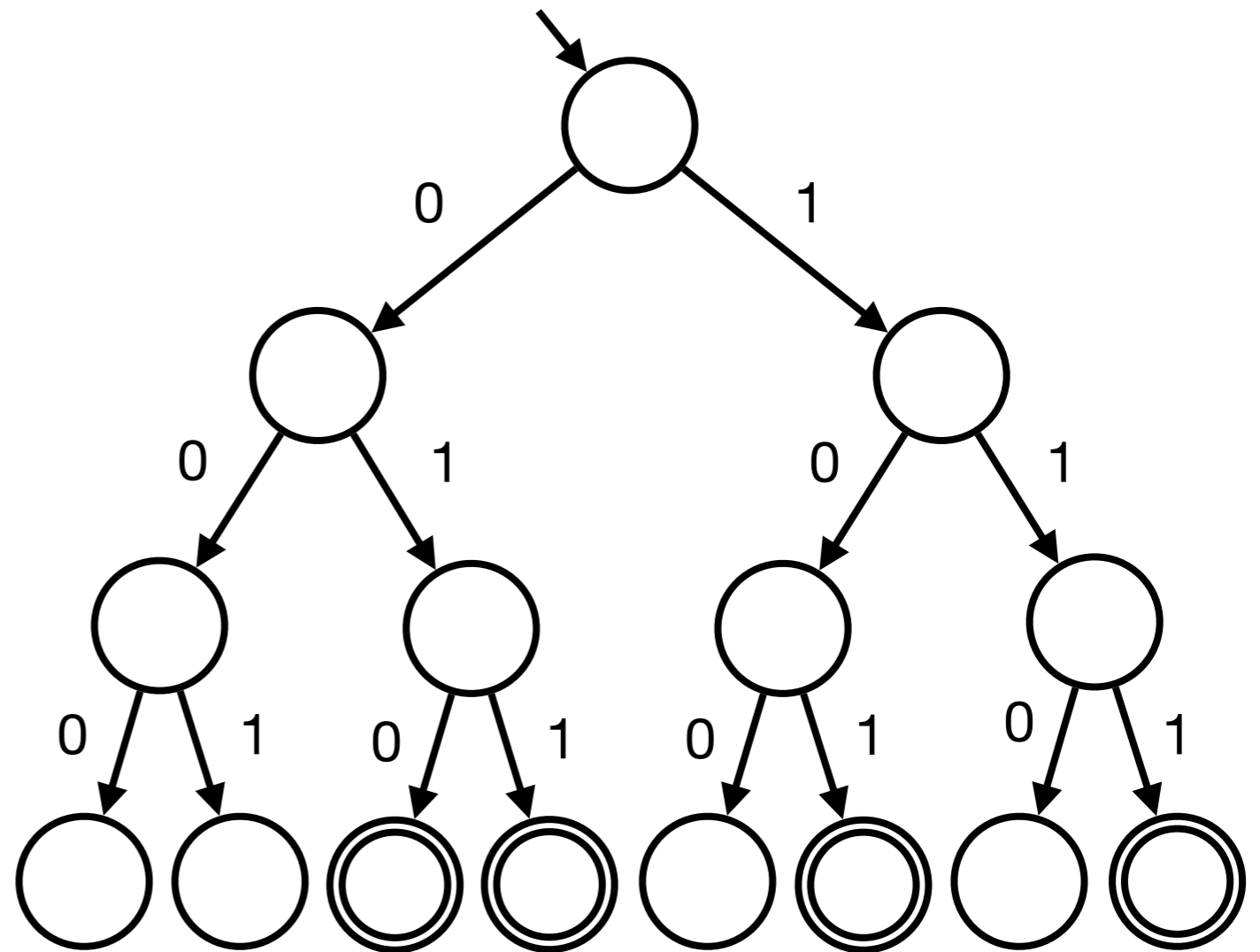
mots acceptés





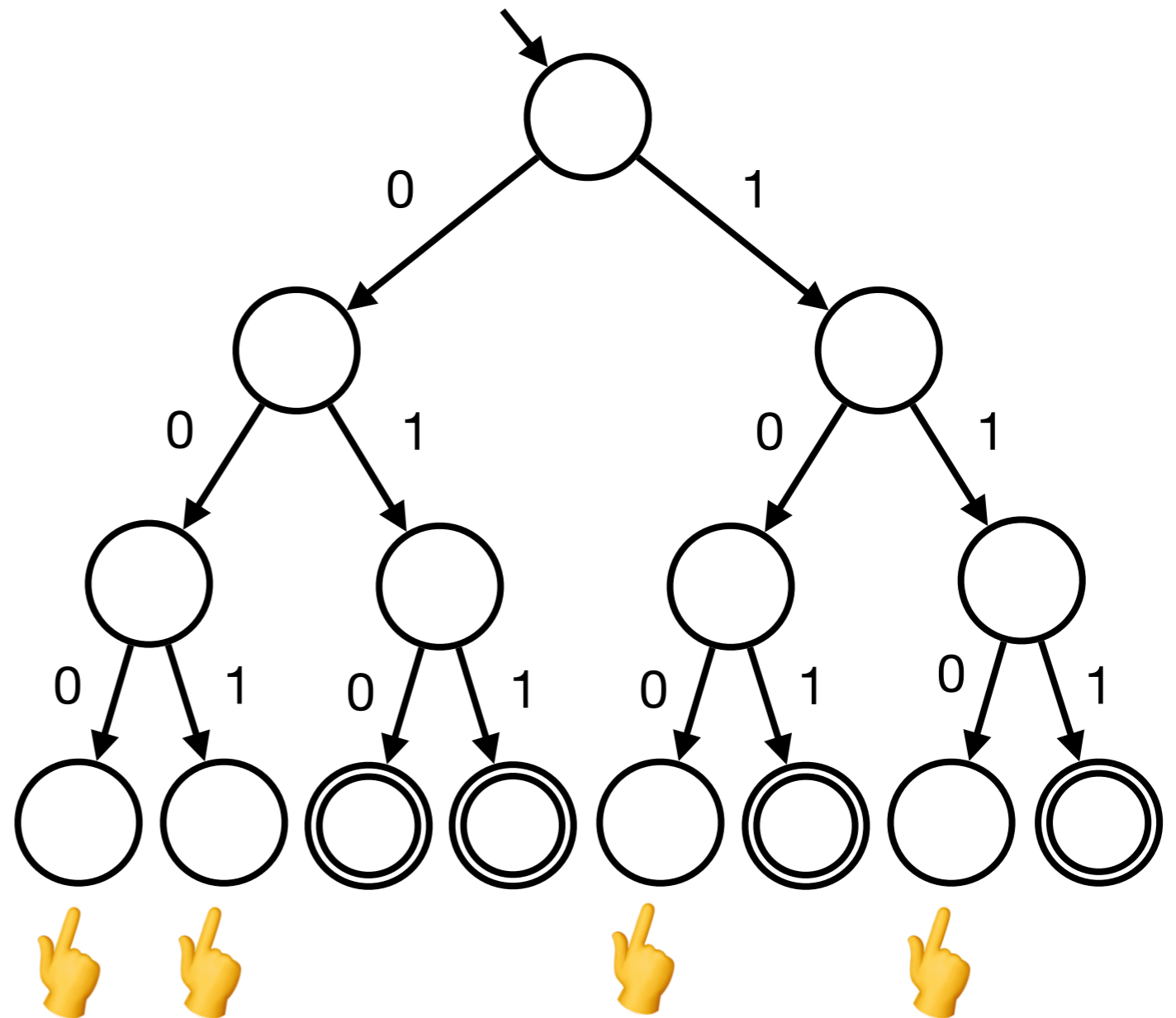
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



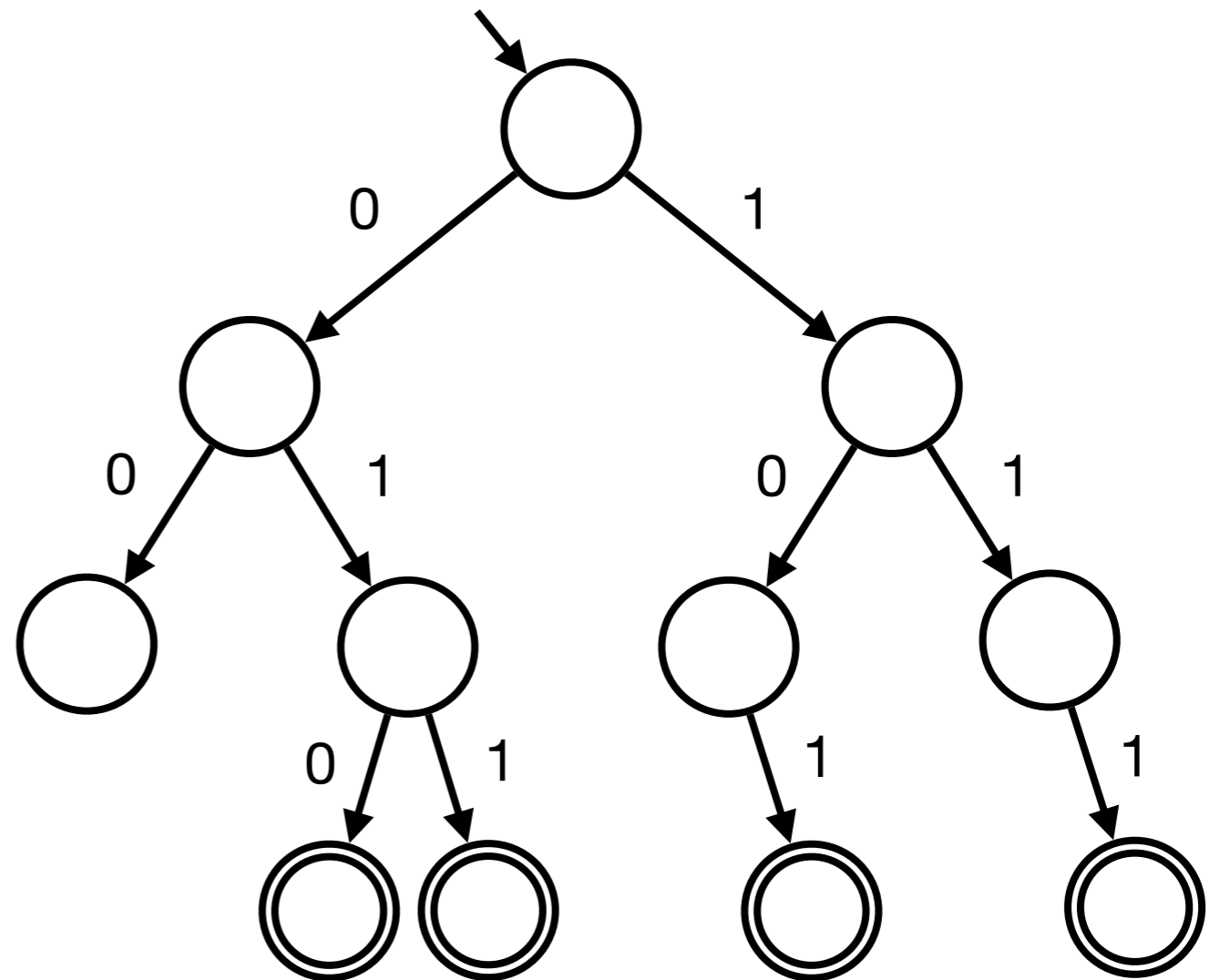
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



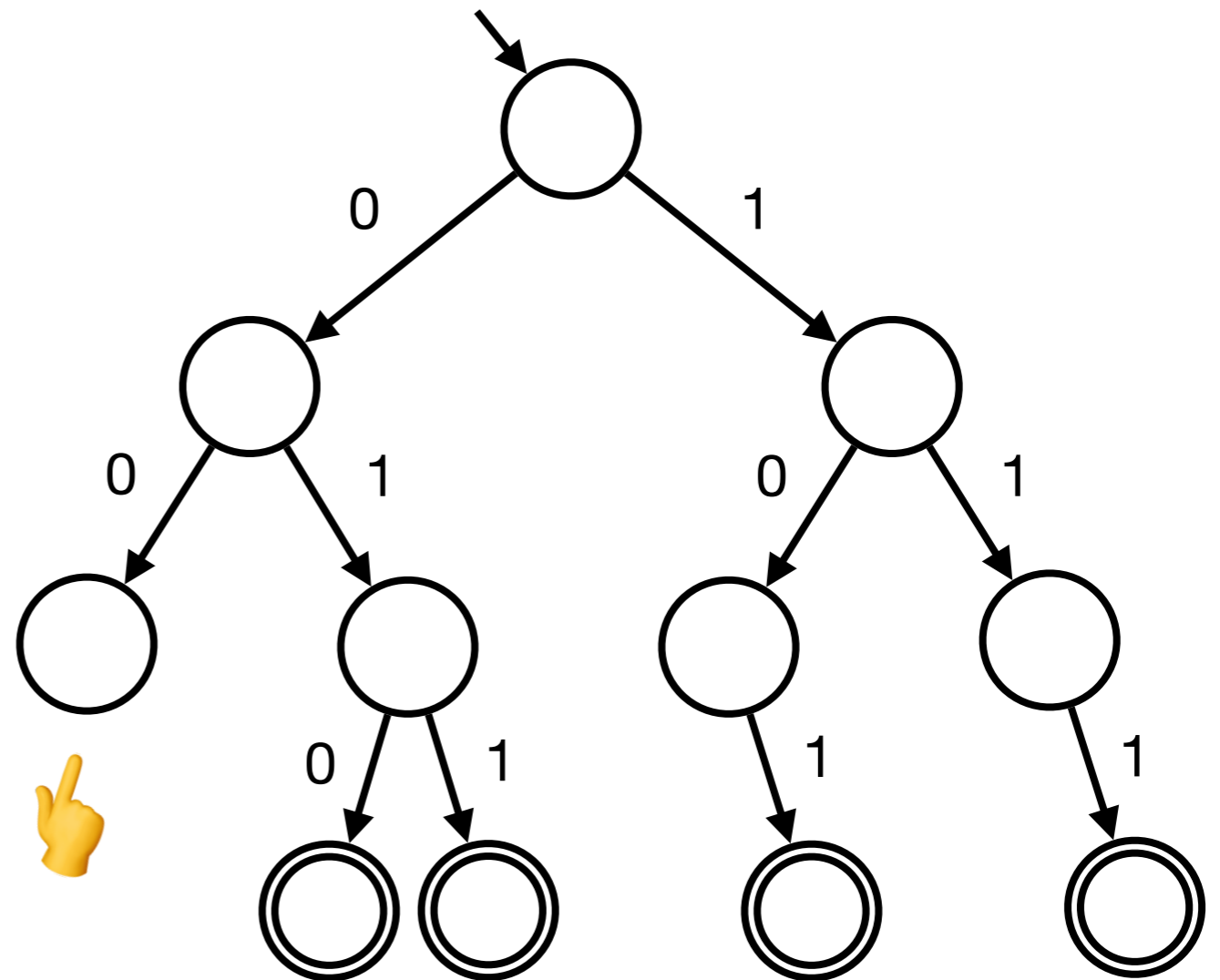
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



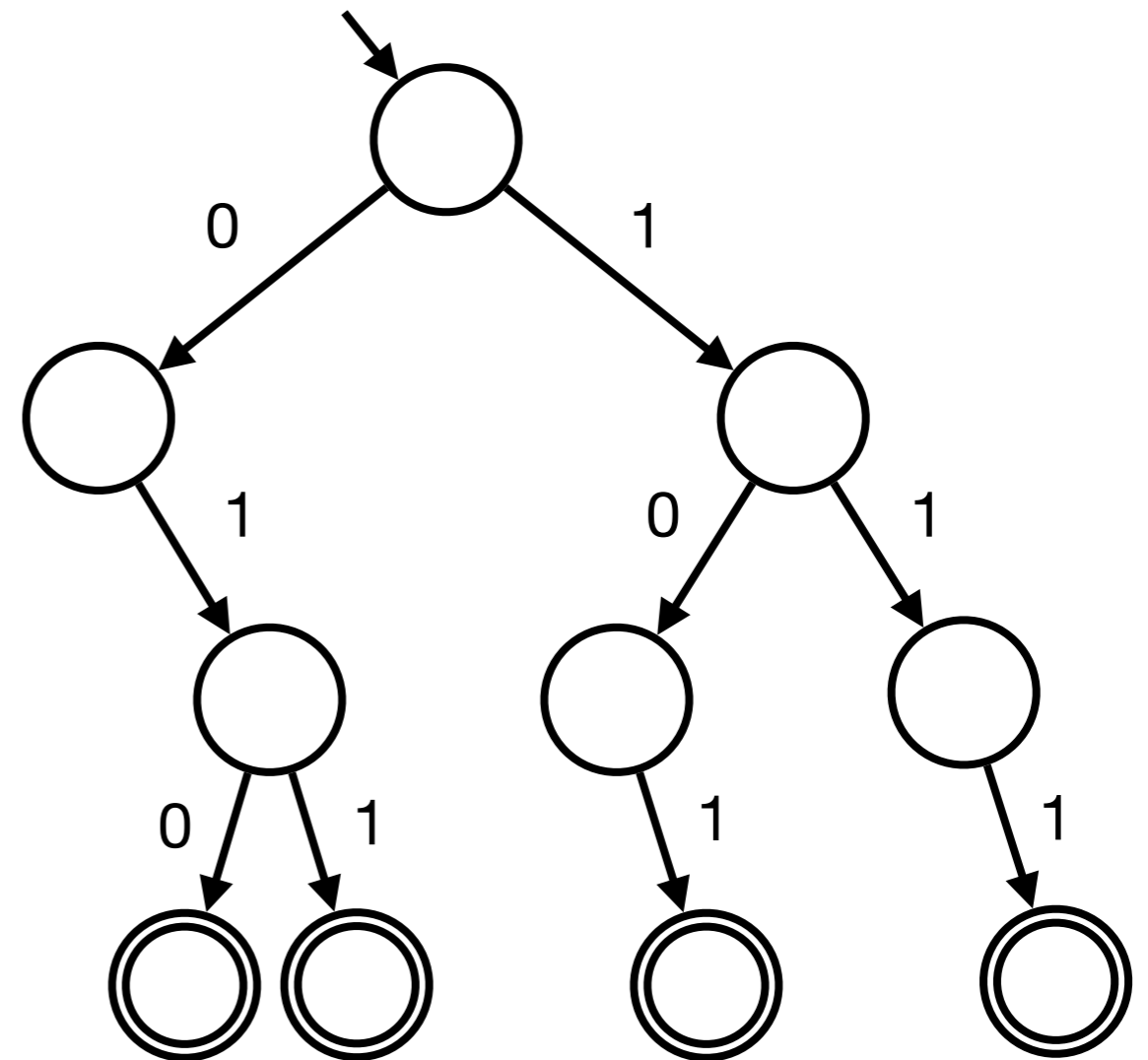
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



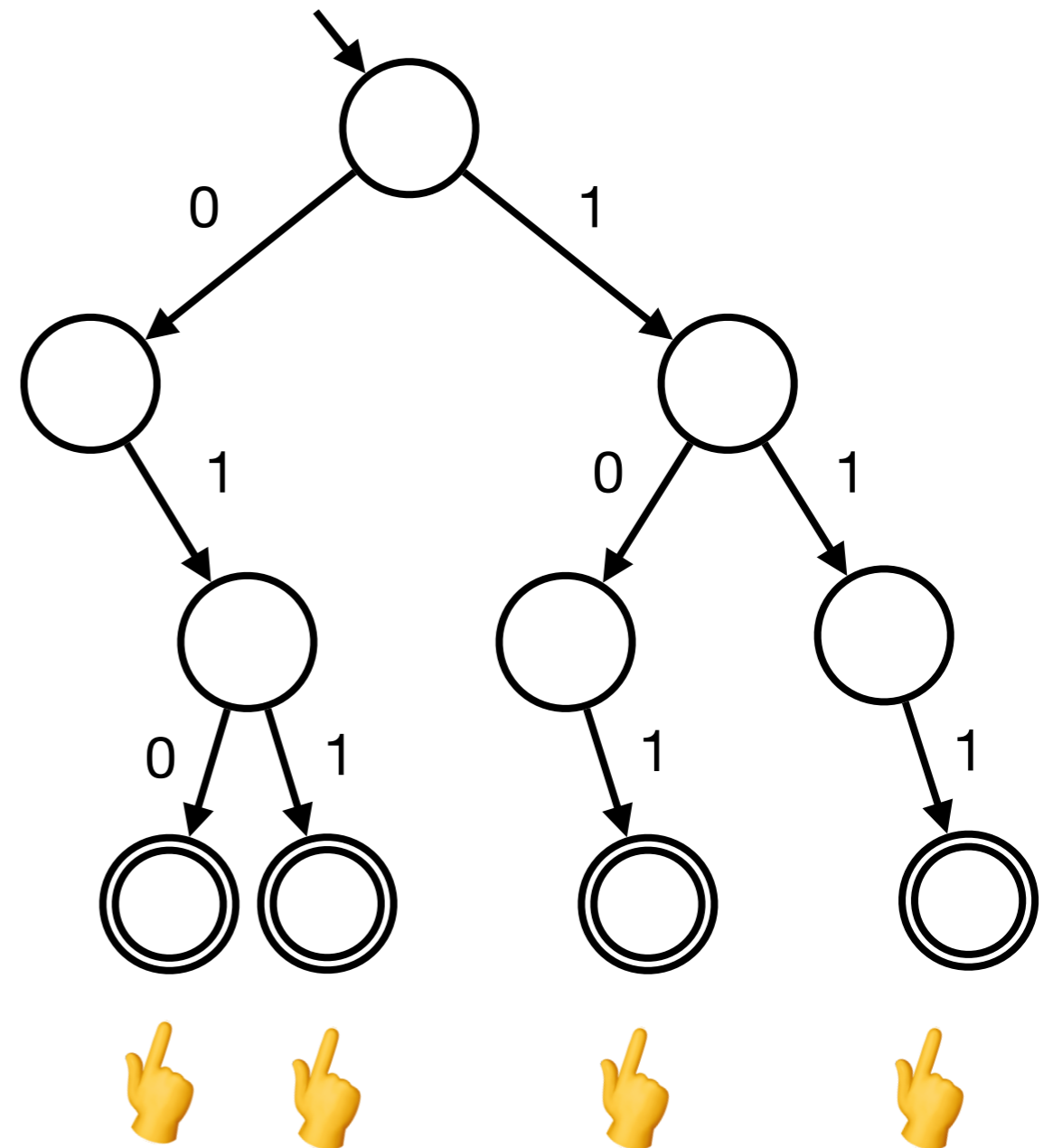
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



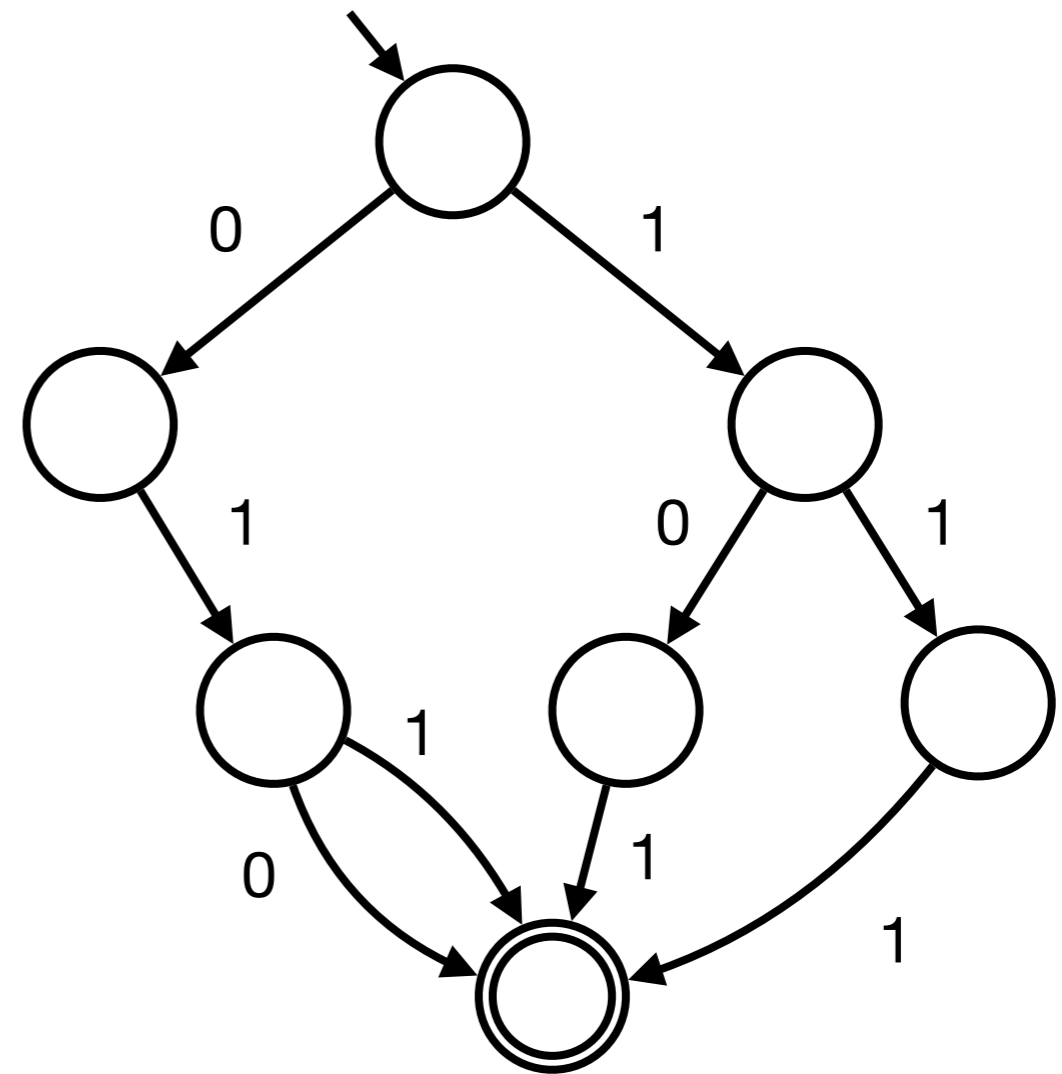
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



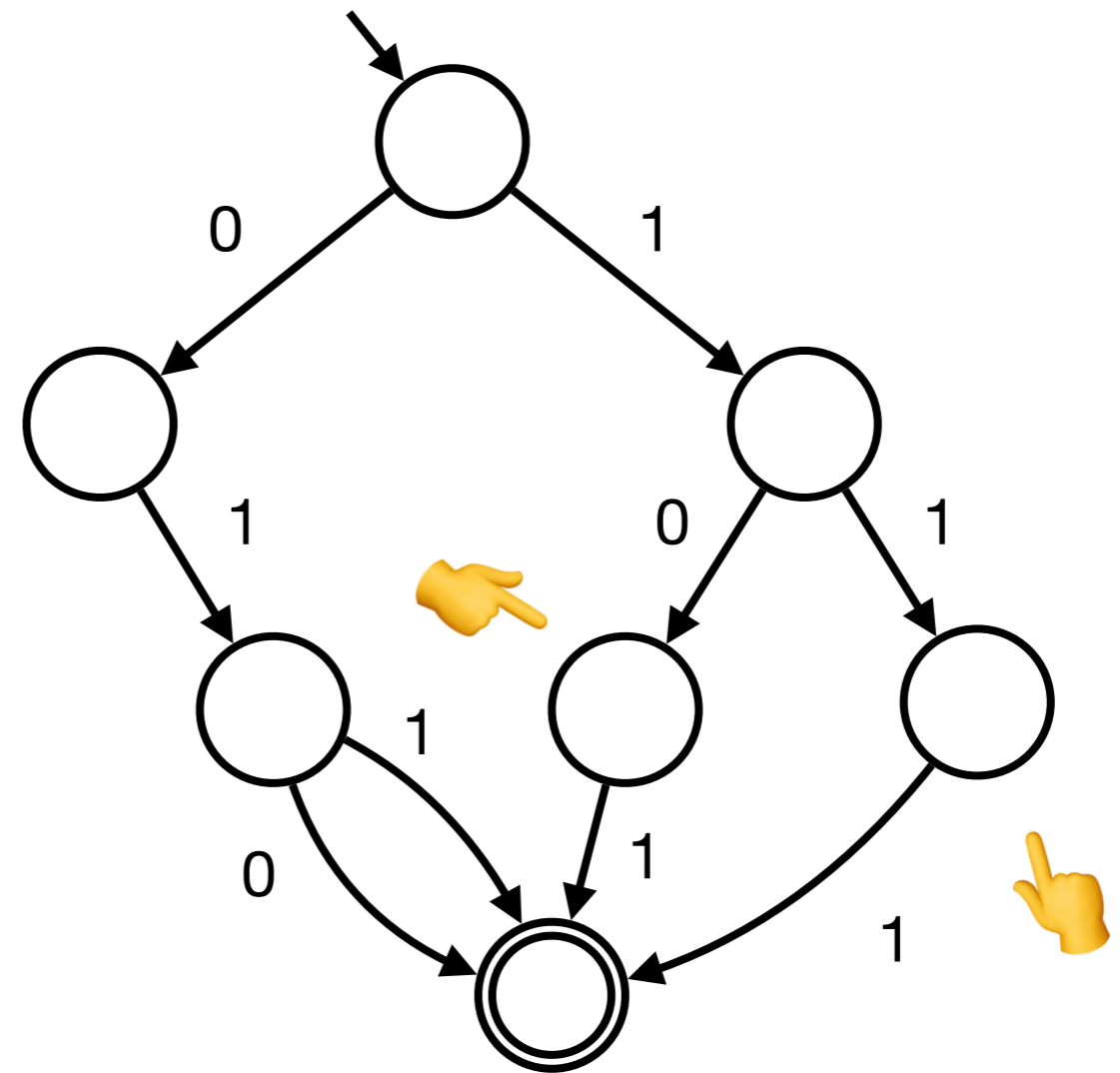
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



# Entiers premiers sur 3 bits

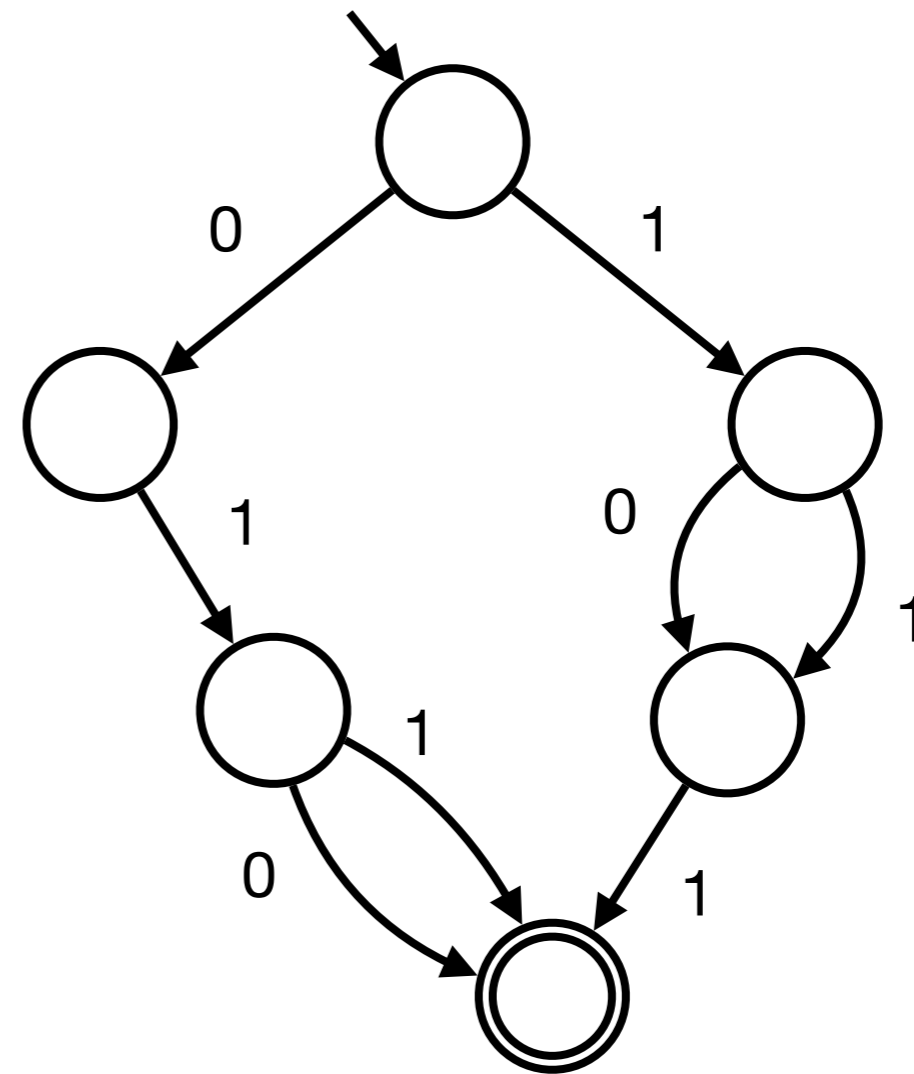
| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |





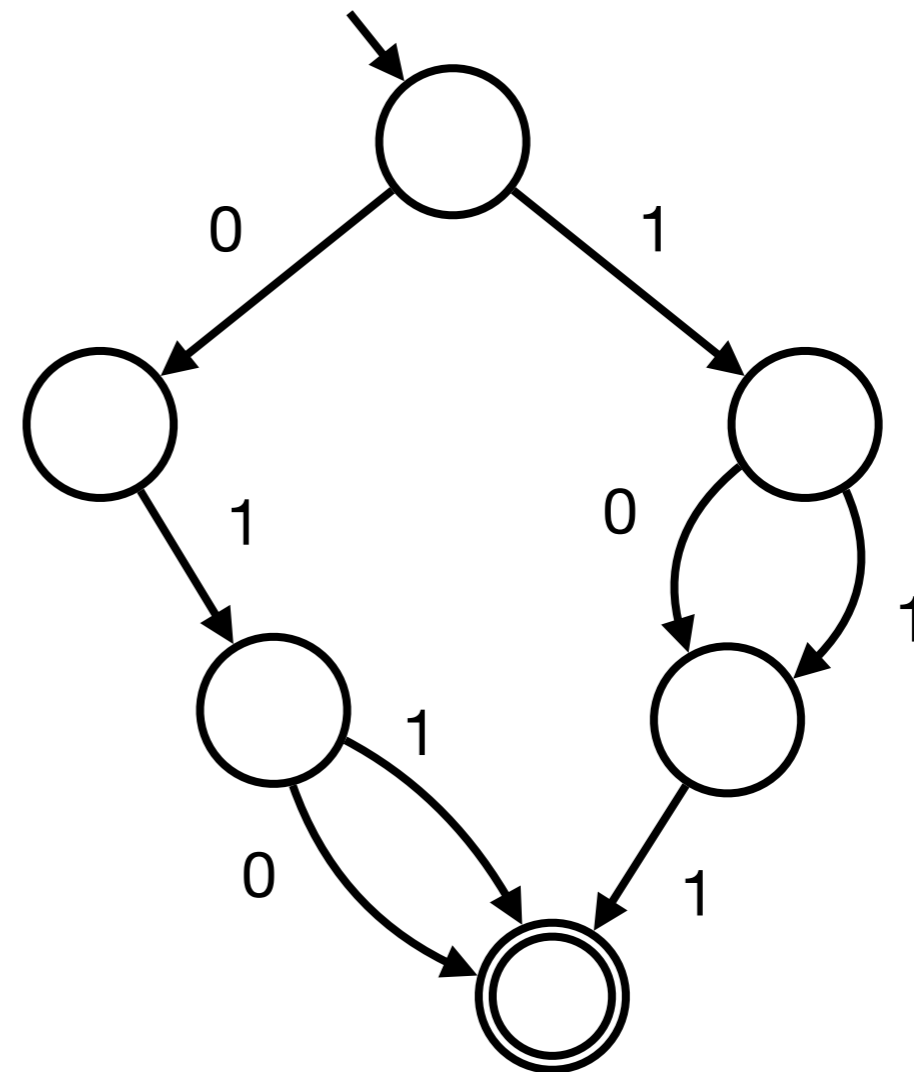
# Entiers premiers sur 3 bits

| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



# Entiers premiers sur 3 bits

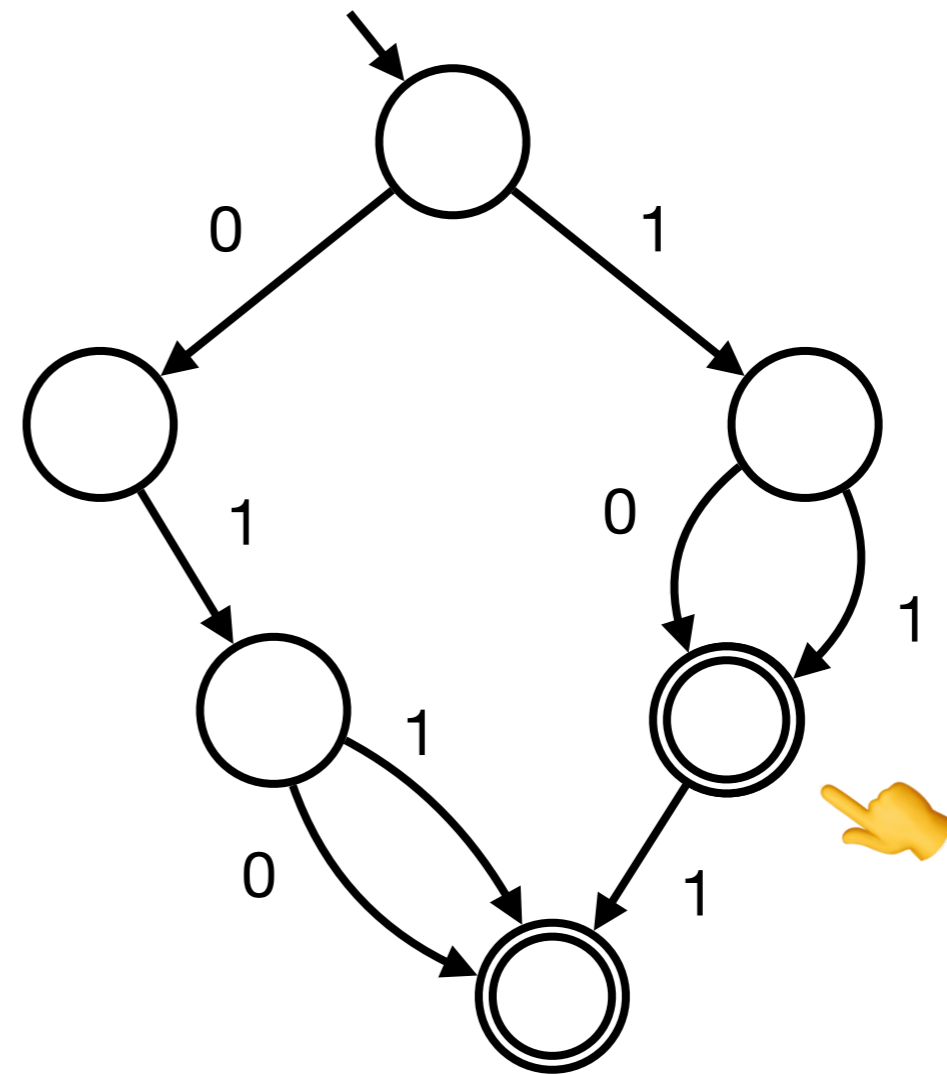
| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



Et si on veut aussi accepter 10 et 11 ?

# Entiers premiers sur 3 bits

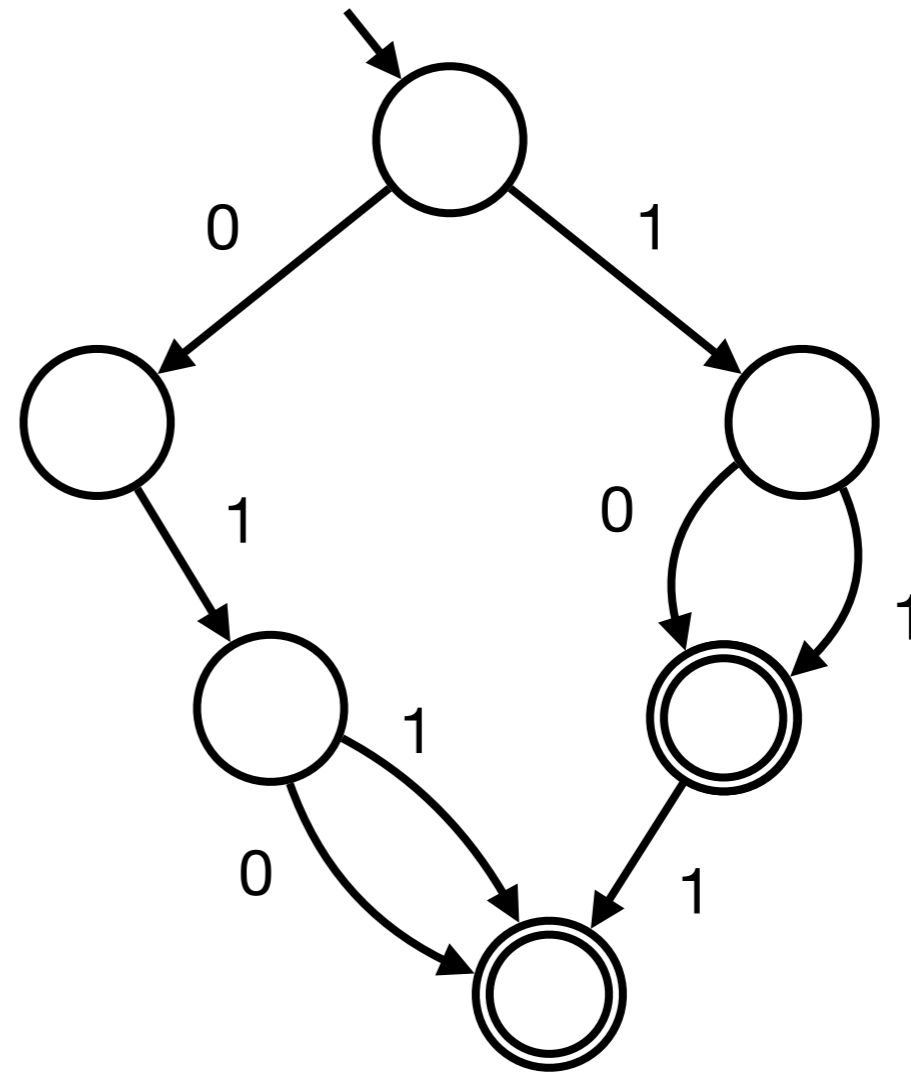
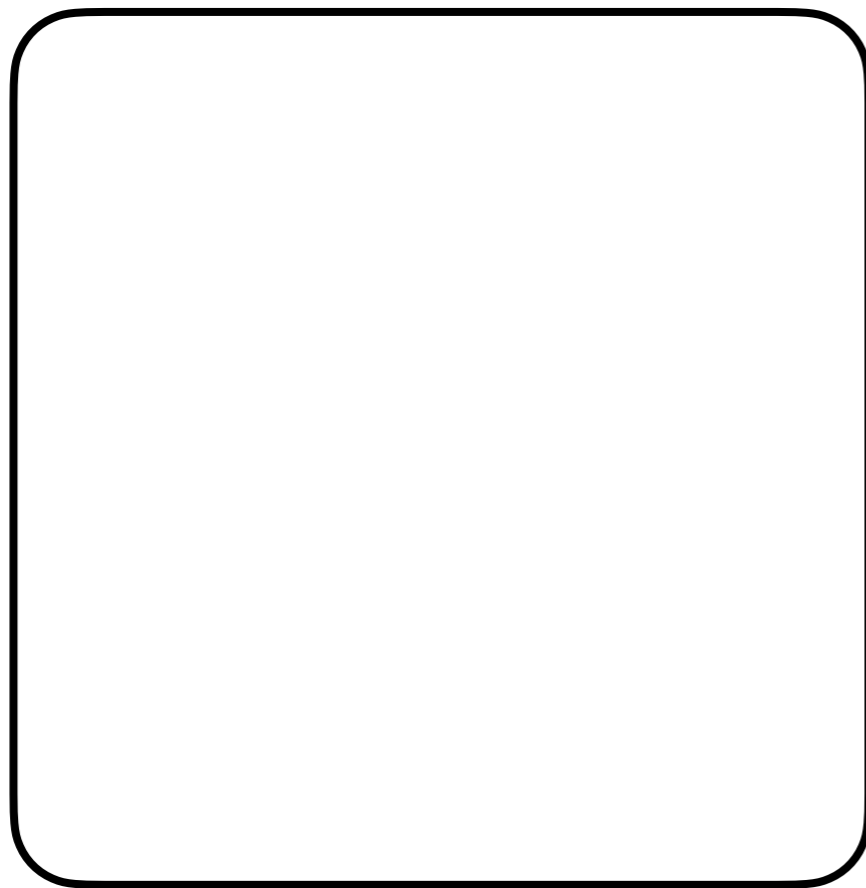
| dec | bin | premier ? |
|-----|-----|-----------|
| 0   | 000 | non       |
| 1   | 001 | non       |
| 2   | 010 | oui       |
| 3   | 011 | oui       |
| 4   | 100 | non       |
| 5   | 101 | oui       |
| 6   | 110 | non       |
| 7   | 111 | oui       |



Et si on veut aussi accepter 10 et 11 ?

# Entiers premiers sur 3 bits

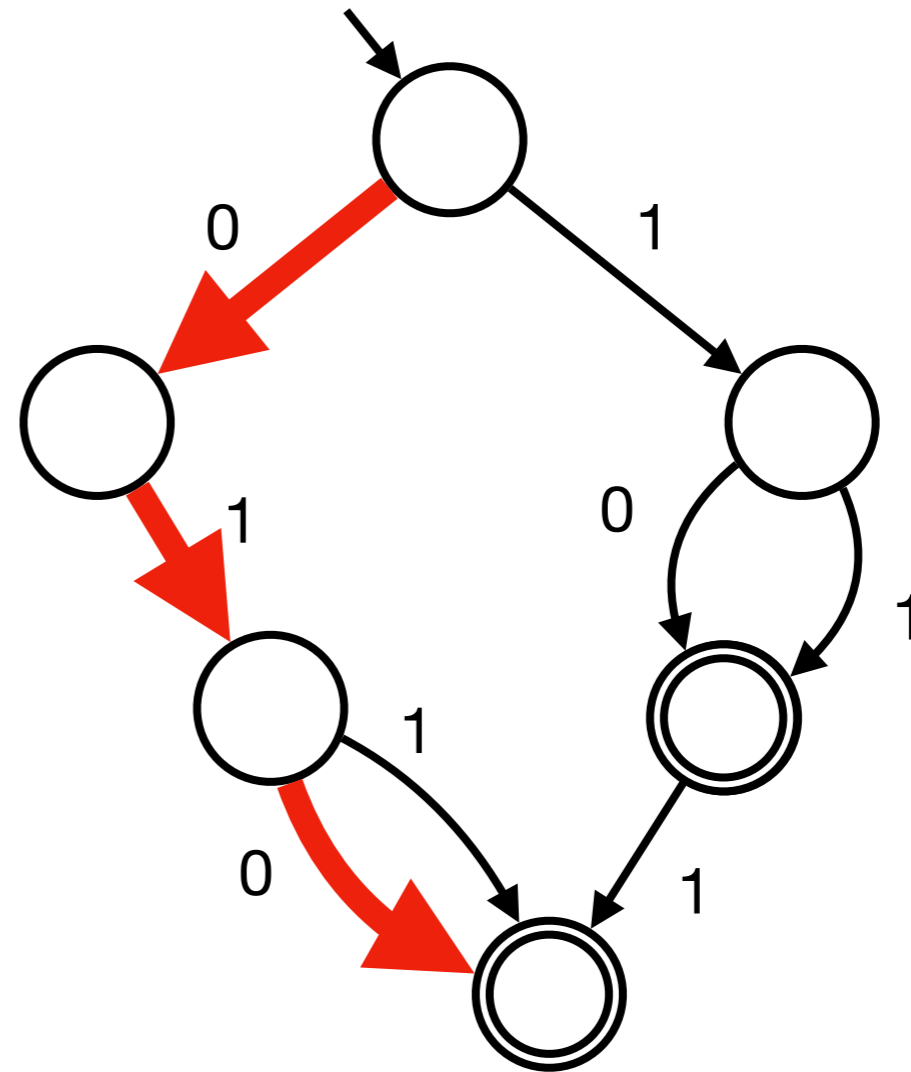
mots acceptés



# Entiers premiers sur 3 bits

mots acceptés

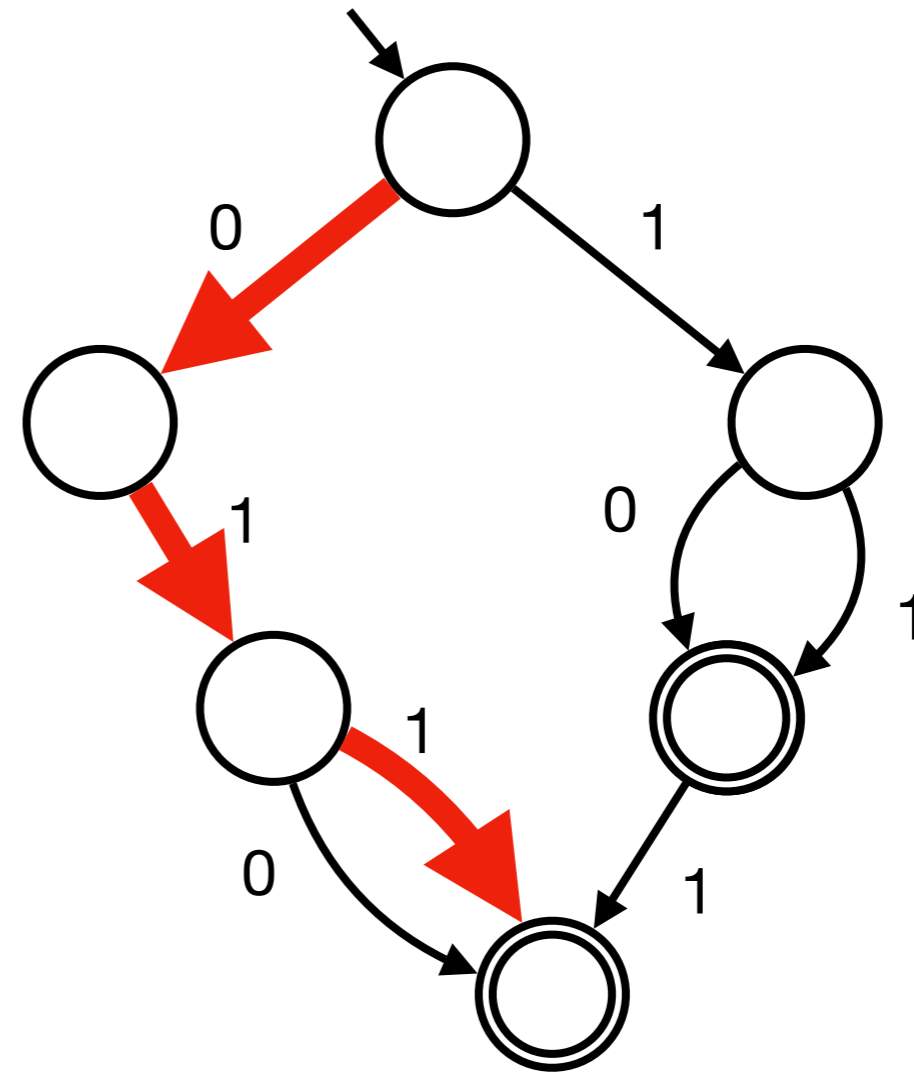
010



# Entiers premiers sur 3 bits

mots acceptés

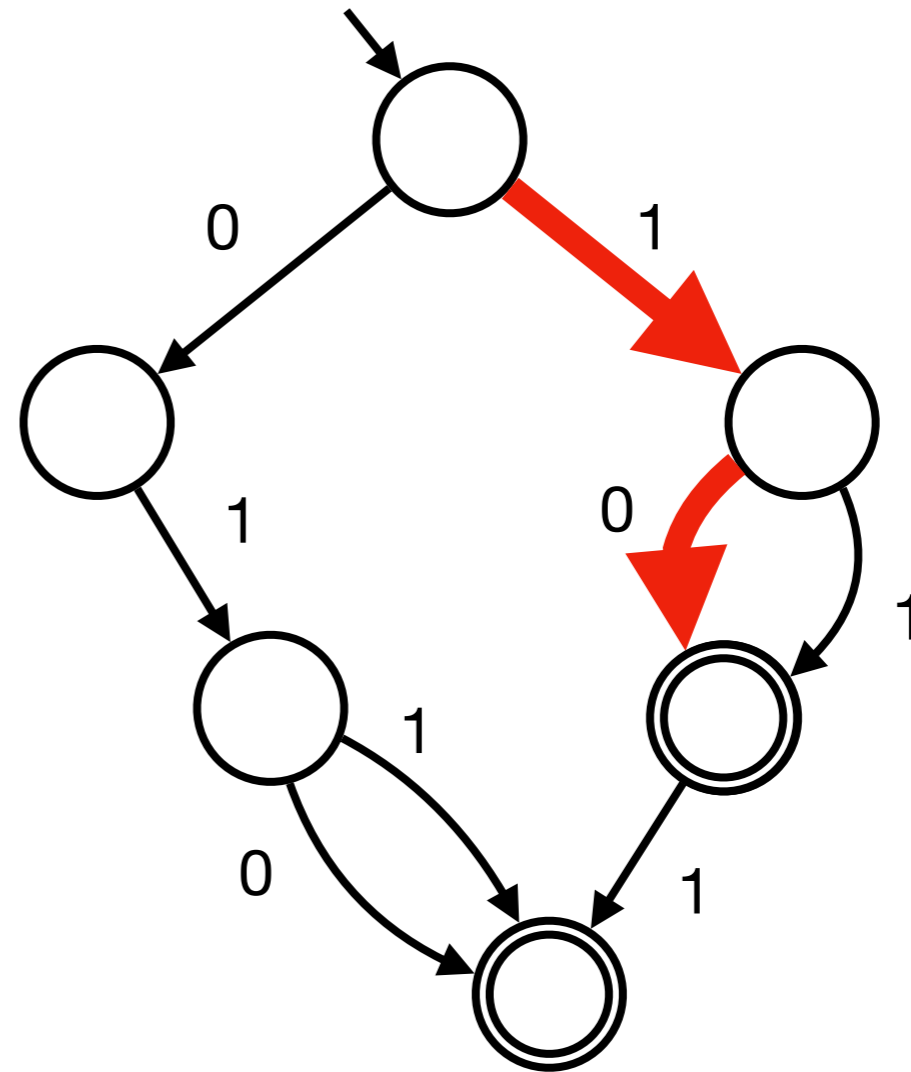
010      011



# Entiers premiers sur 3 bits

mots acceptés

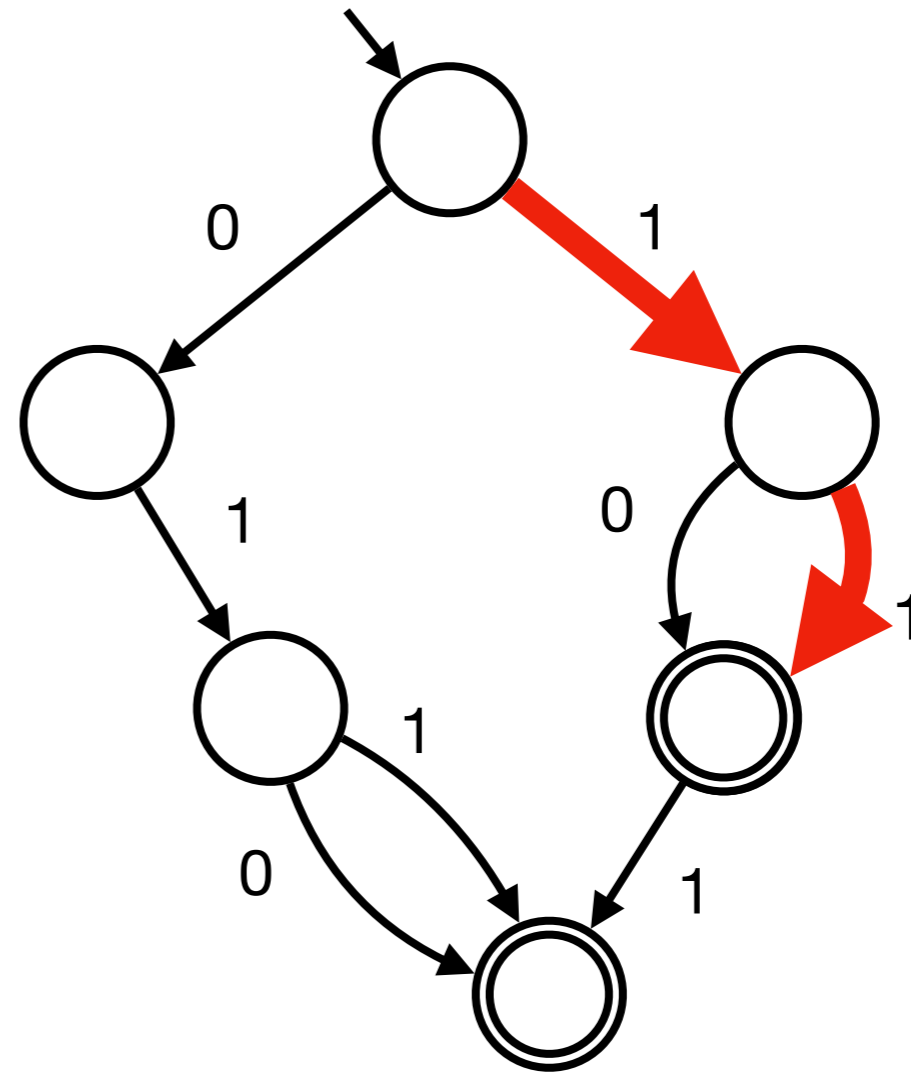
010      011  
10



# Entiers premiers sur 3 bits

mots acceptés

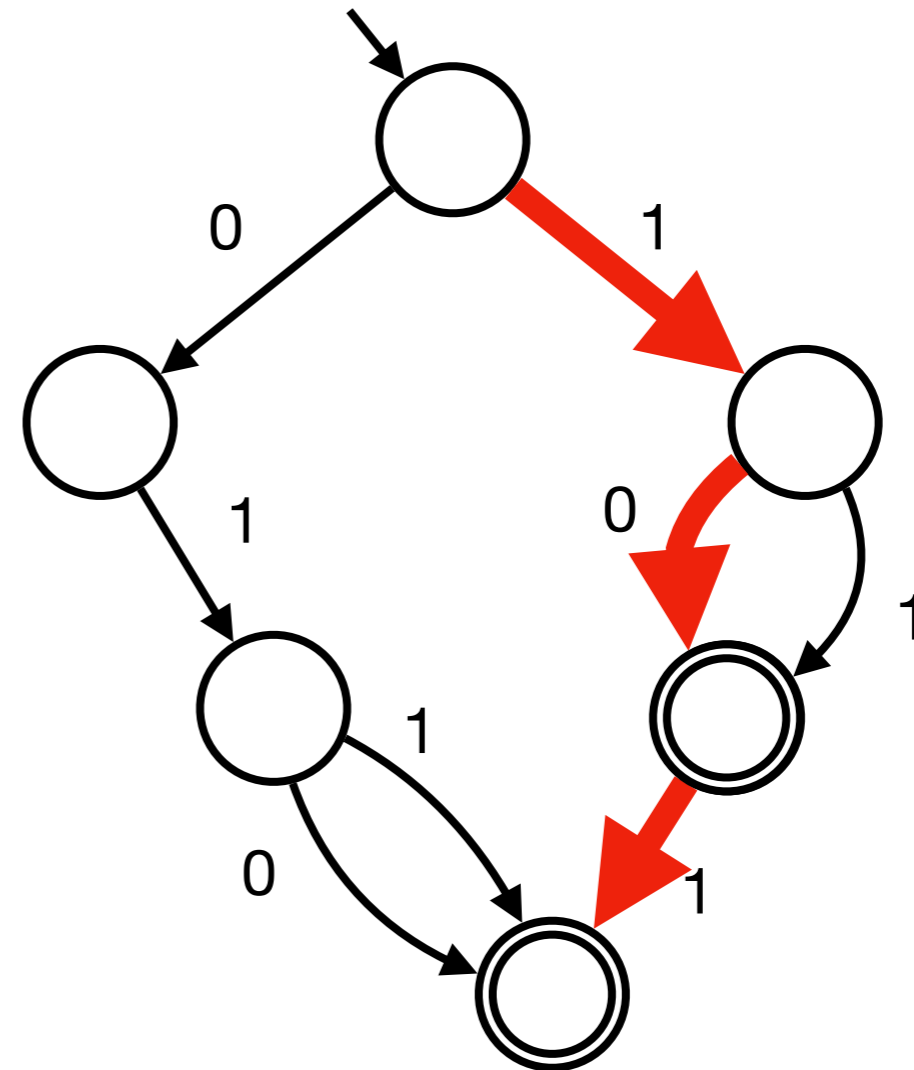
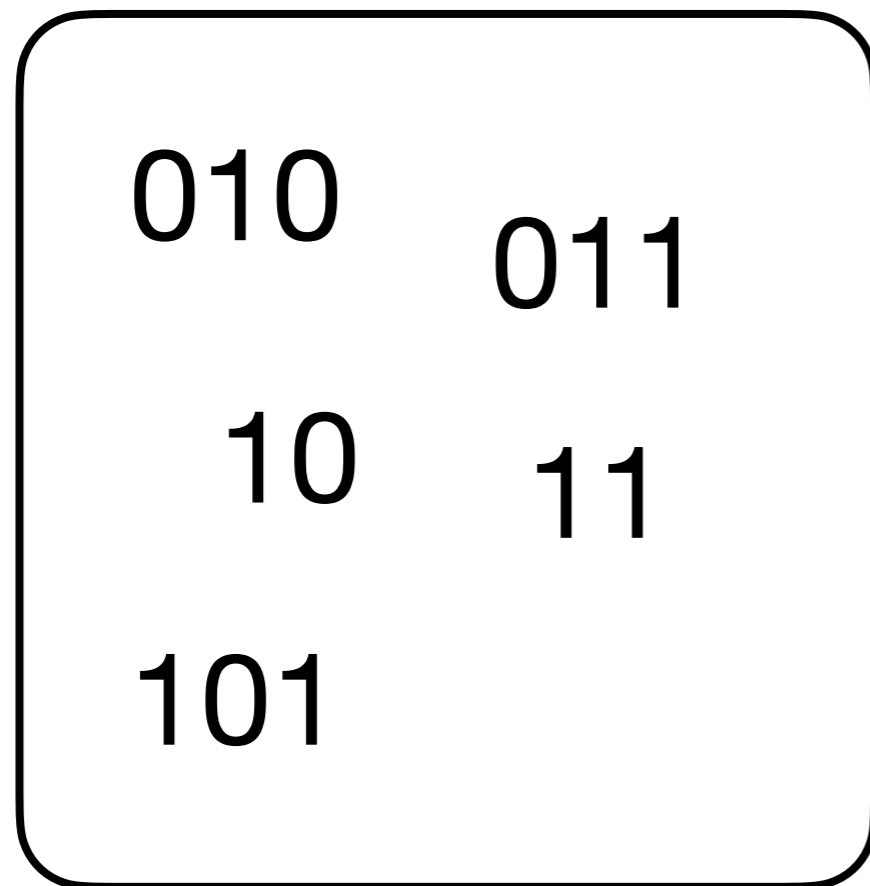
|     |     |
|-----|-----|
| 010 | 011 |
| 10  | 11  |





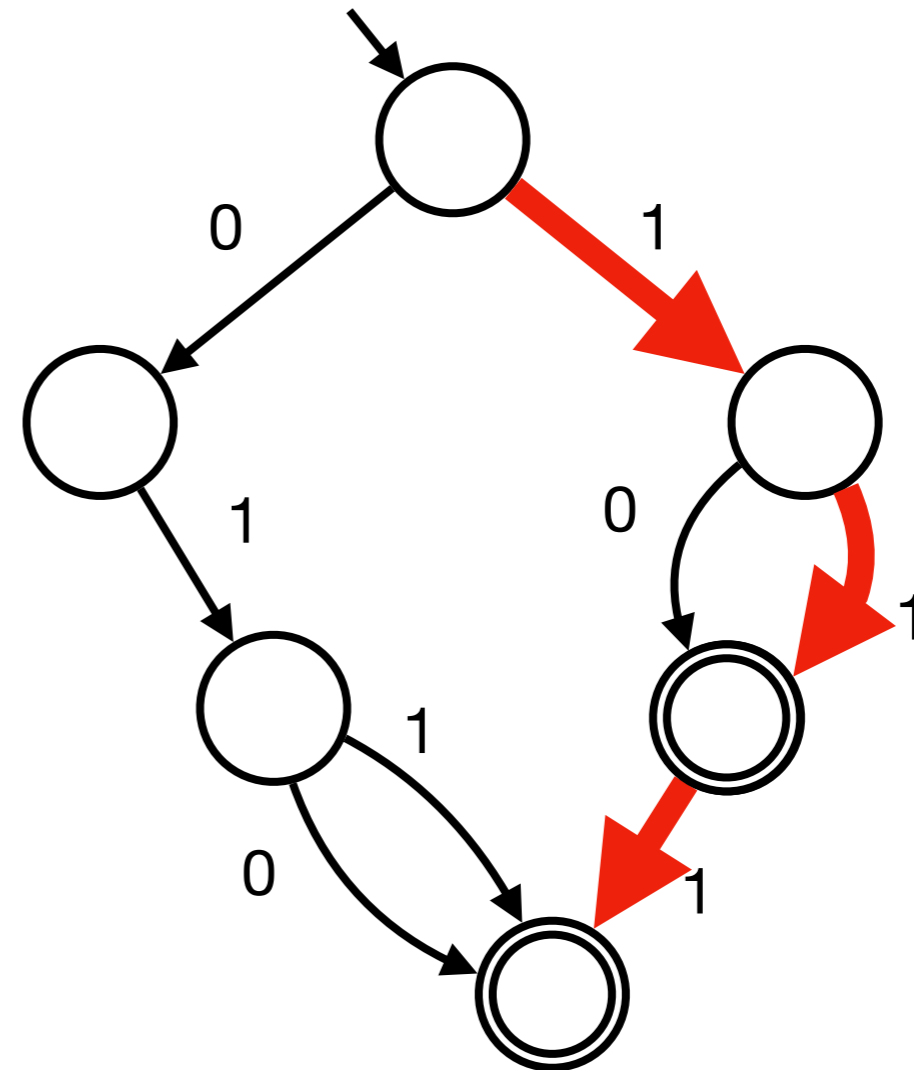
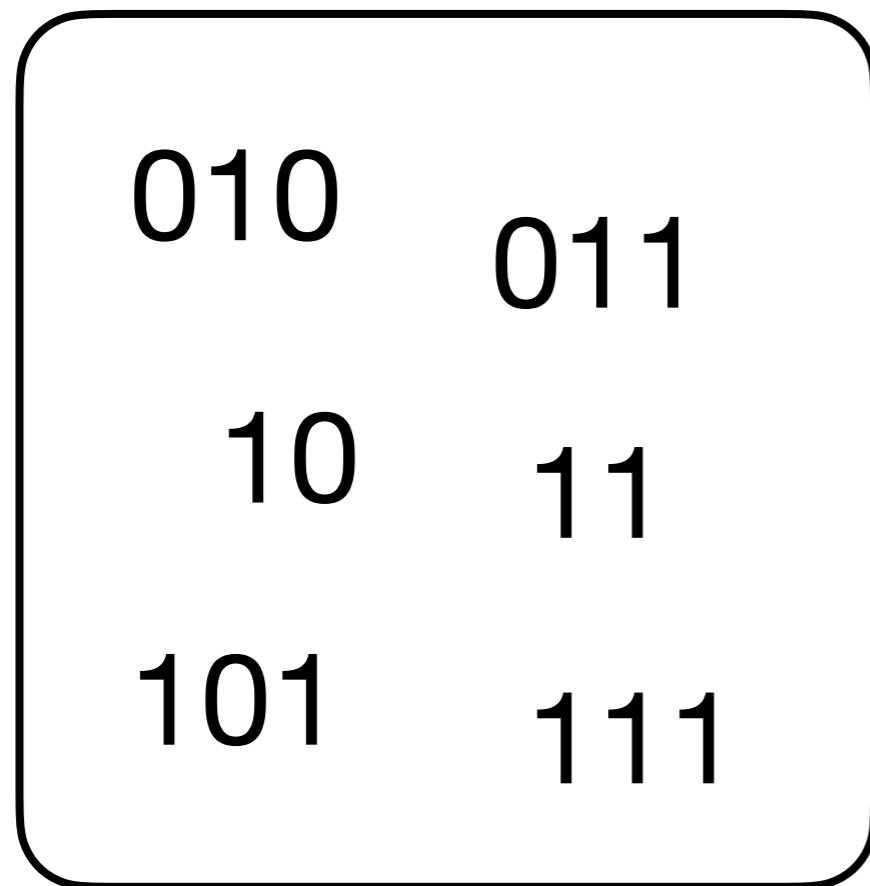
# Entiers premiers sur 3 bits

mots acceptés



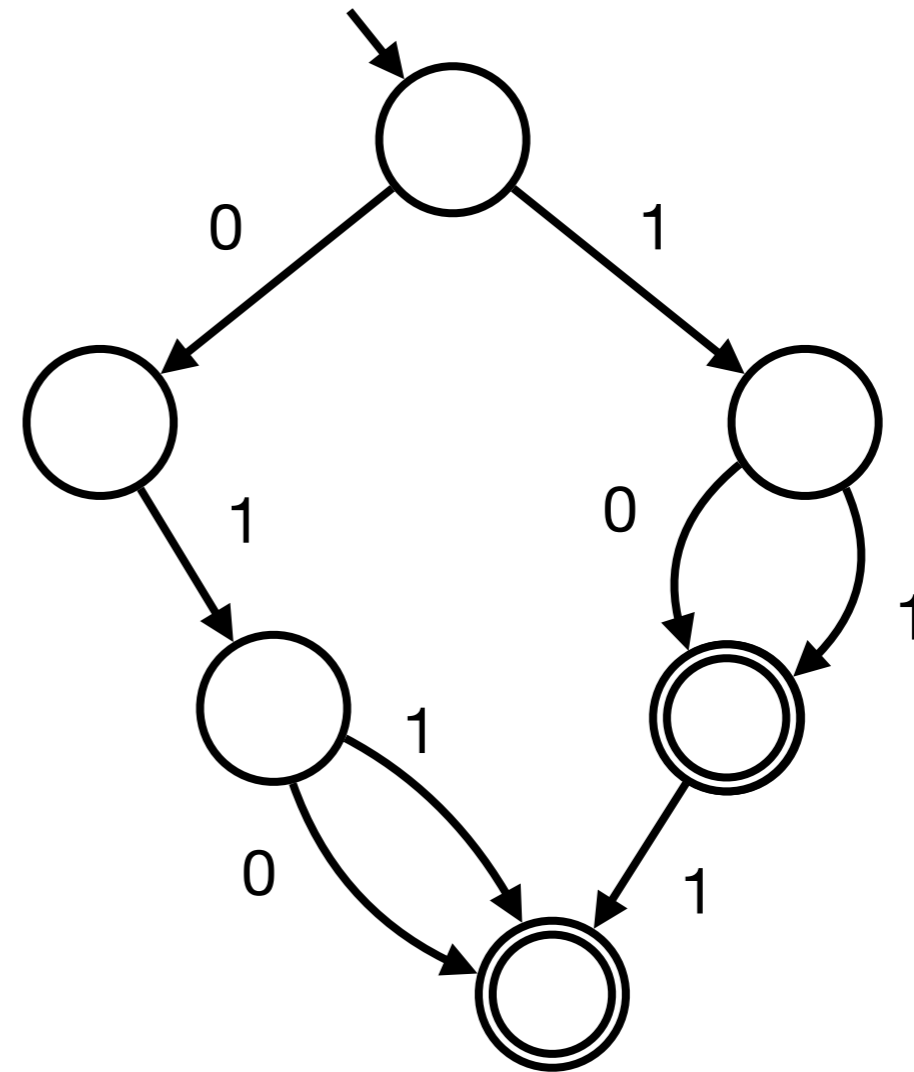
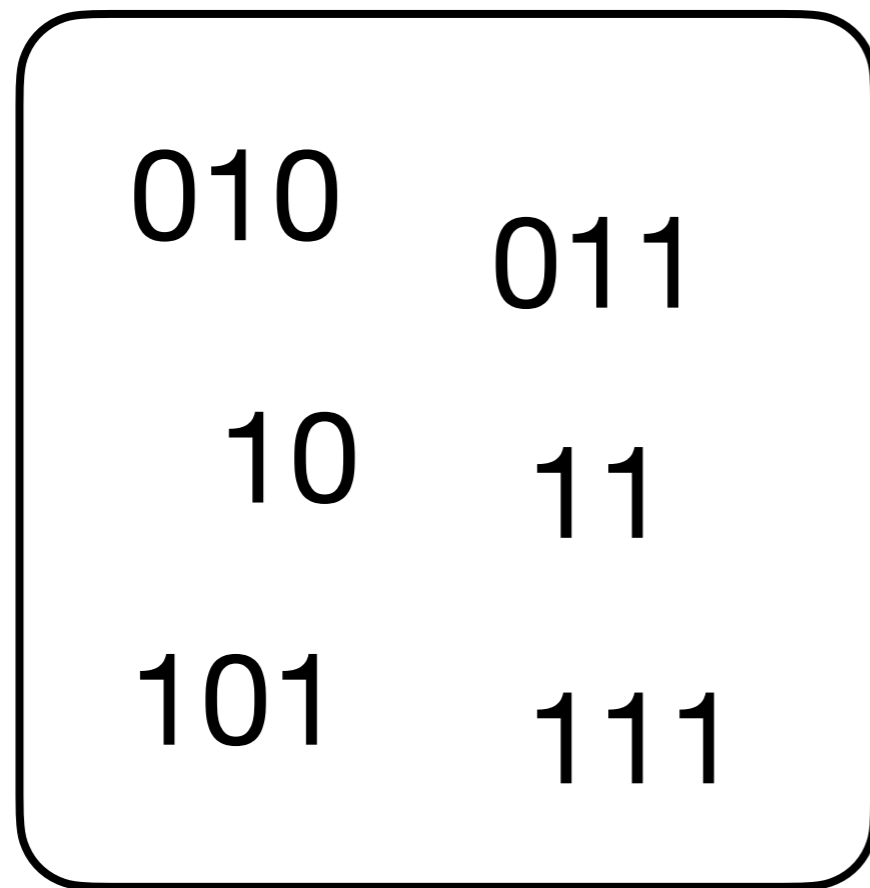
# Entiers premiers sur 3 bits

mots acceptés



# Entiers premiers sur 3 bits

mots acceptés



# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

mots acceptés



0

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

mots acceptés

0 10

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

mots acceptés

0 10

110

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

mots acceptés

0 10

110

1110



# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

mots acceptés

0 10 11110

110

1110

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

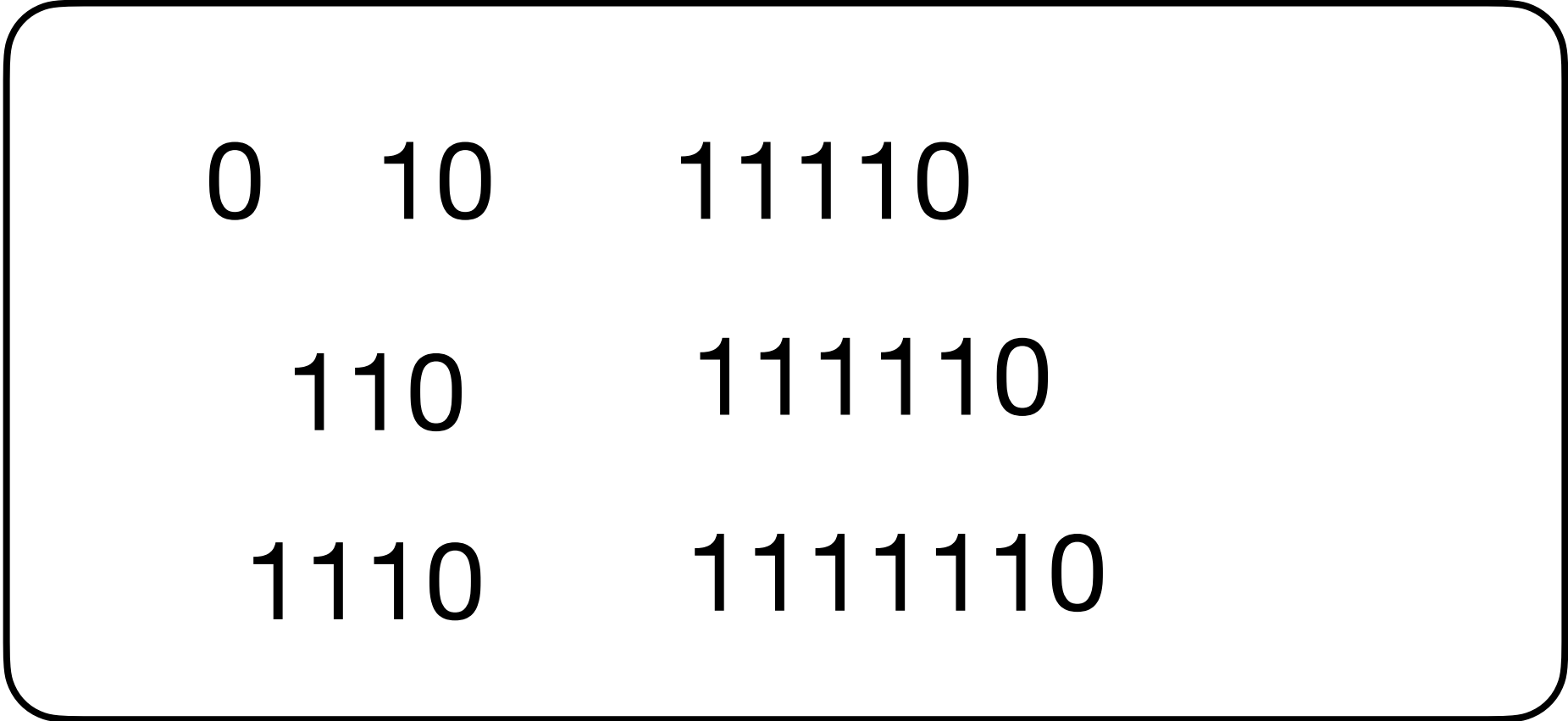
mots acceptés

0 10 11110  
110 111110  
1110

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

mots acceptés

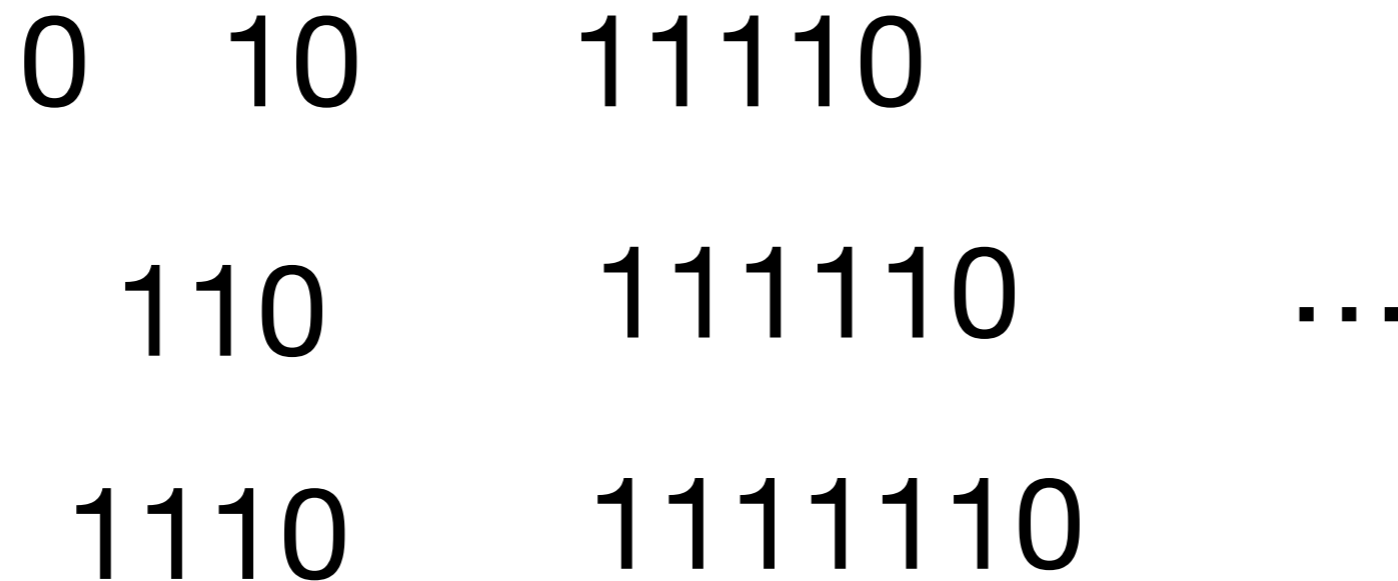


0 10 1110  
110 11110  
1110 111110

# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0

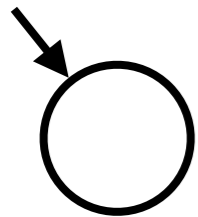
mots acceptés



0 10 1110  
110 11110 ...  
1110 111110

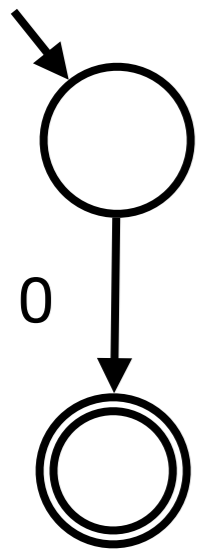
# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zéro ou plus 1 et terminent par 0



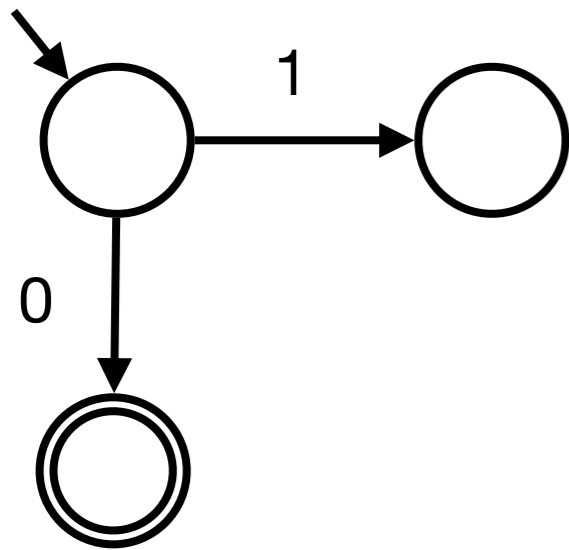
# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par 0 ou plus 1 et terminent par 0



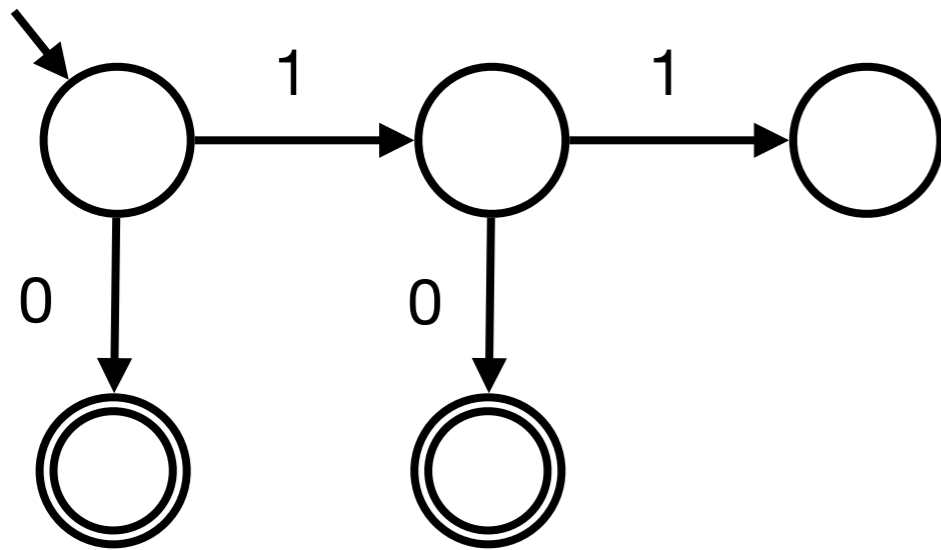
# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zéro ou plus 1 et terminent par 0



# Comment accepter un ensemble infini de mots ?

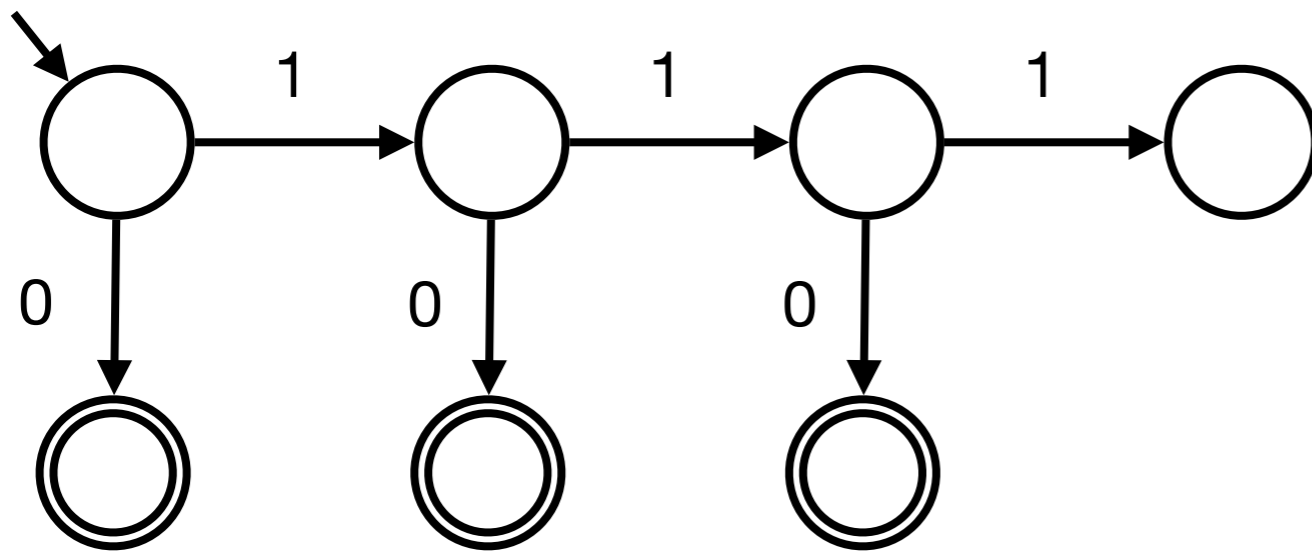
Exemple : les séquences de 0 et 1 qui commencent par zéro ou plus 1 et terminent par 0





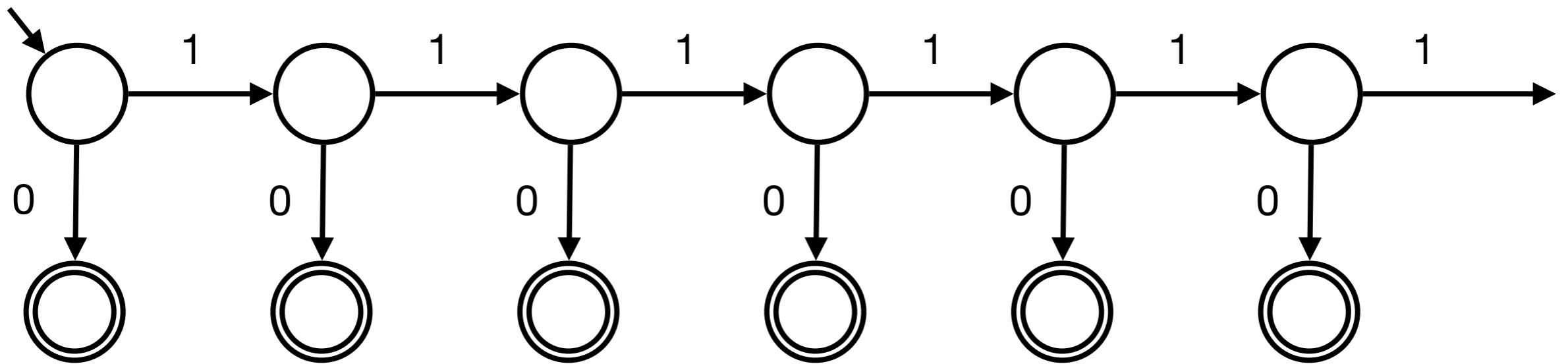
# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zéro ou plus 1 et terminent par 0



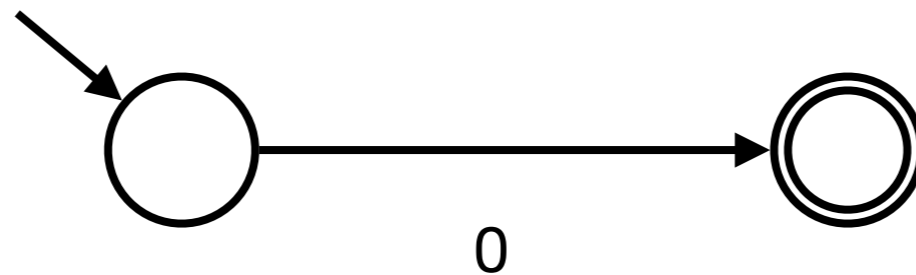
# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zéro ou plus 1 et terminent par 0



# Comment accepter un ensemble infini de mots ?

Exemple : les séquences de 0 et 1 qui commencent par zéro ou plus 1 et terminent par 0

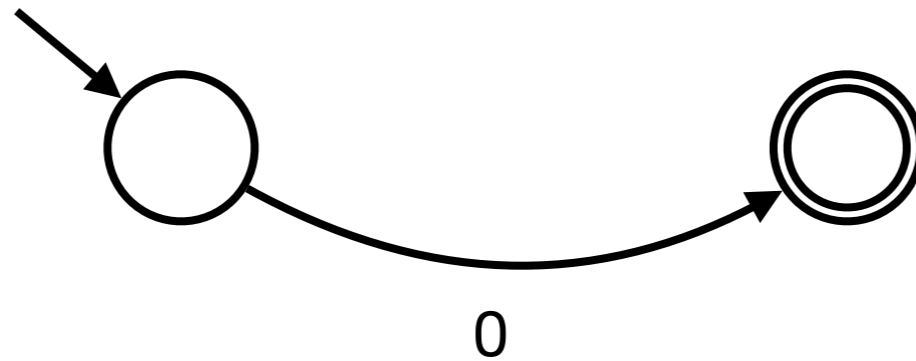


# Comment accepter un ensemble infini de mots ?

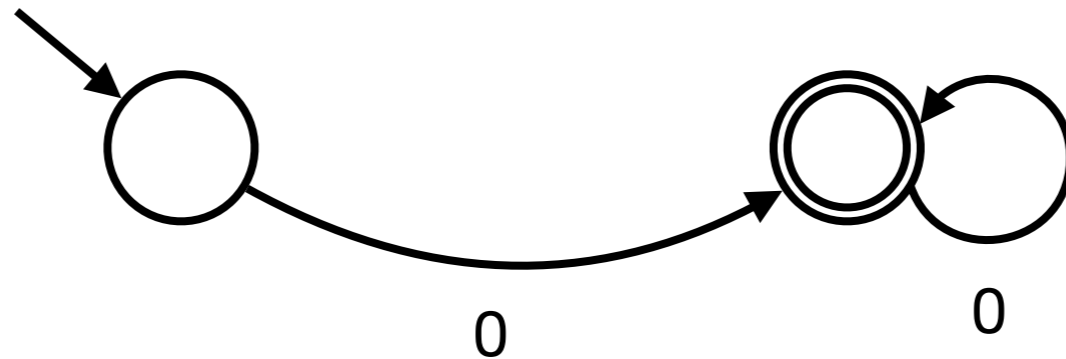
Exemple : les séquences de 0 et 1 qui commencent par zero ou plus 1 et terminent par 0



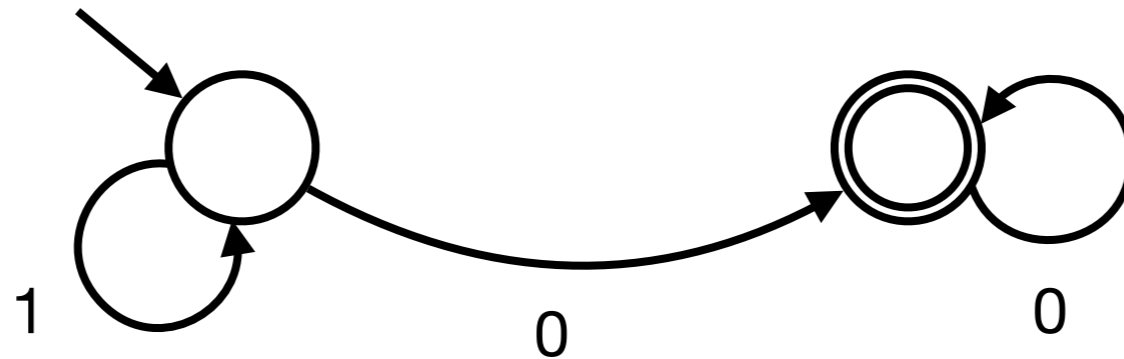
# Automate pour les séquences qui se terminent par 0



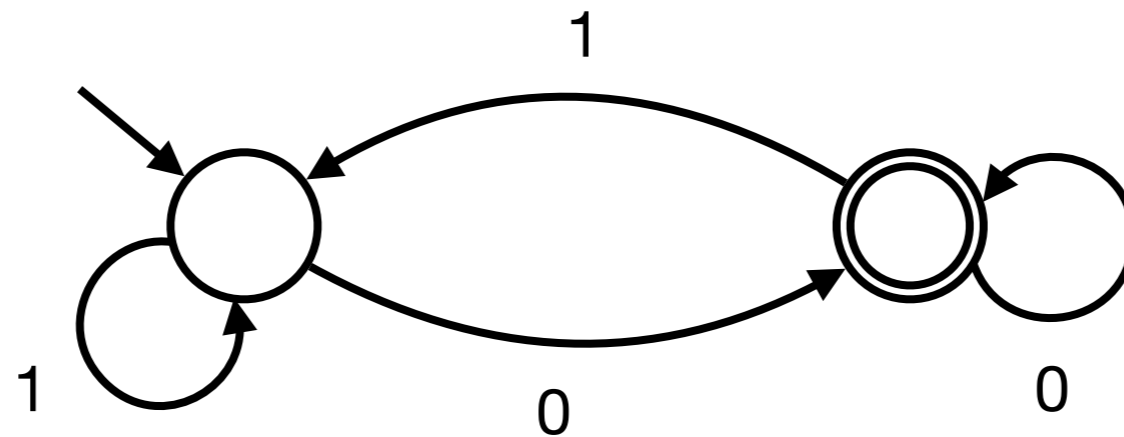
# Automate pour les séquences qui se terminent par 0



# Automate pour les séquences qui se terminent par 0



# Automate pour les séquences qui se terminent par 0

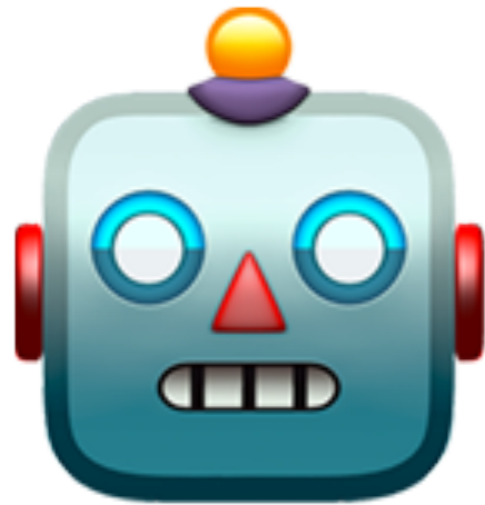




# Recherche de mots dans un texte



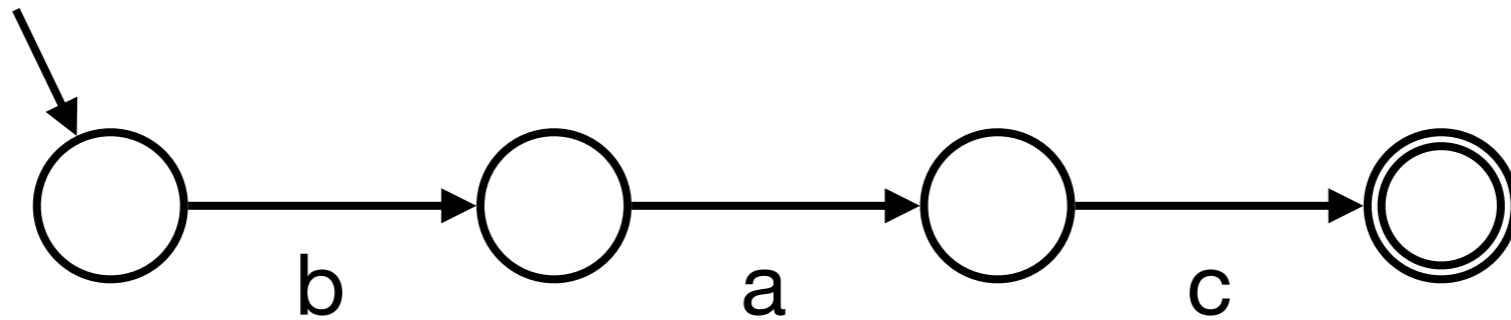
# Recherche de mots dans un texte



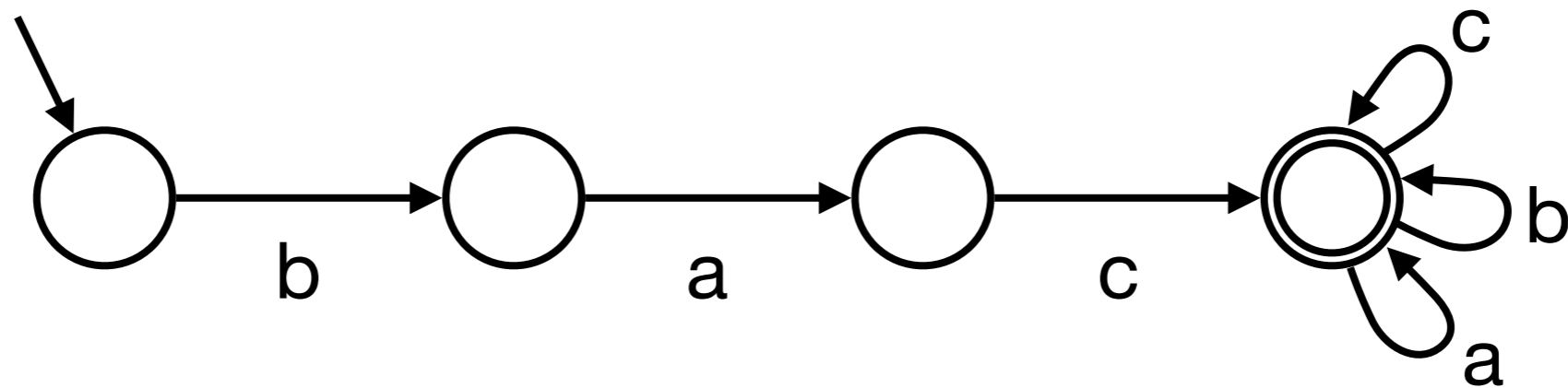
# Exemple de recherche

- On utilise, pour simplicité, l'alphabet réduit {a, b, c}
- On veut décrire un automate qui accepte un texte si et seulement si il contient le mot « bac »
- Textes acceptés : ab**ac**aa, **ba**cccb, bcc**ba**c
- Textes non acceptés : abcaaa, cab

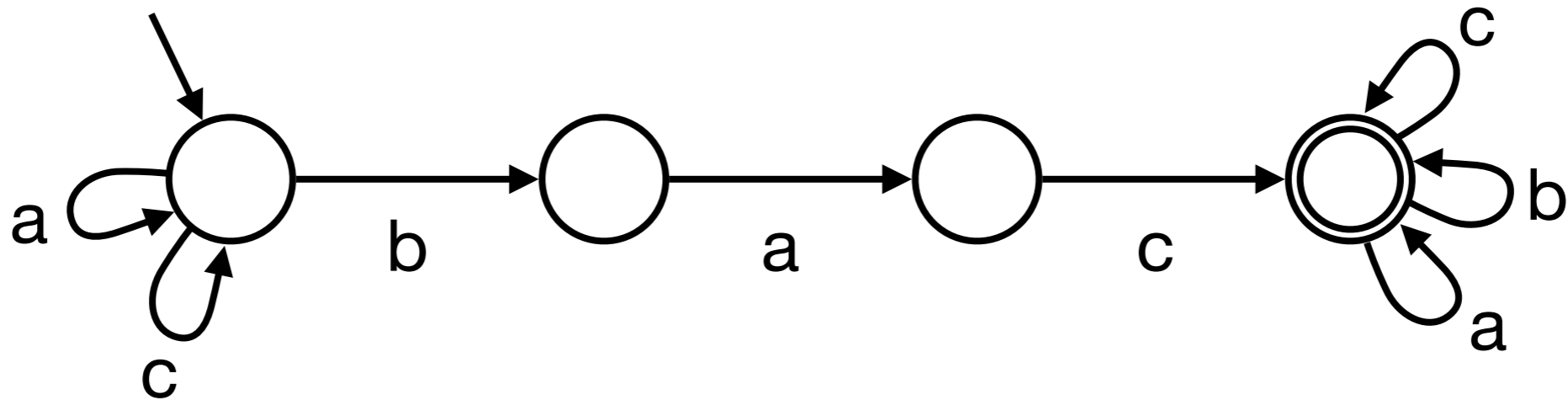
# Automate pour le textes qui contient « bac »



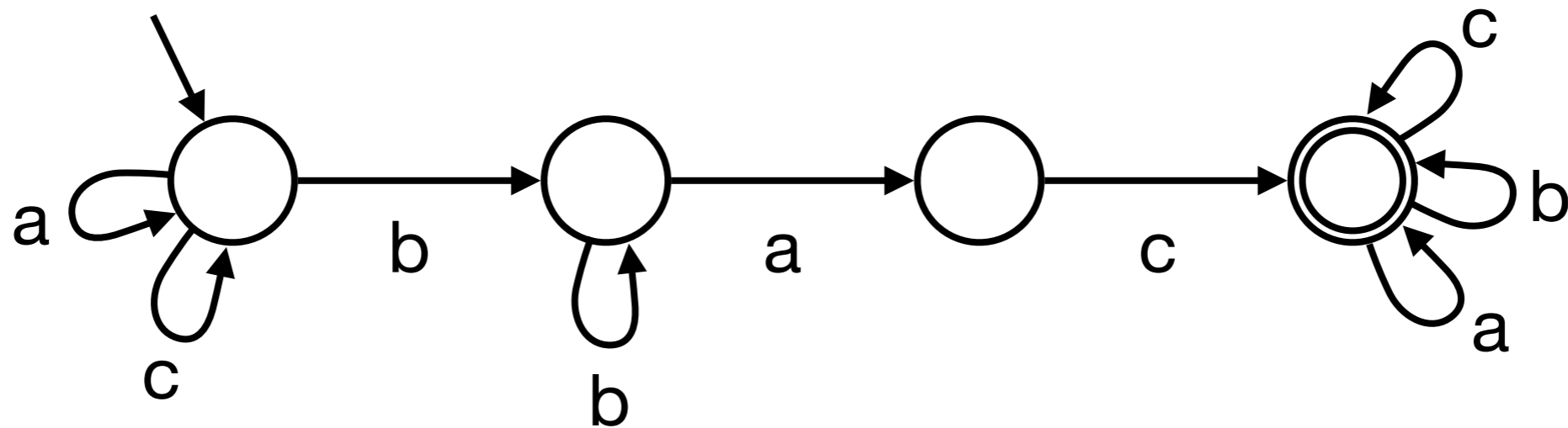
# Automate pour le textes qui contient « bac »



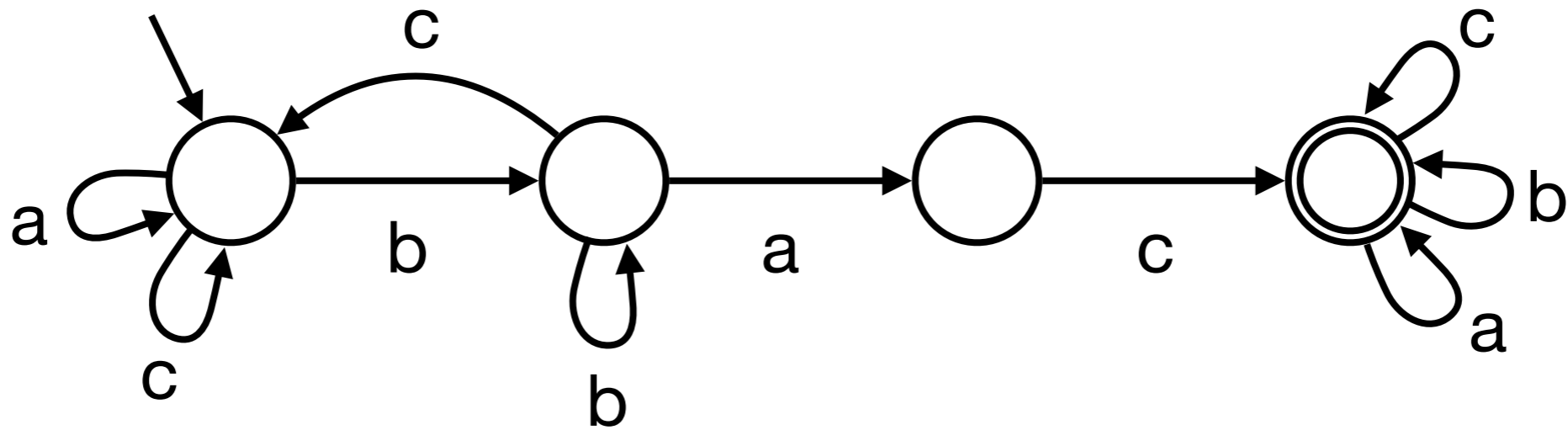
# Automate pour le textes qui contient « bac »



# Automate pour le textes qui contient « bac »

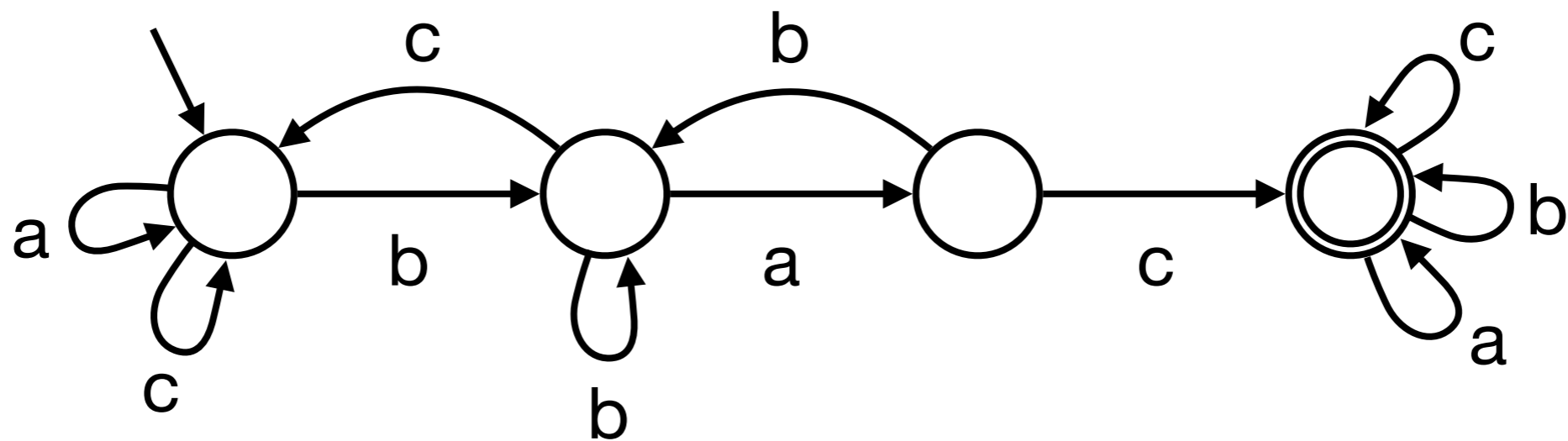


# Automate pour le textes qui contient « bac »

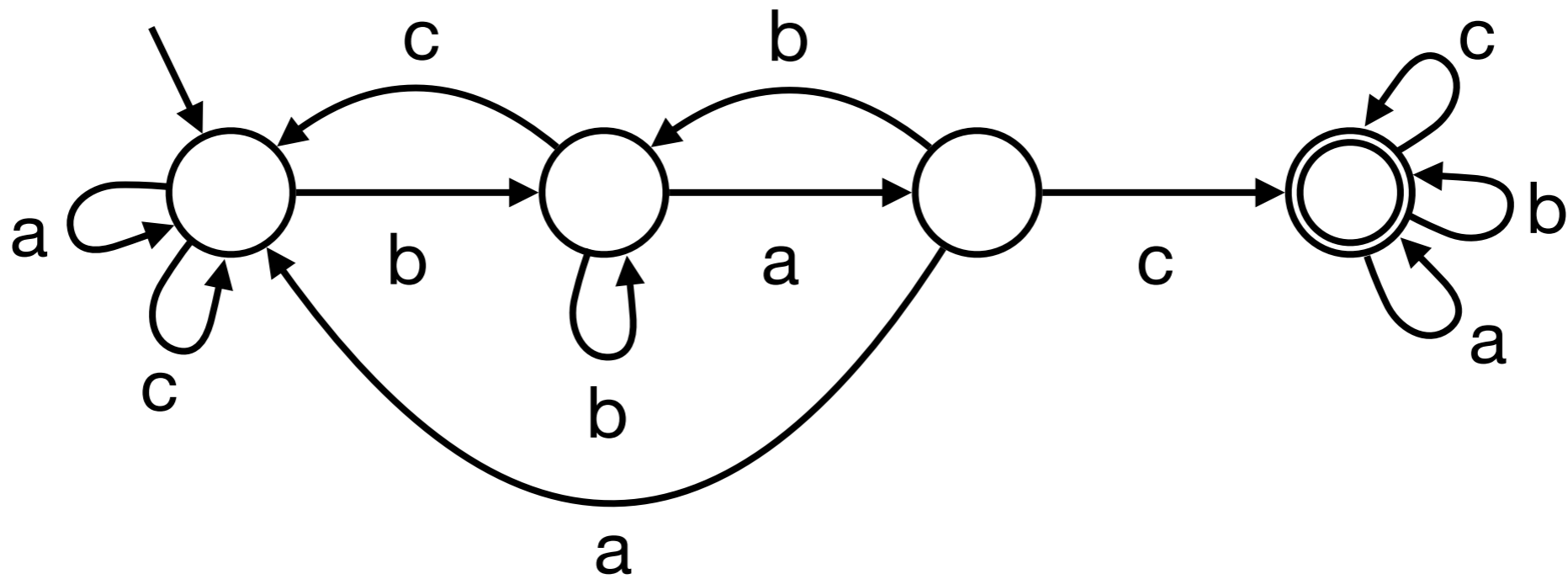




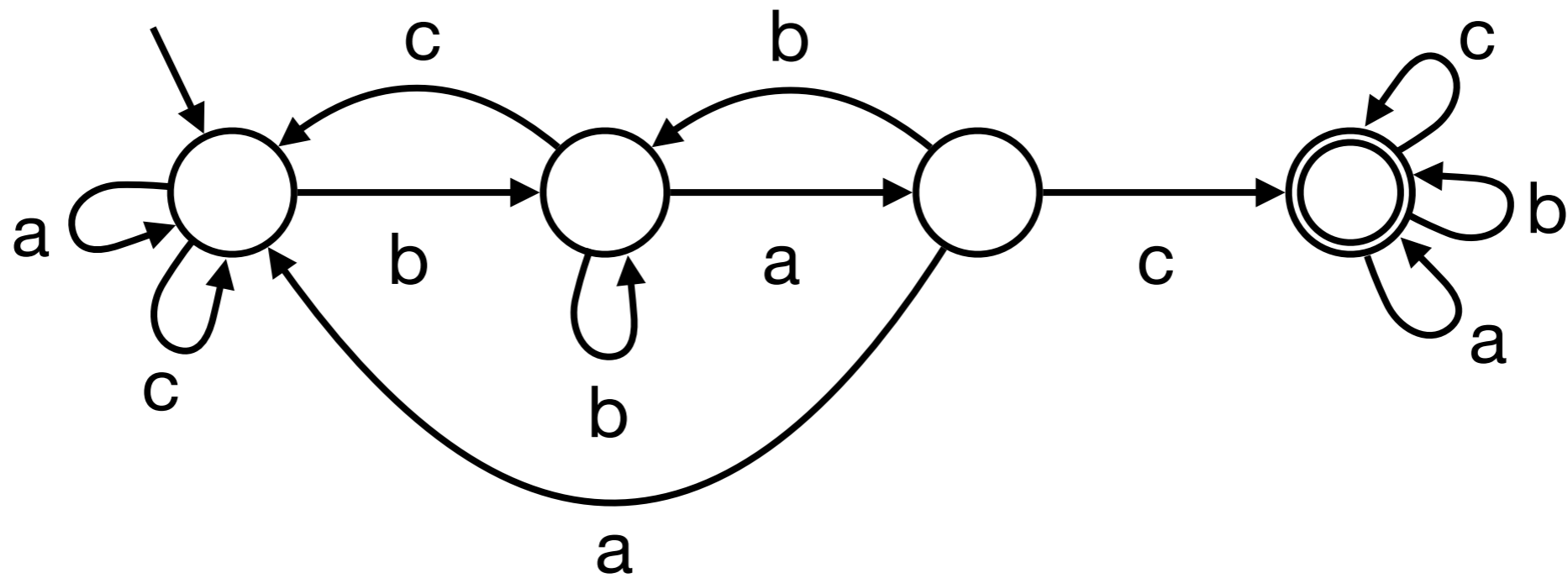
# Automate pour le textes qui contient « bac »



# Automate pour le textes qui contient « bac »

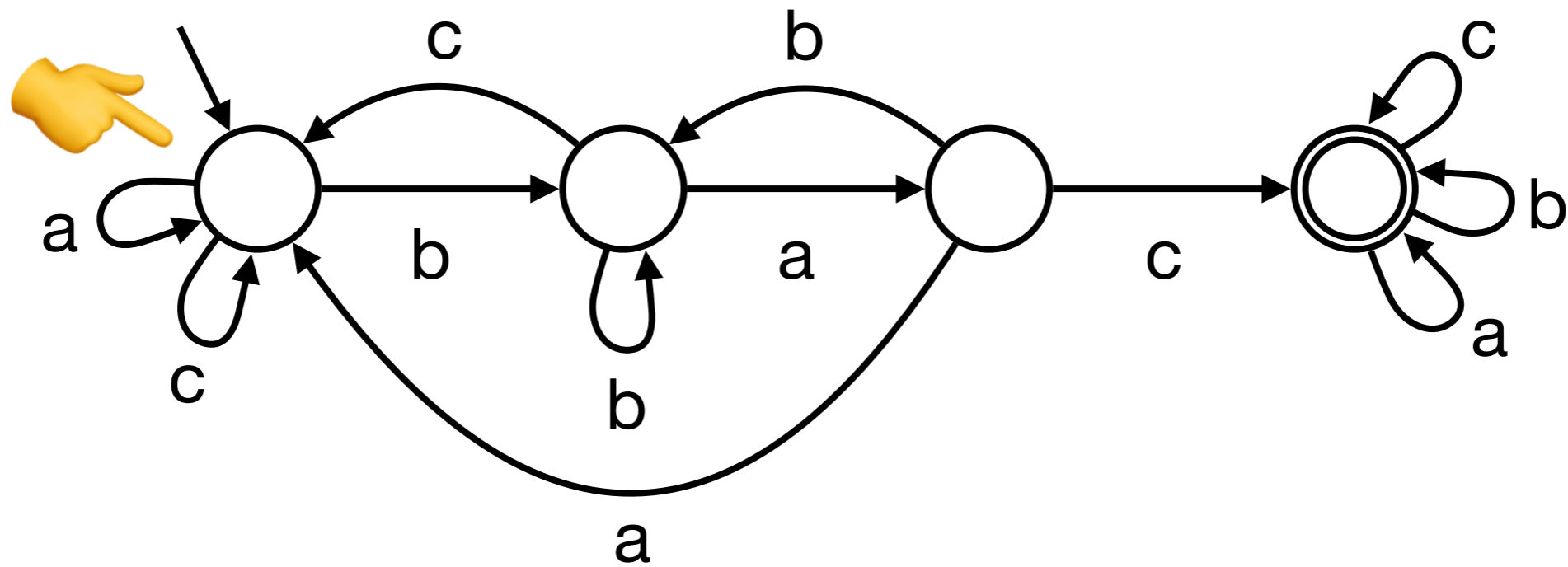


# Exemples de recherche



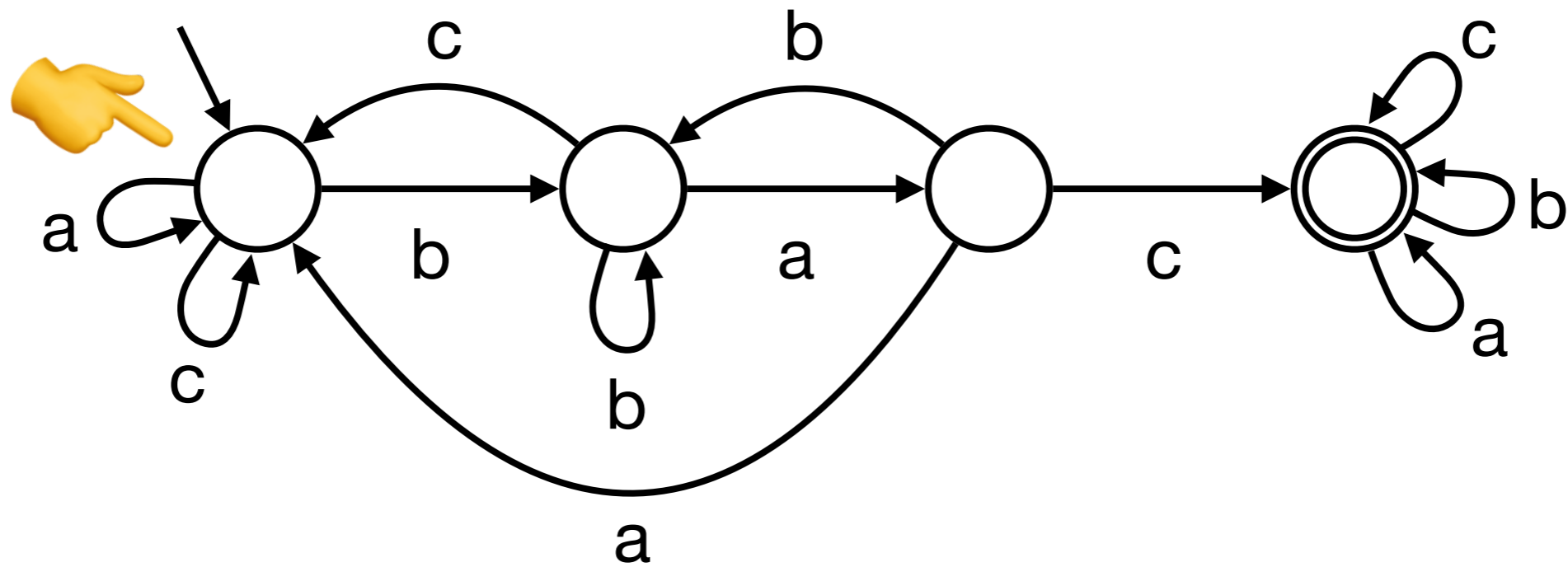
abaca<sup>3</sup>a

# Exemples de recherche



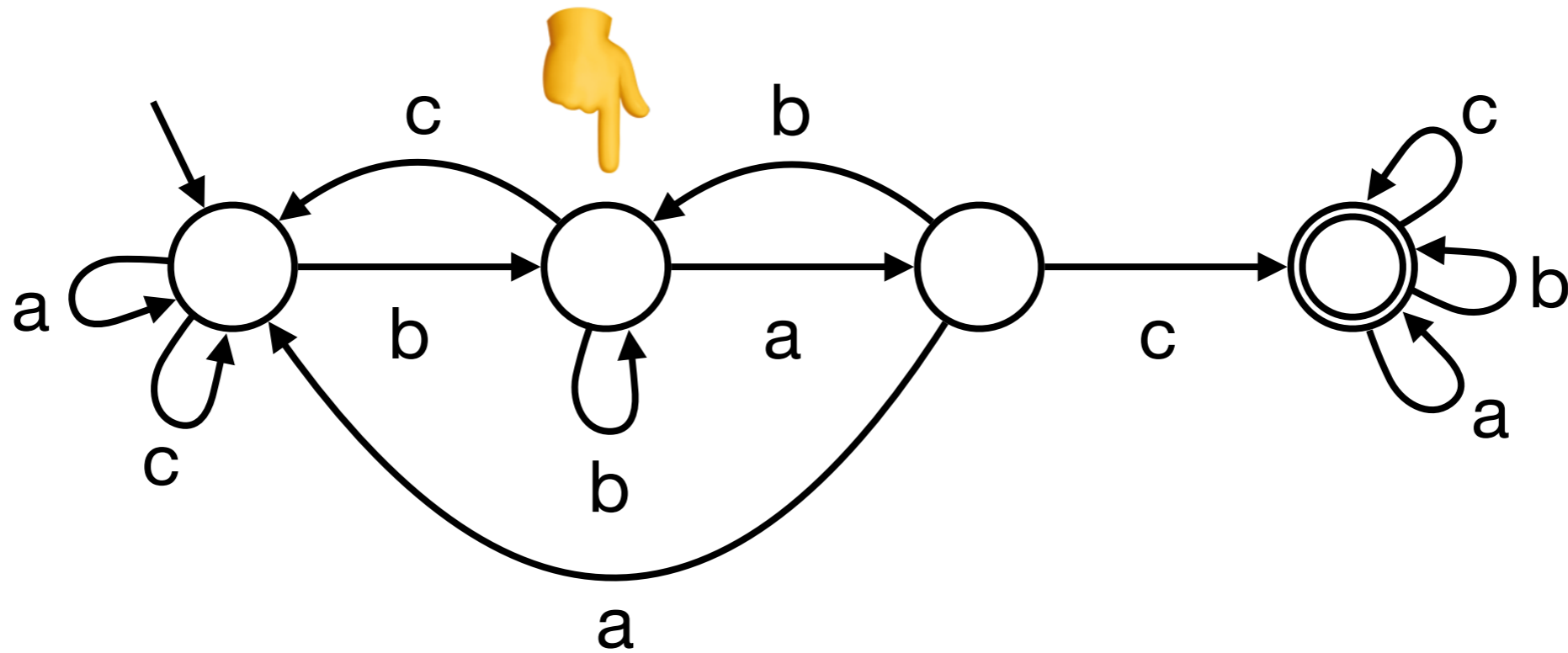
abaca<sup>3</sup>a

# Exemples de recherche



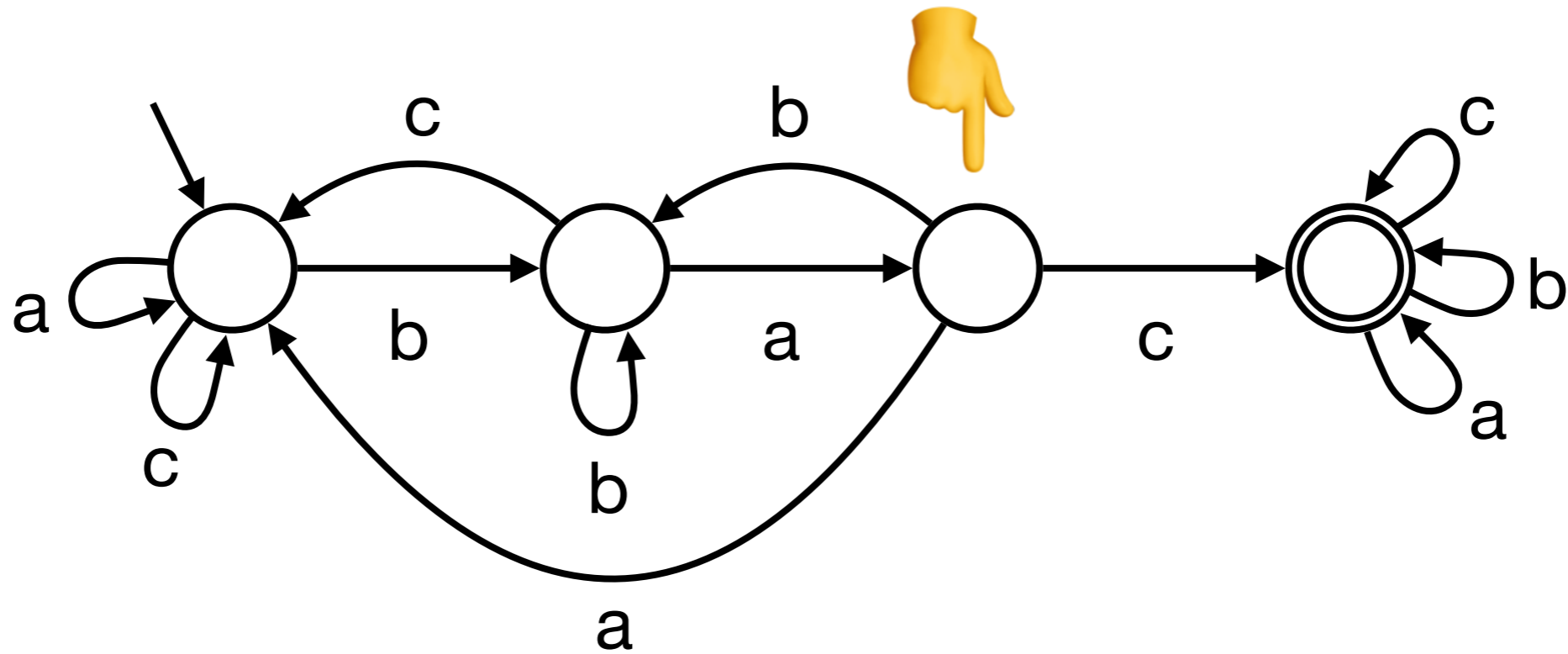
**a**bacaaa

# Exemples de recherche



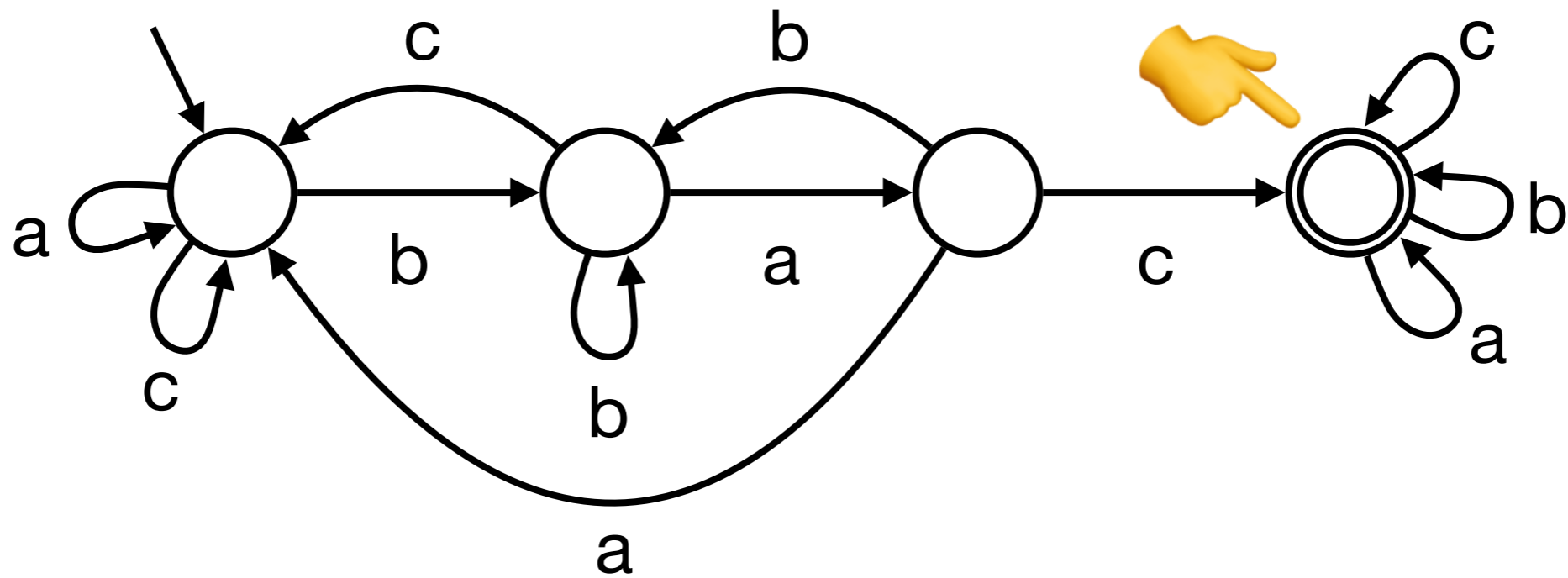
abaca<sup>a</sup>aa

# Exemples de recherche



abacaaa

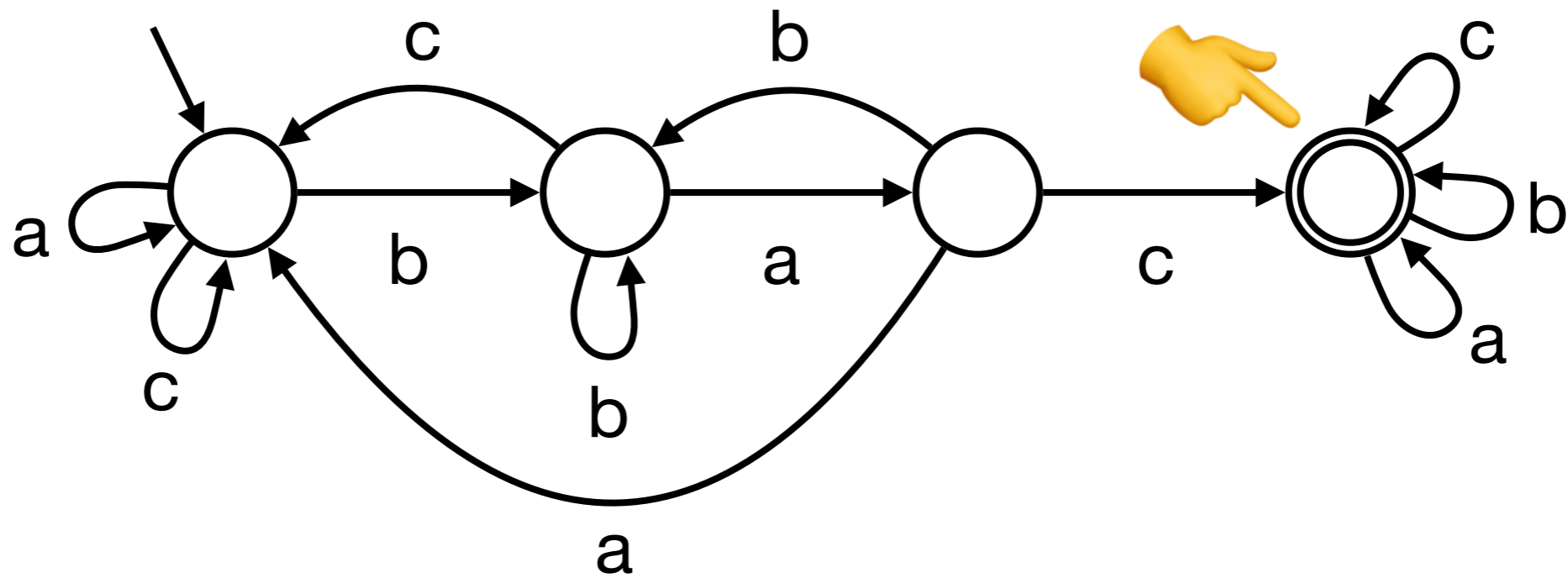
# Exemples de recherche



abacaaa

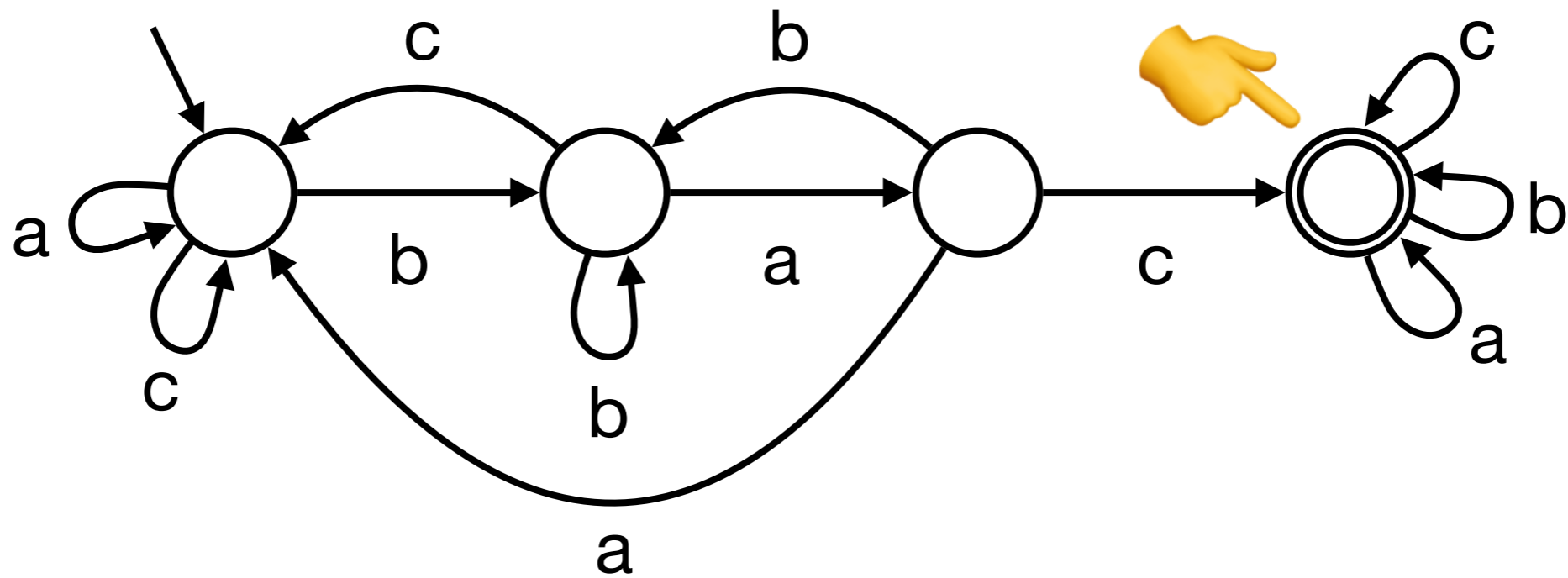


# Exemples de recherche



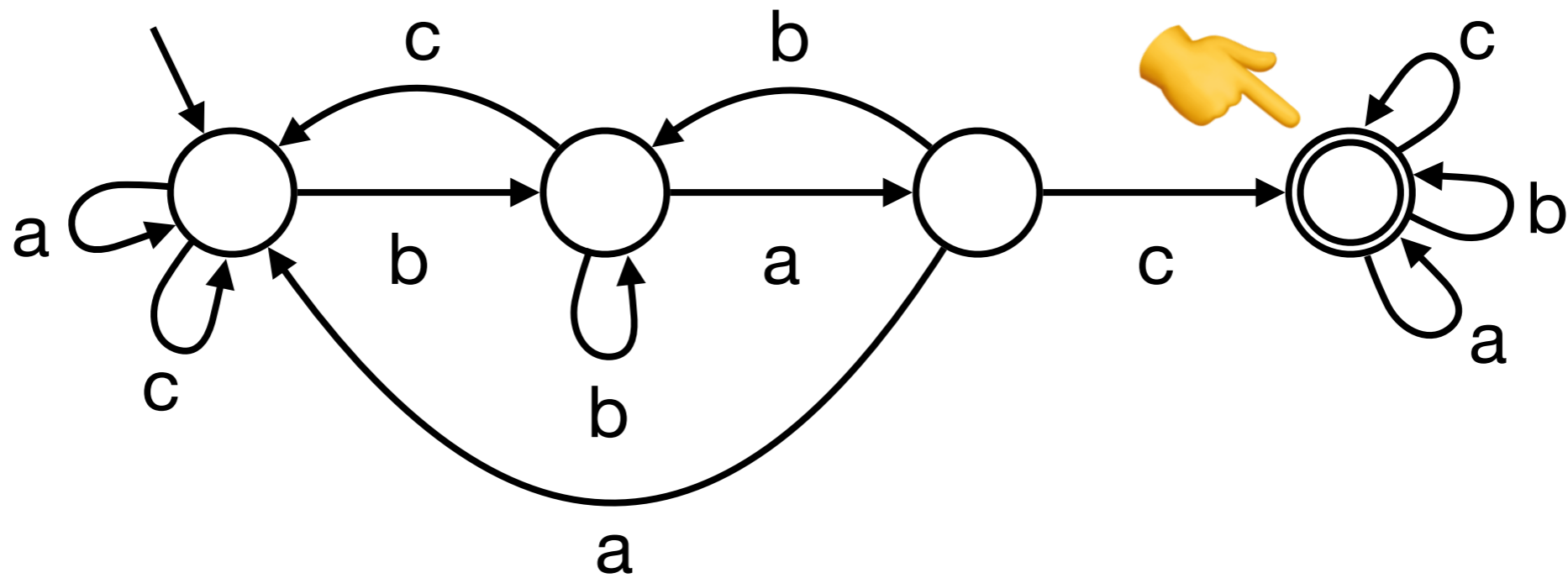
abaca~~aa~~

# Exemples de recherche



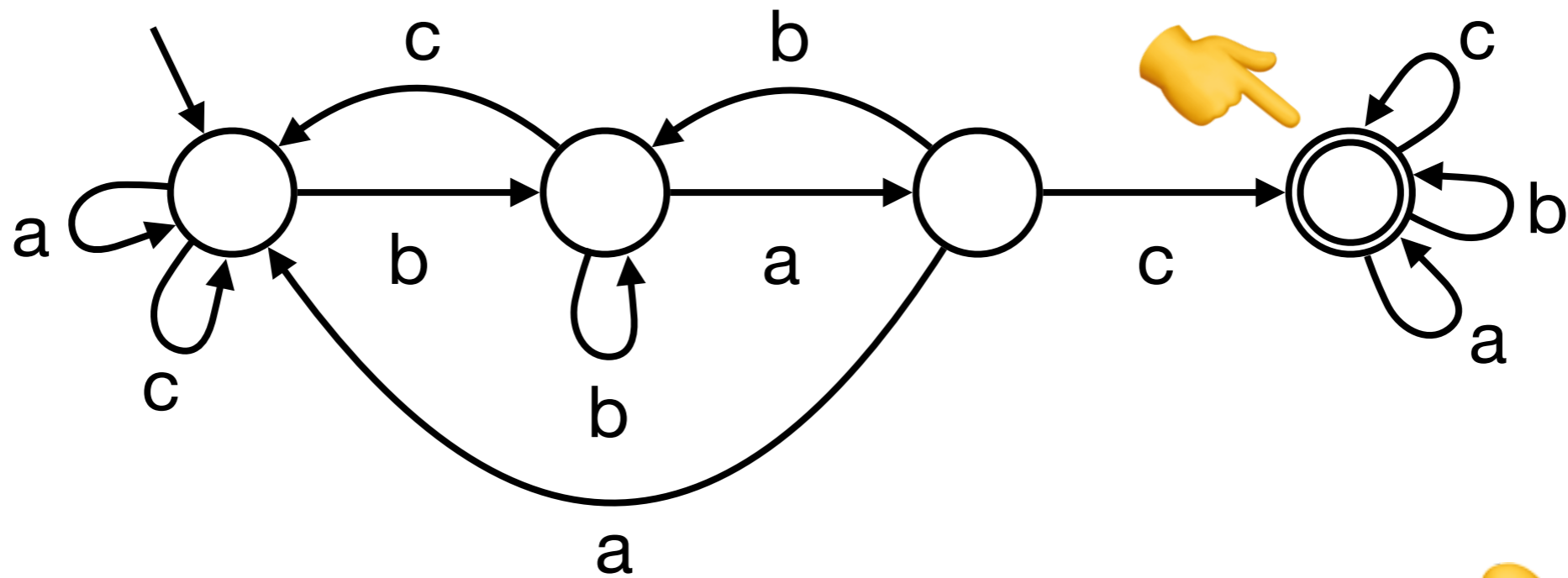
abacaaa

# Exemples de recherche



abacaaa

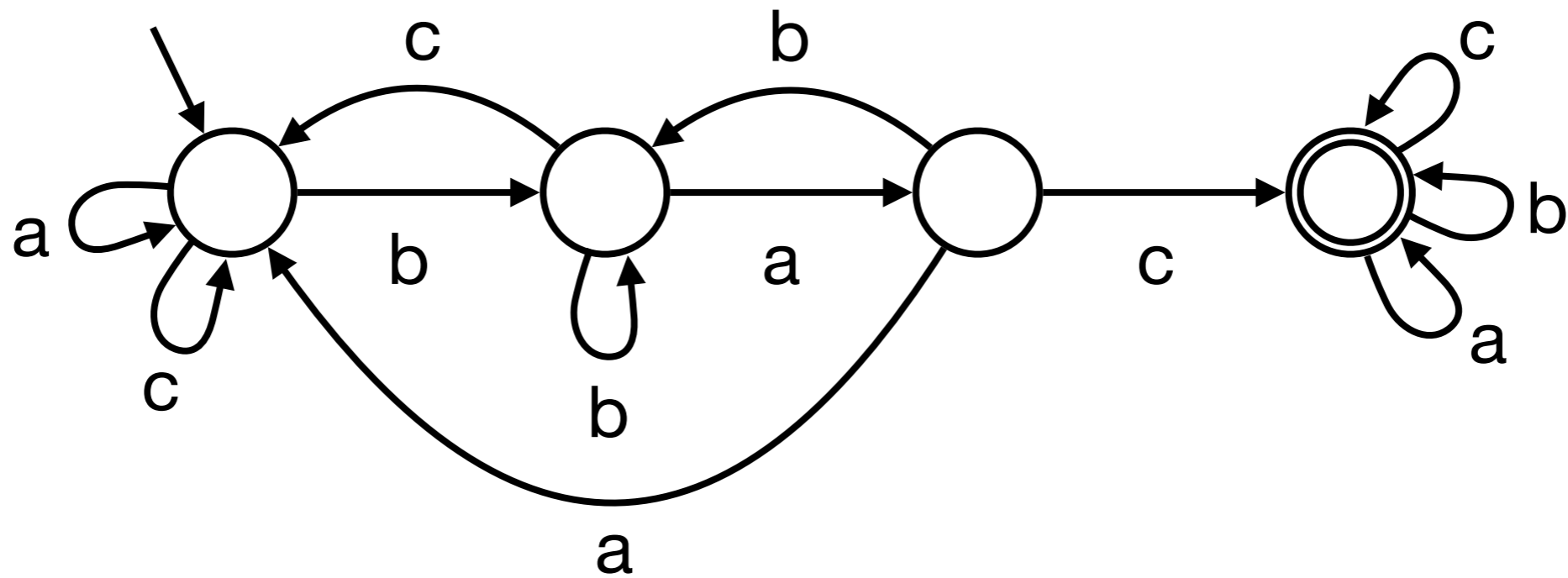
# Exemples de recherche



abacaaa

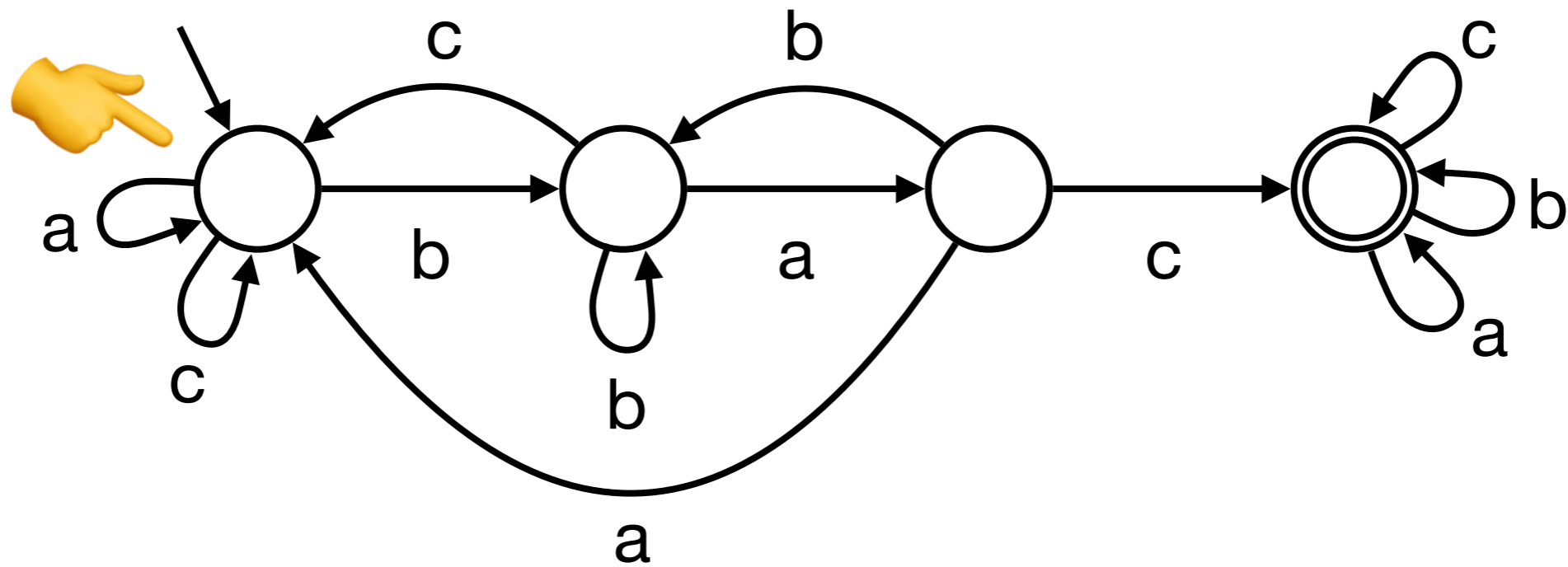


# Exemples de recherche



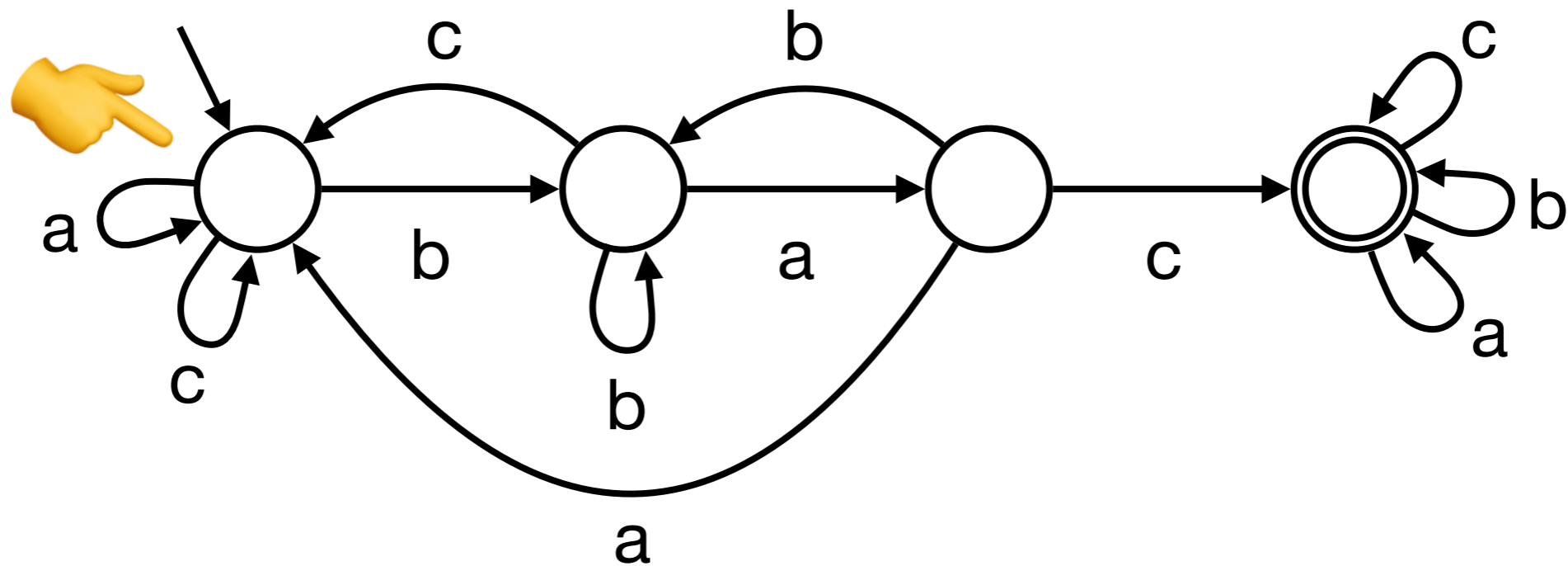
ababaac

# Exemples de recherche



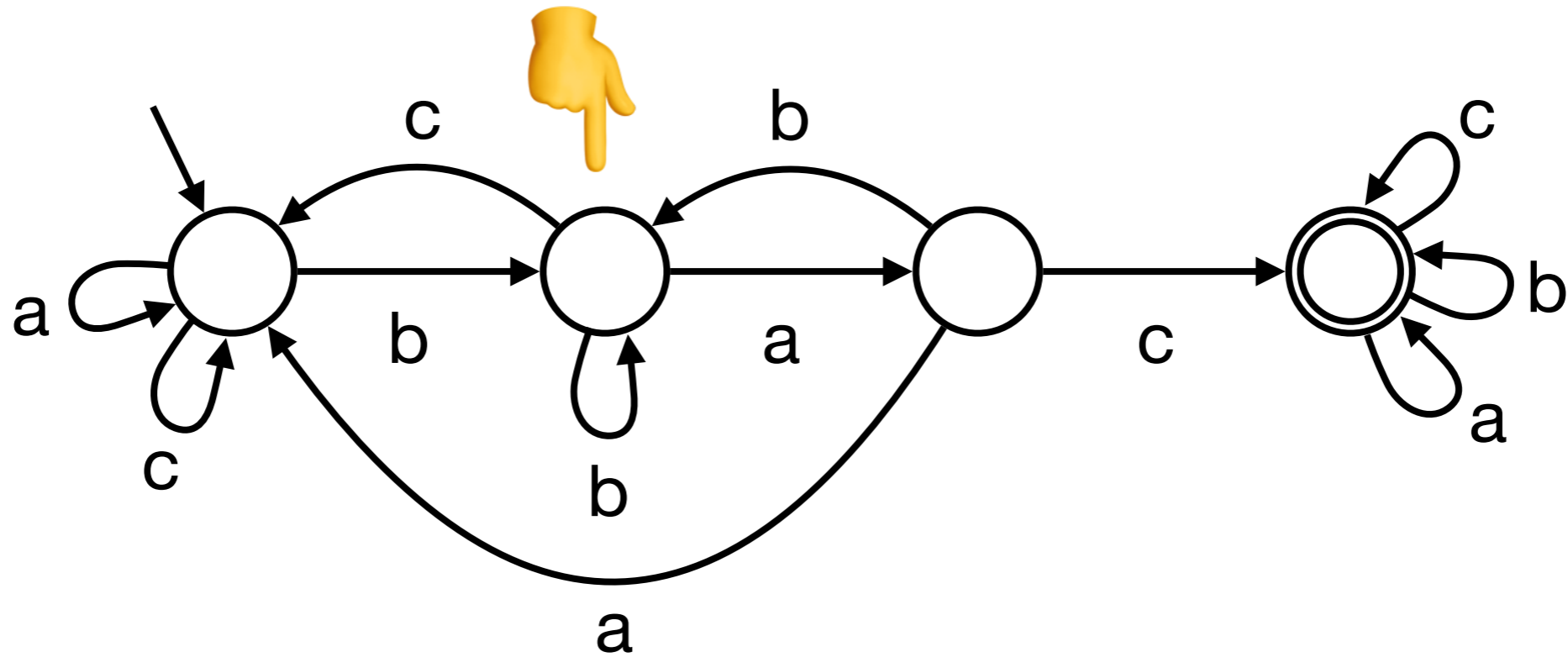
ababaac

# Exemples de recherche



**a**babaac

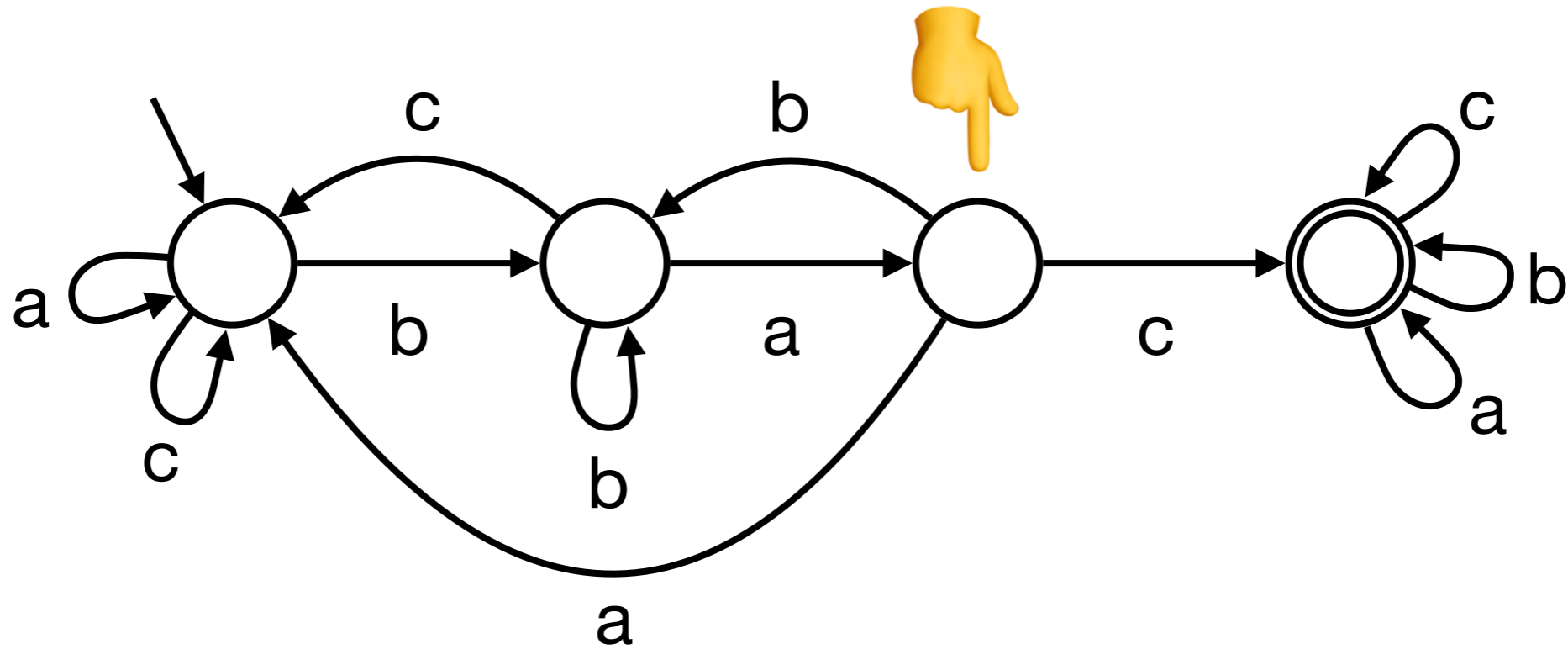
# Exemples de recherche



ababaac

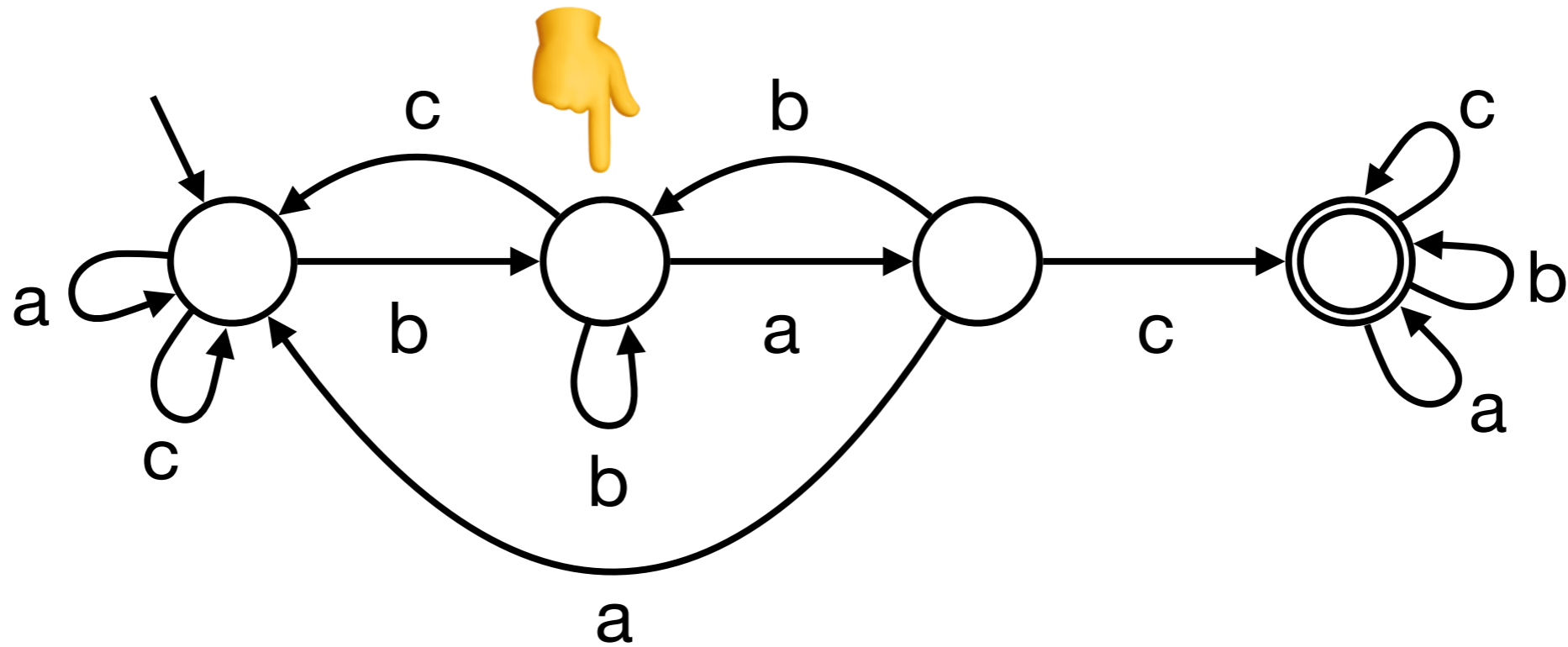


# Exemples de recherche



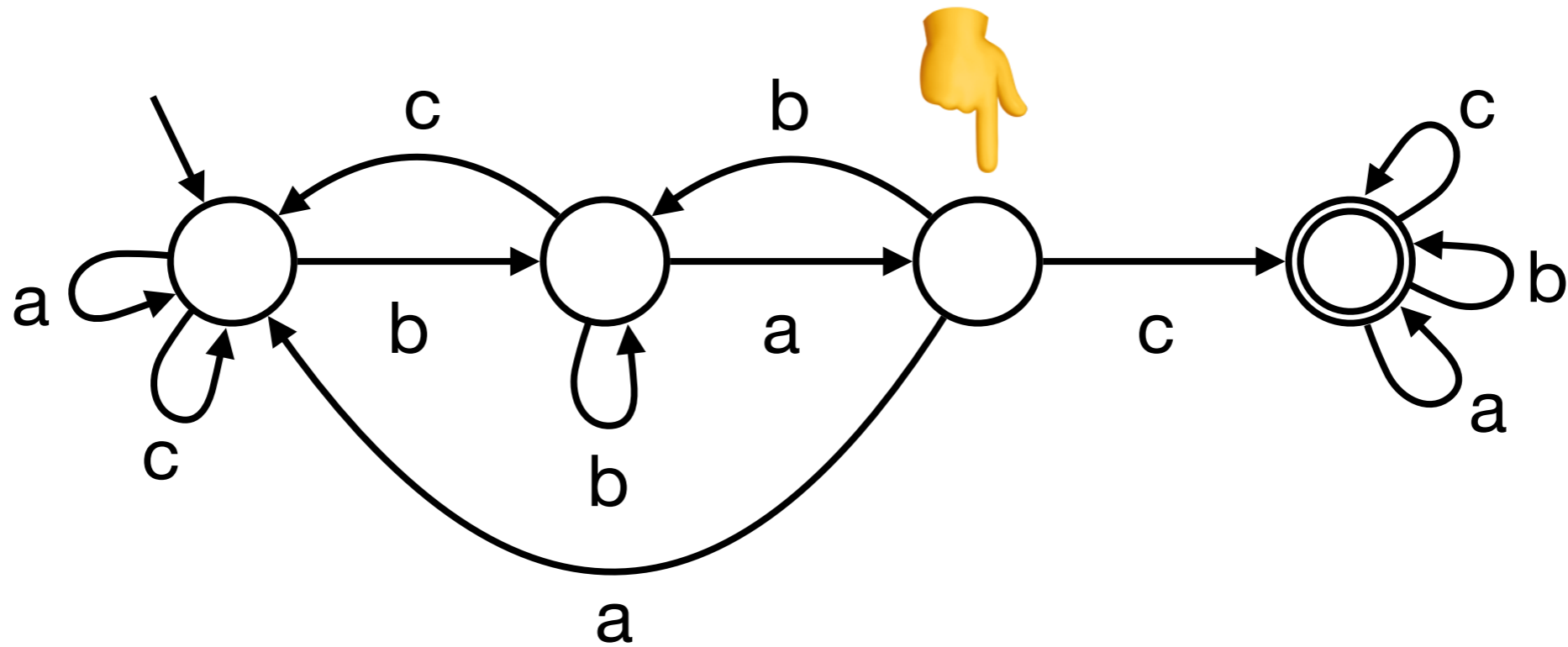
ababaac

# Exemples de recherche



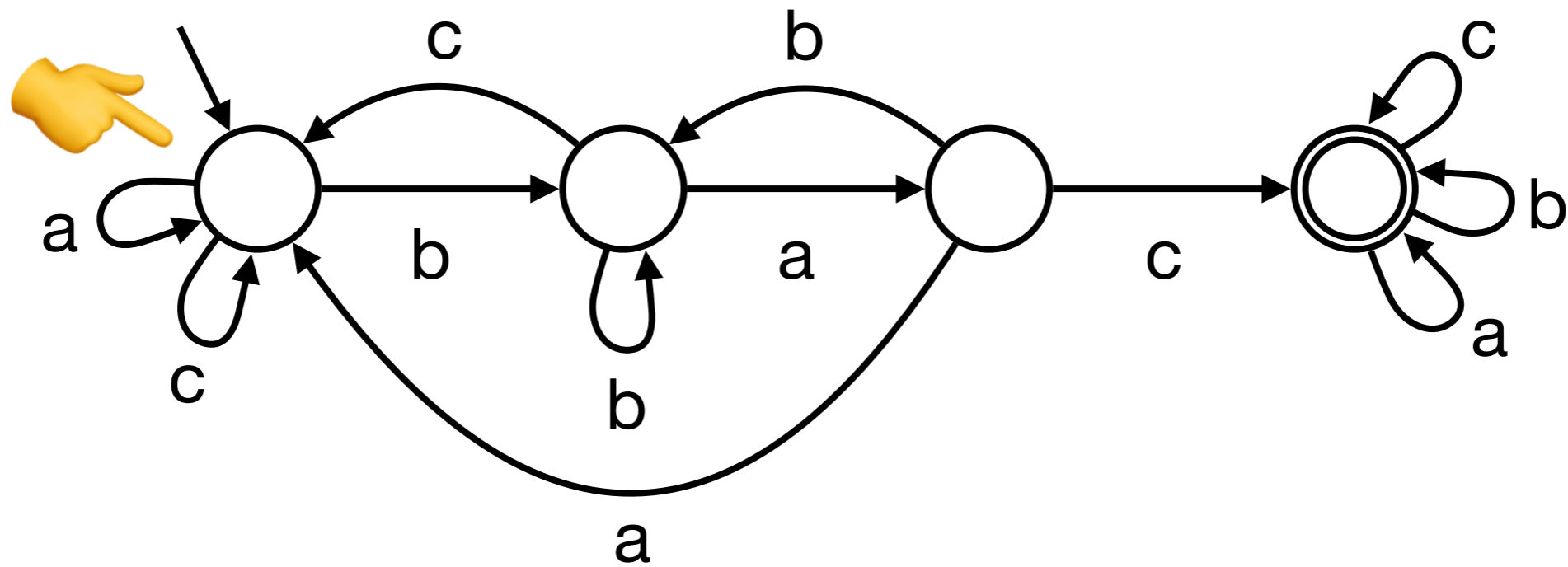
ababaac

# Exemples de recherche



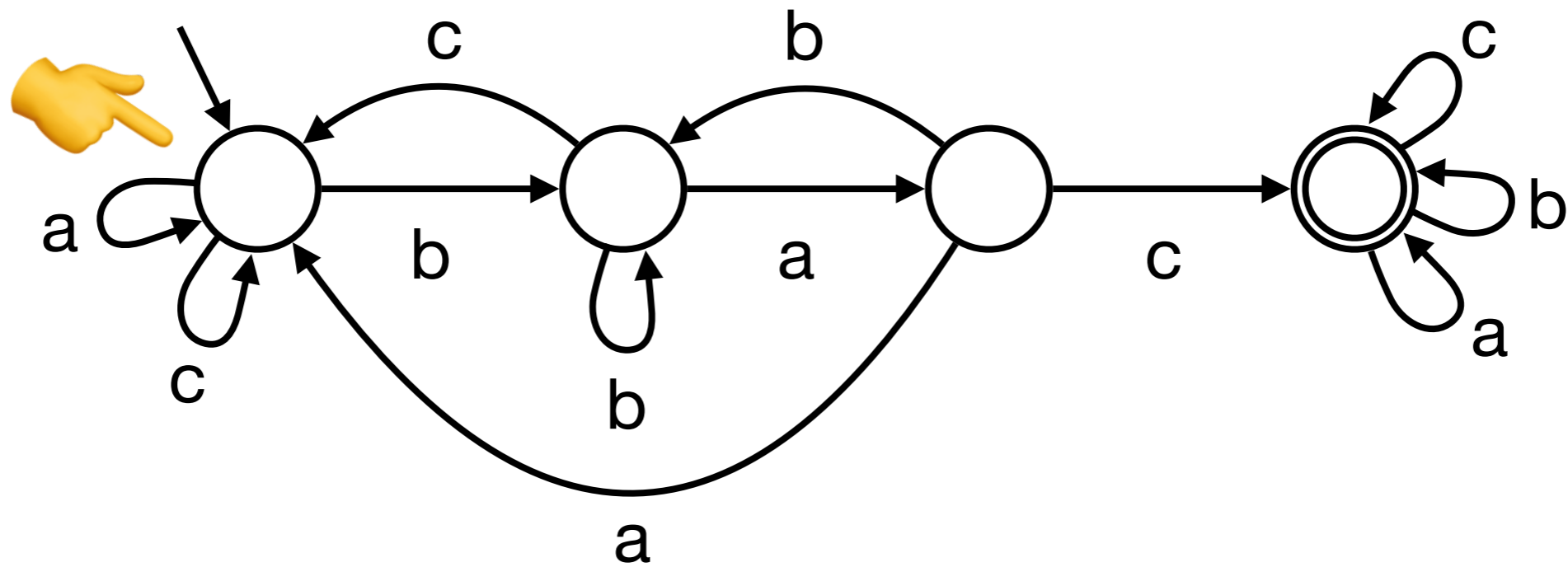
ababaac

# Exemples de recherche



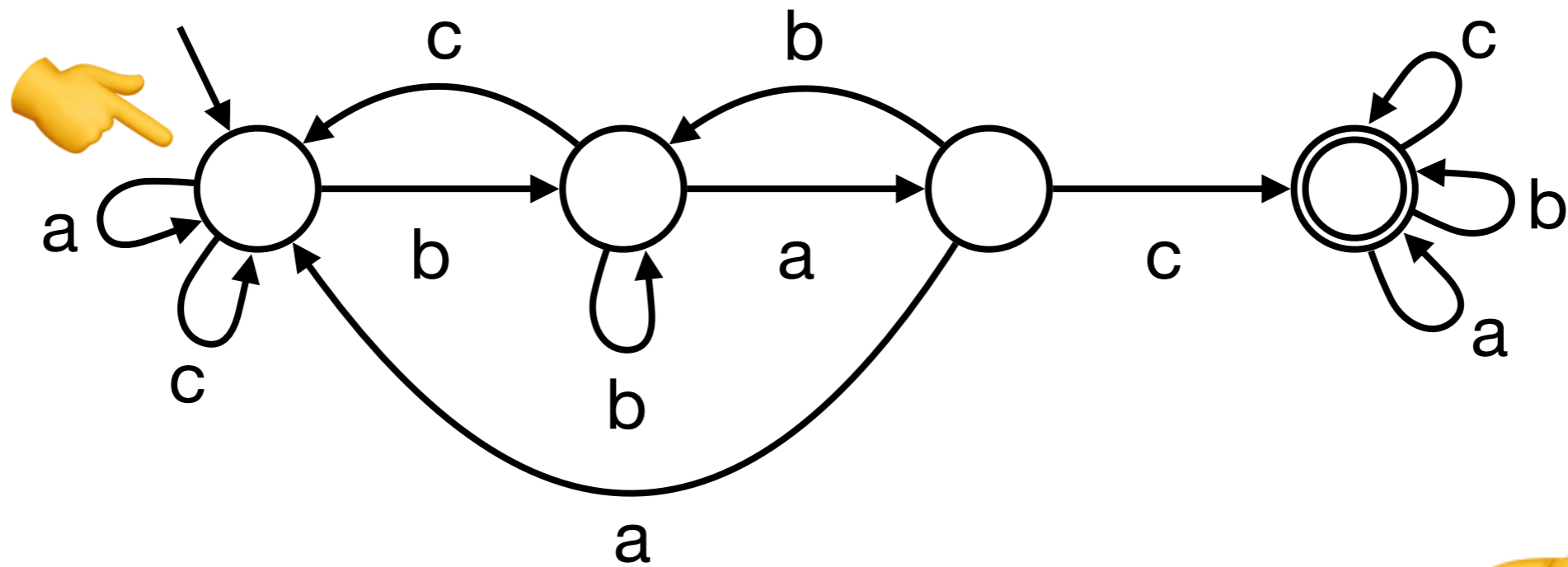
ababaac

# Exemples de recherche



ababaac

# Exemples de recherche

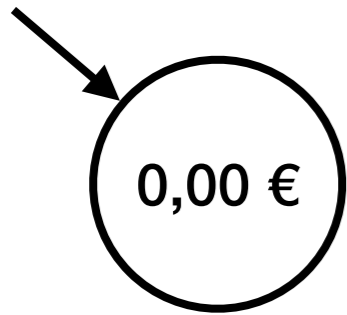


ababaac



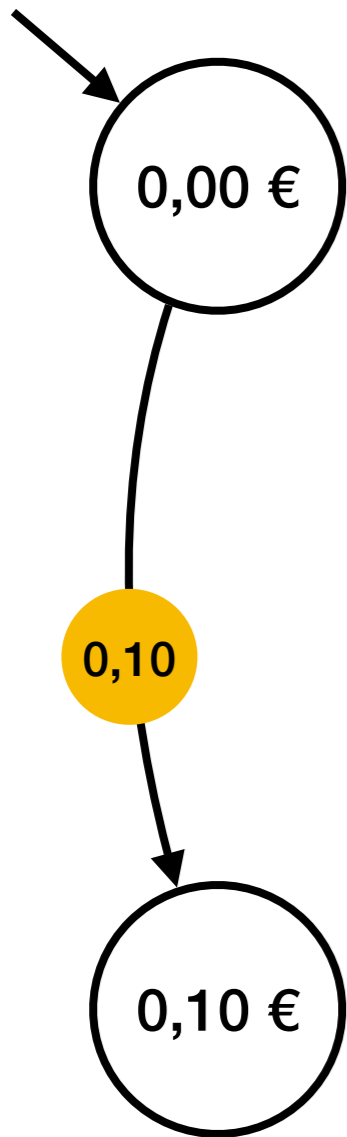
# Modélisation d'un distributeur de café

- Café à 0,30 €
- The à 0,50 €
- La machine n'accepte que les pièces de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut annuler à tout moment (et récupérer la monnaie)

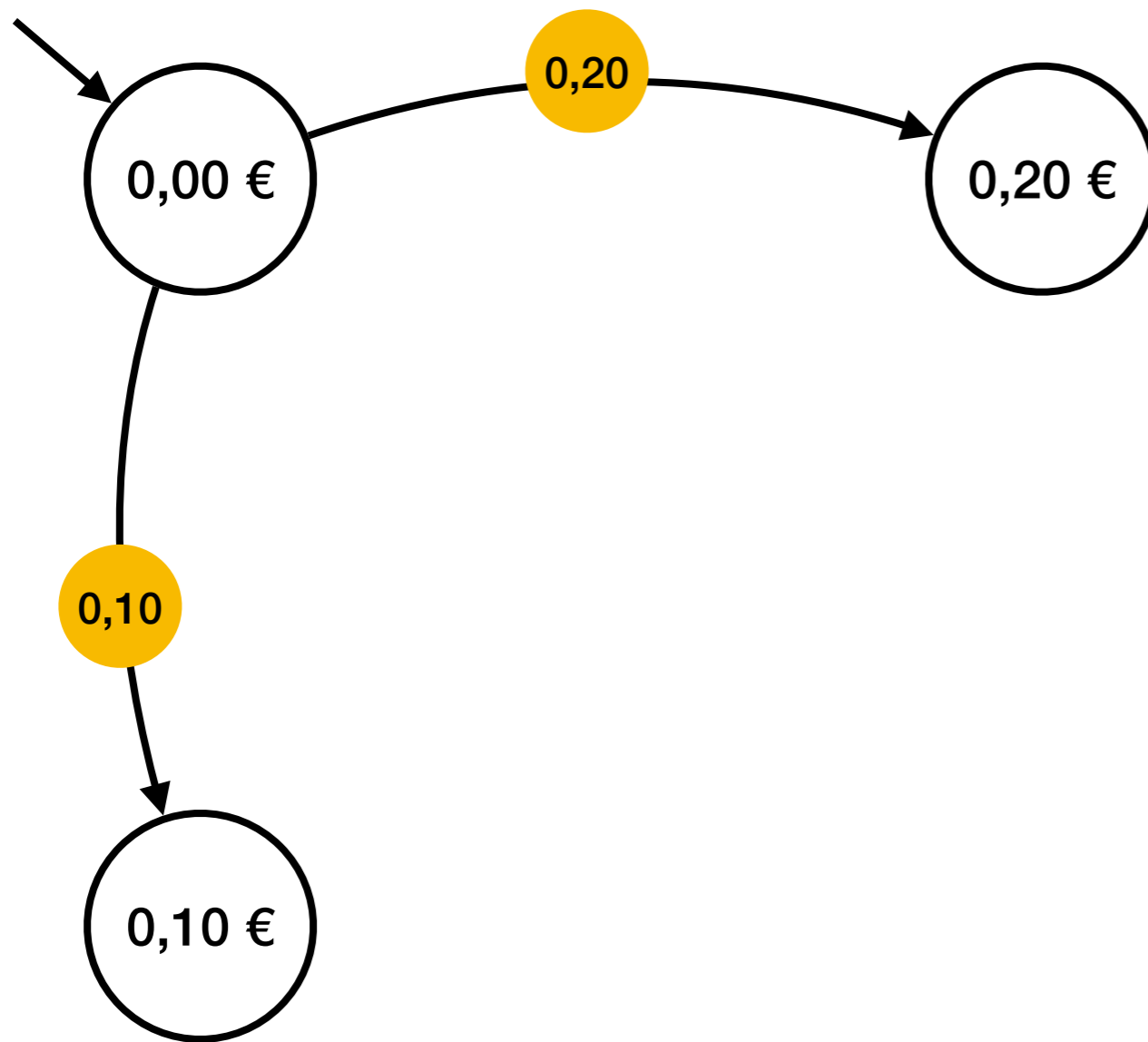


- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

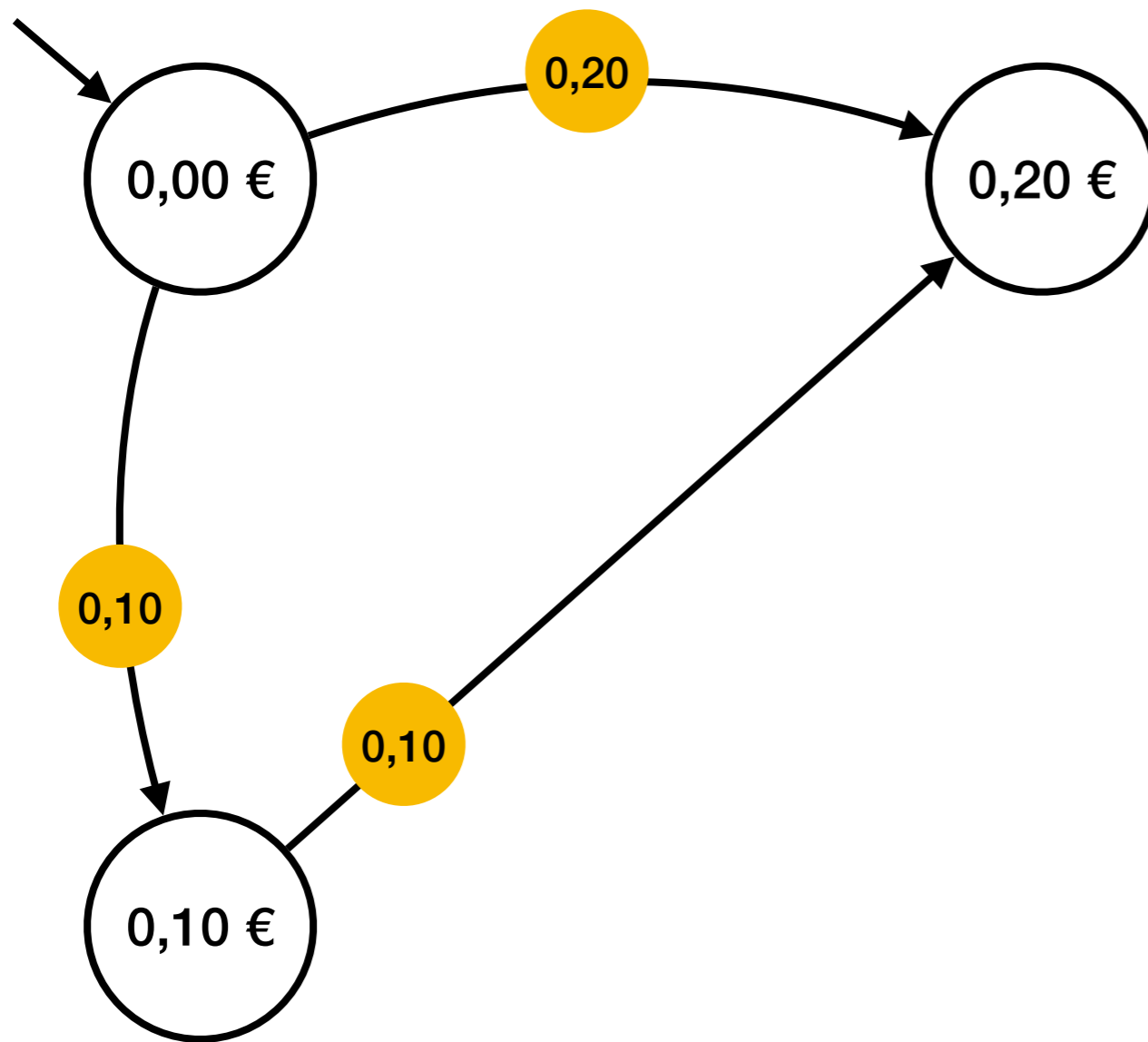




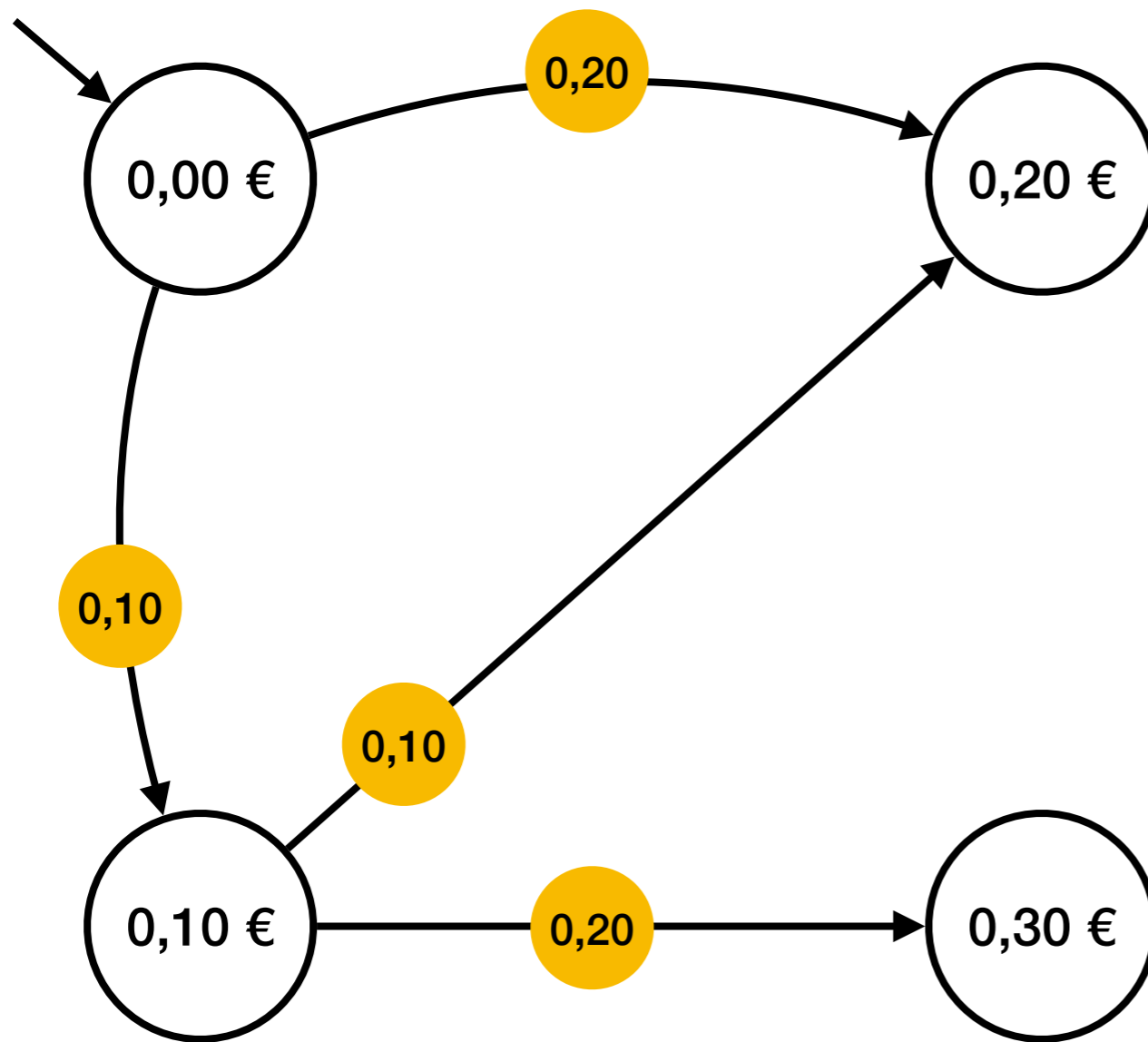
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



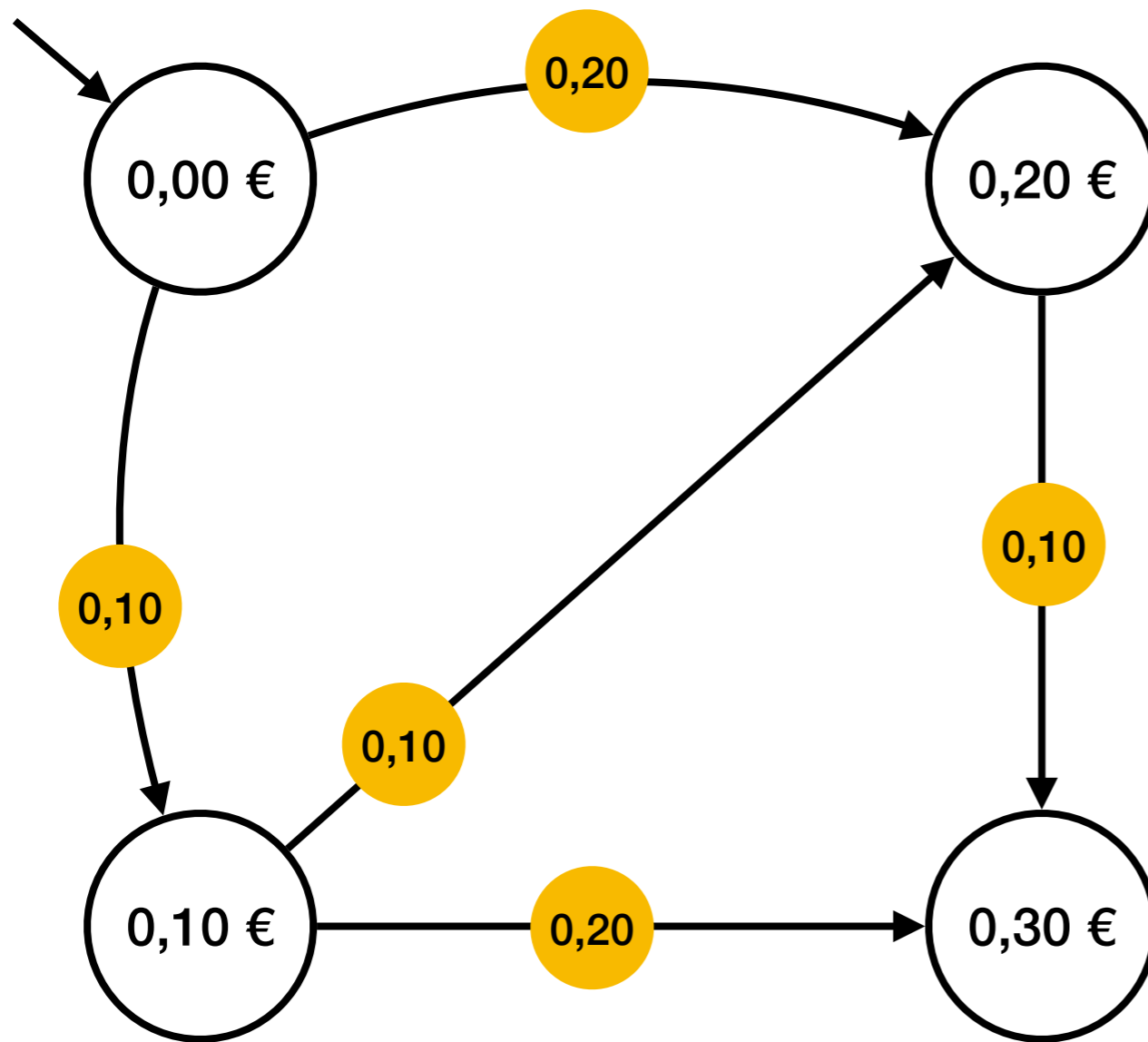
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



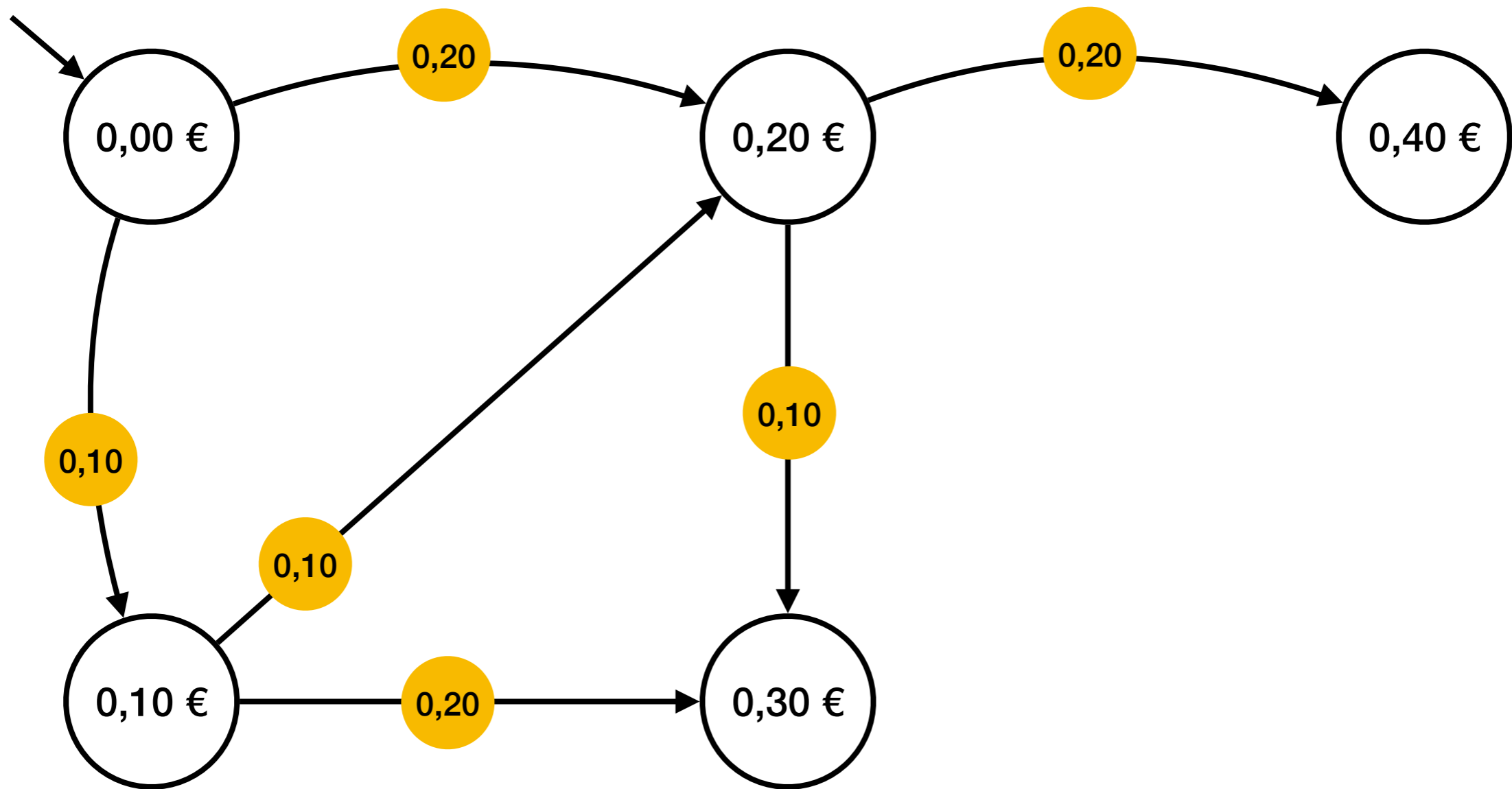
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



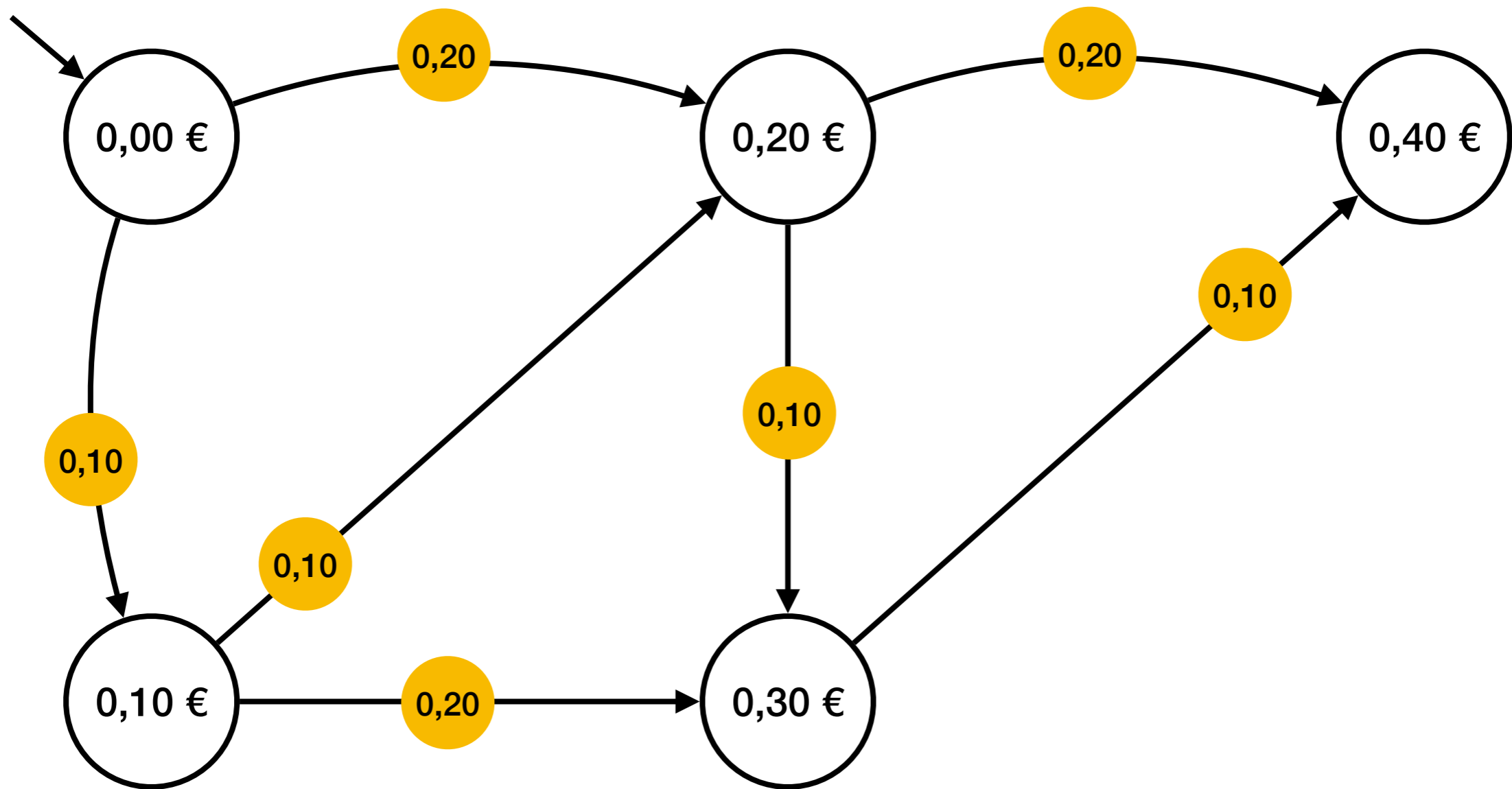
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



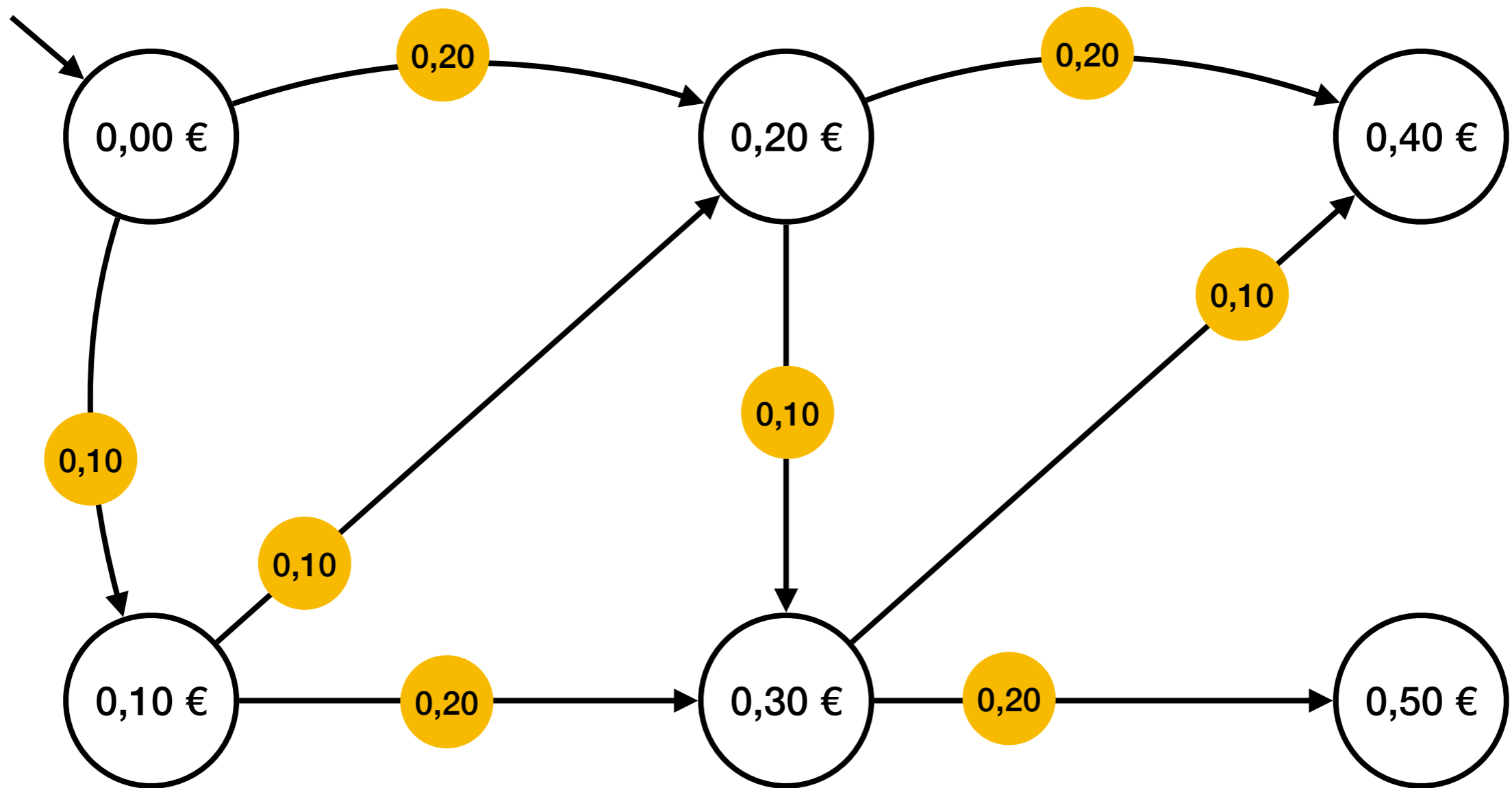
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

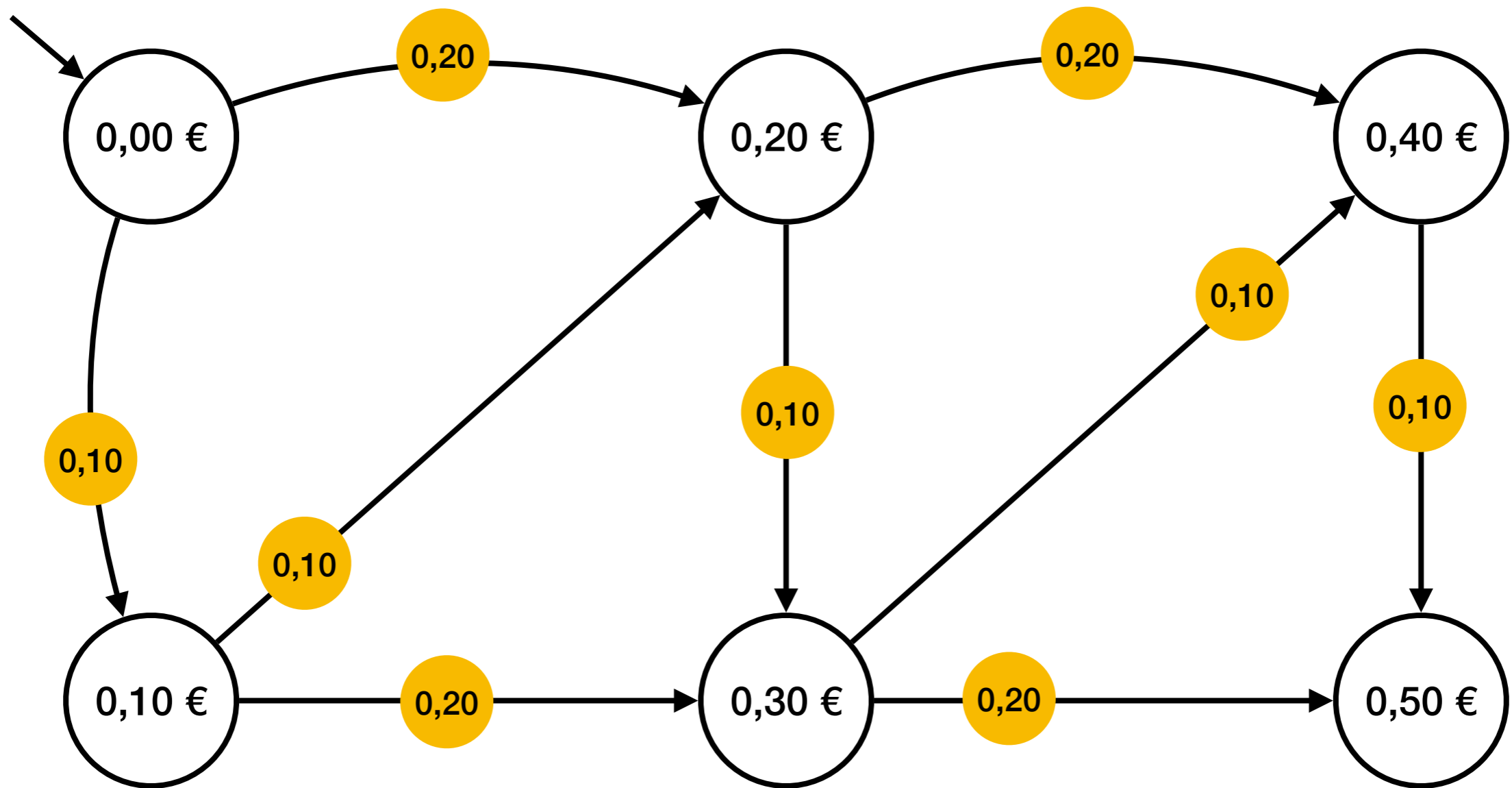


- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

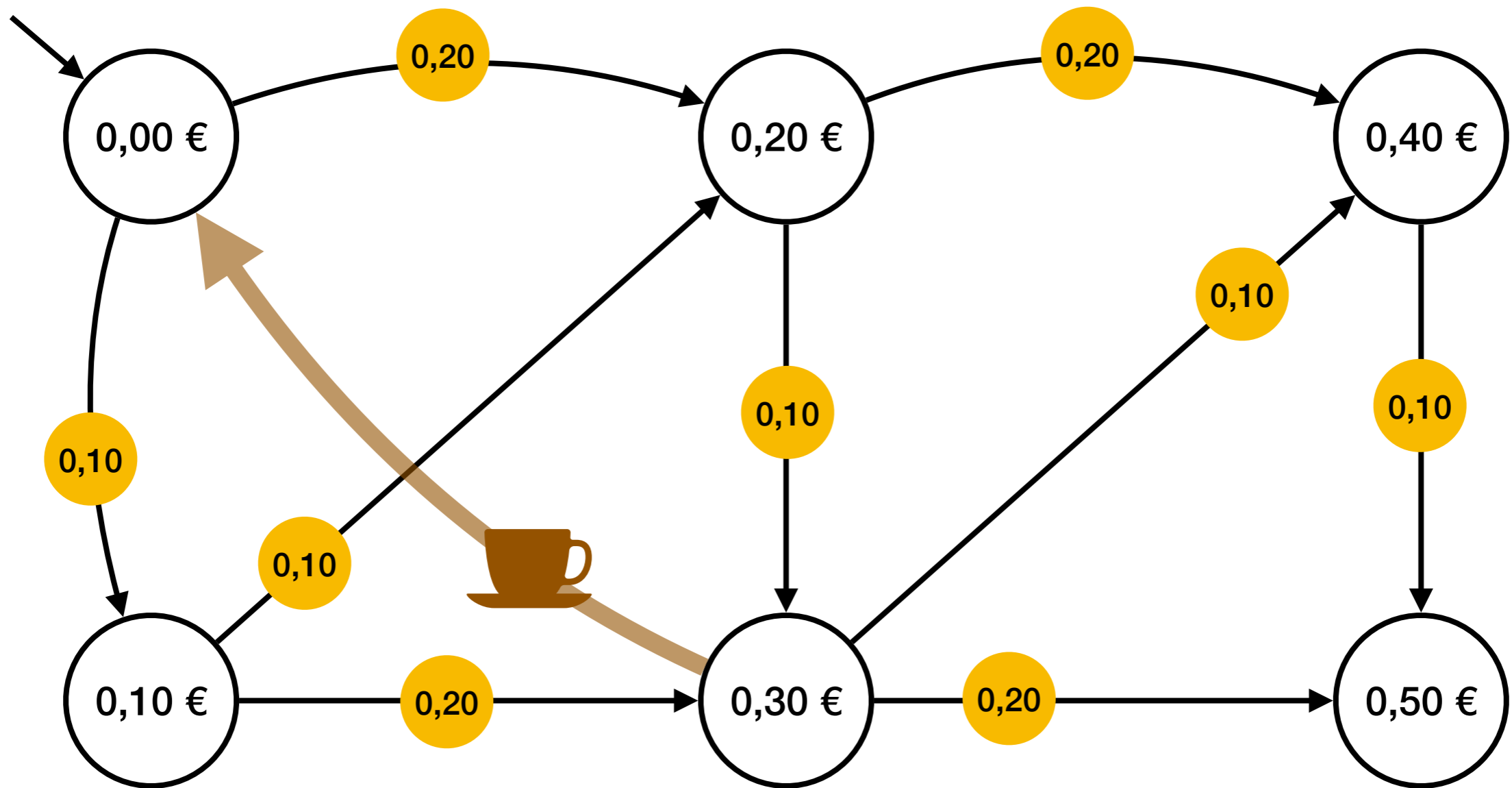


- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

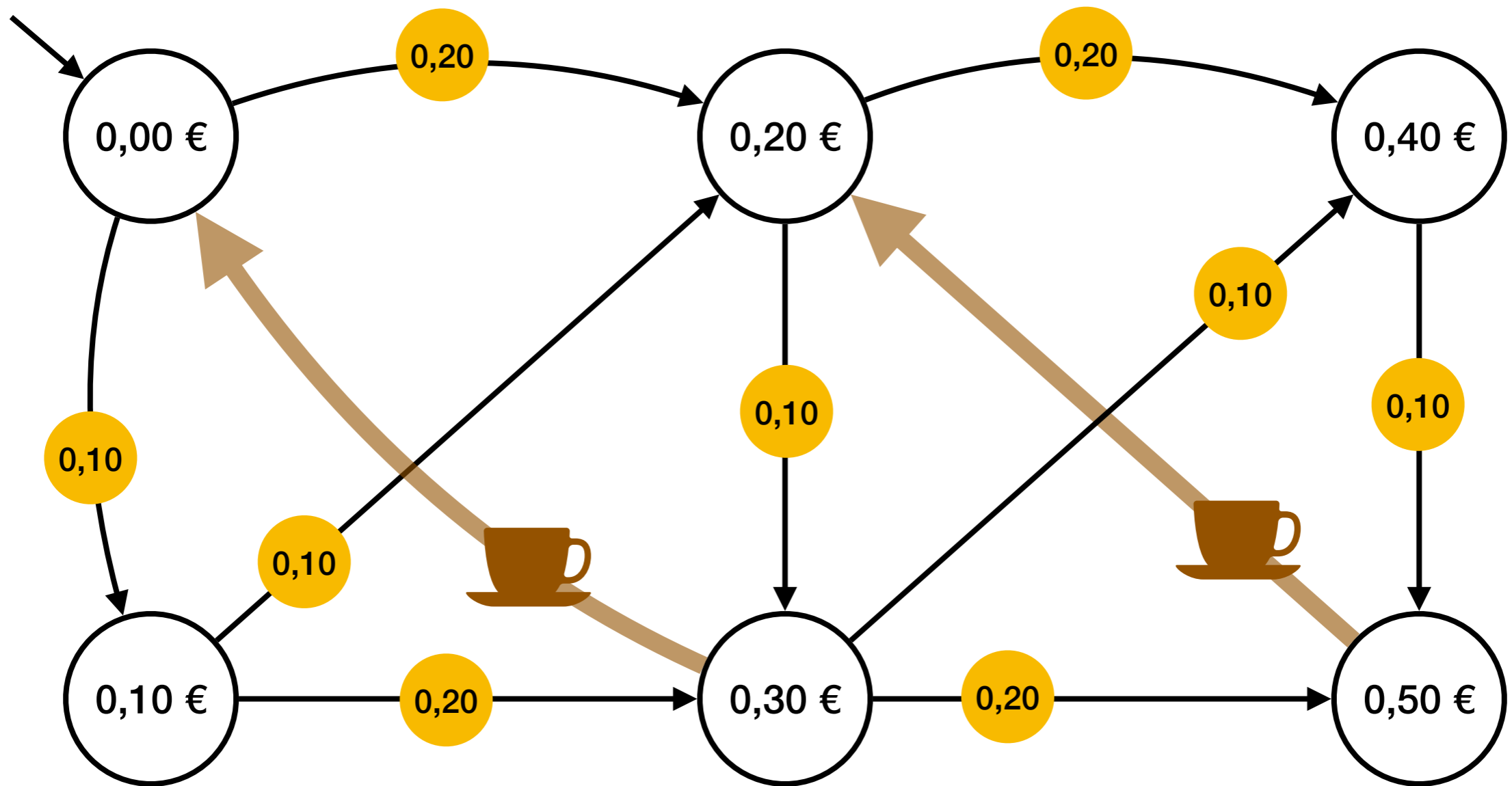




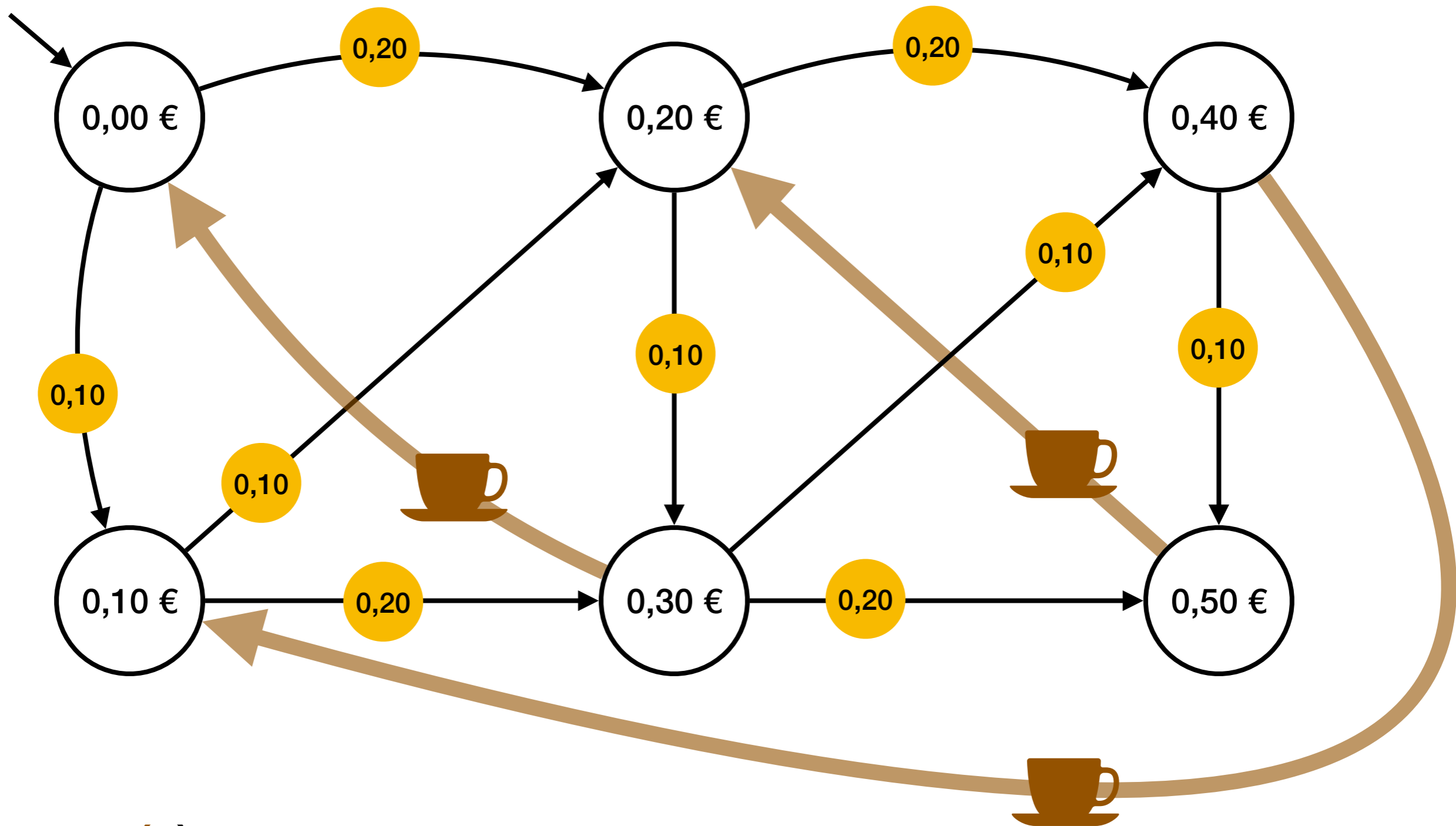
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



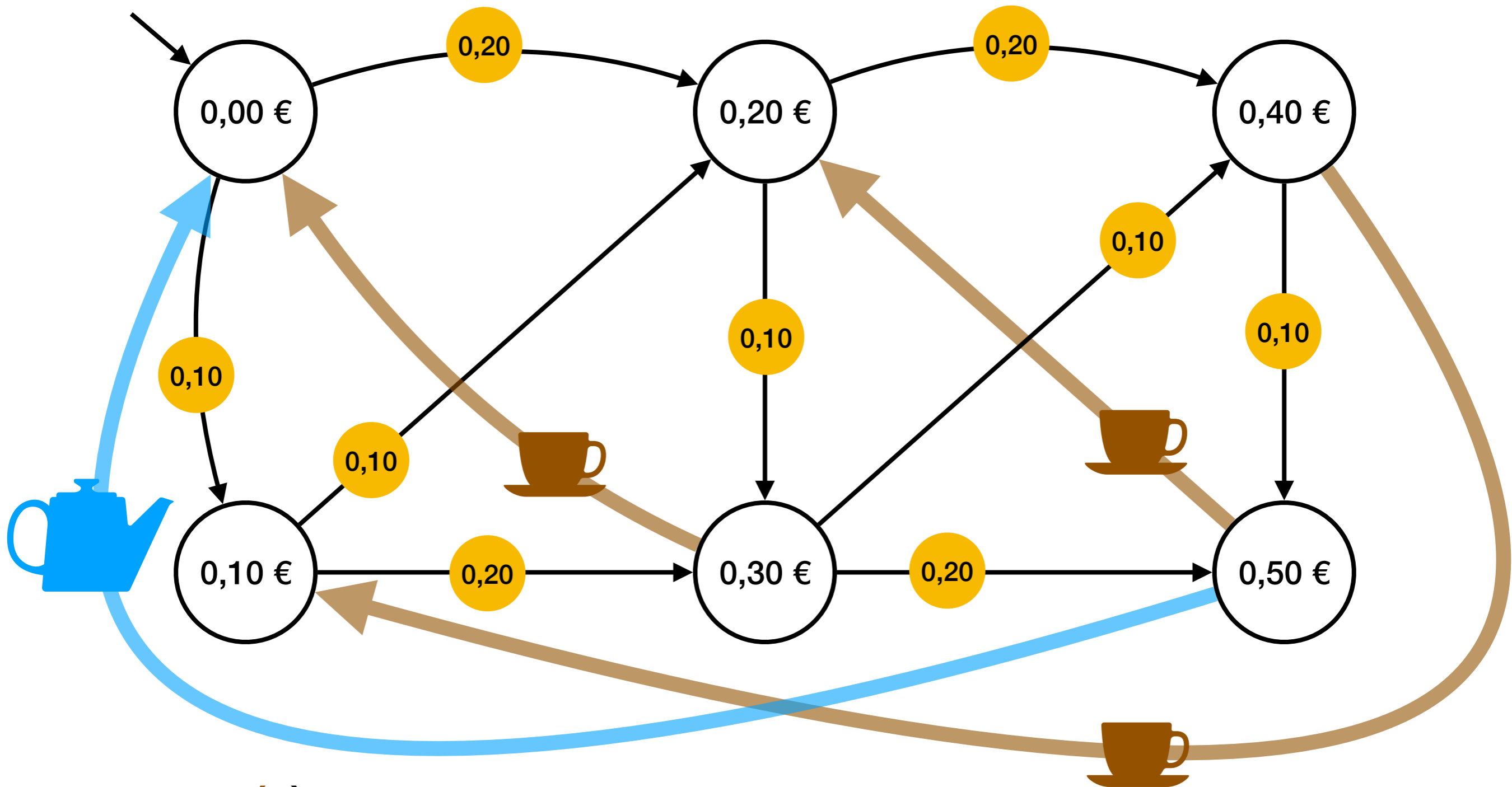
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



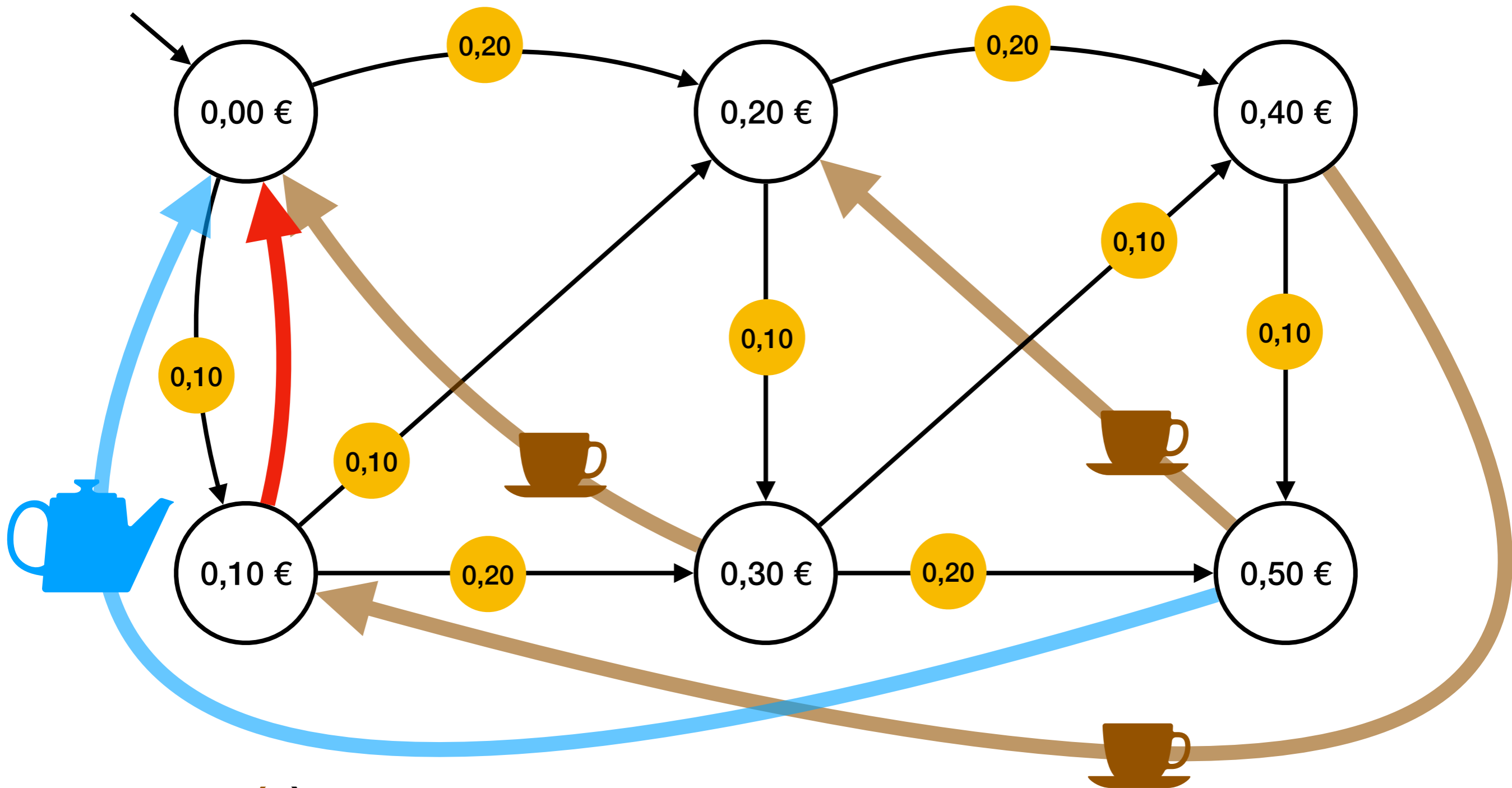
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



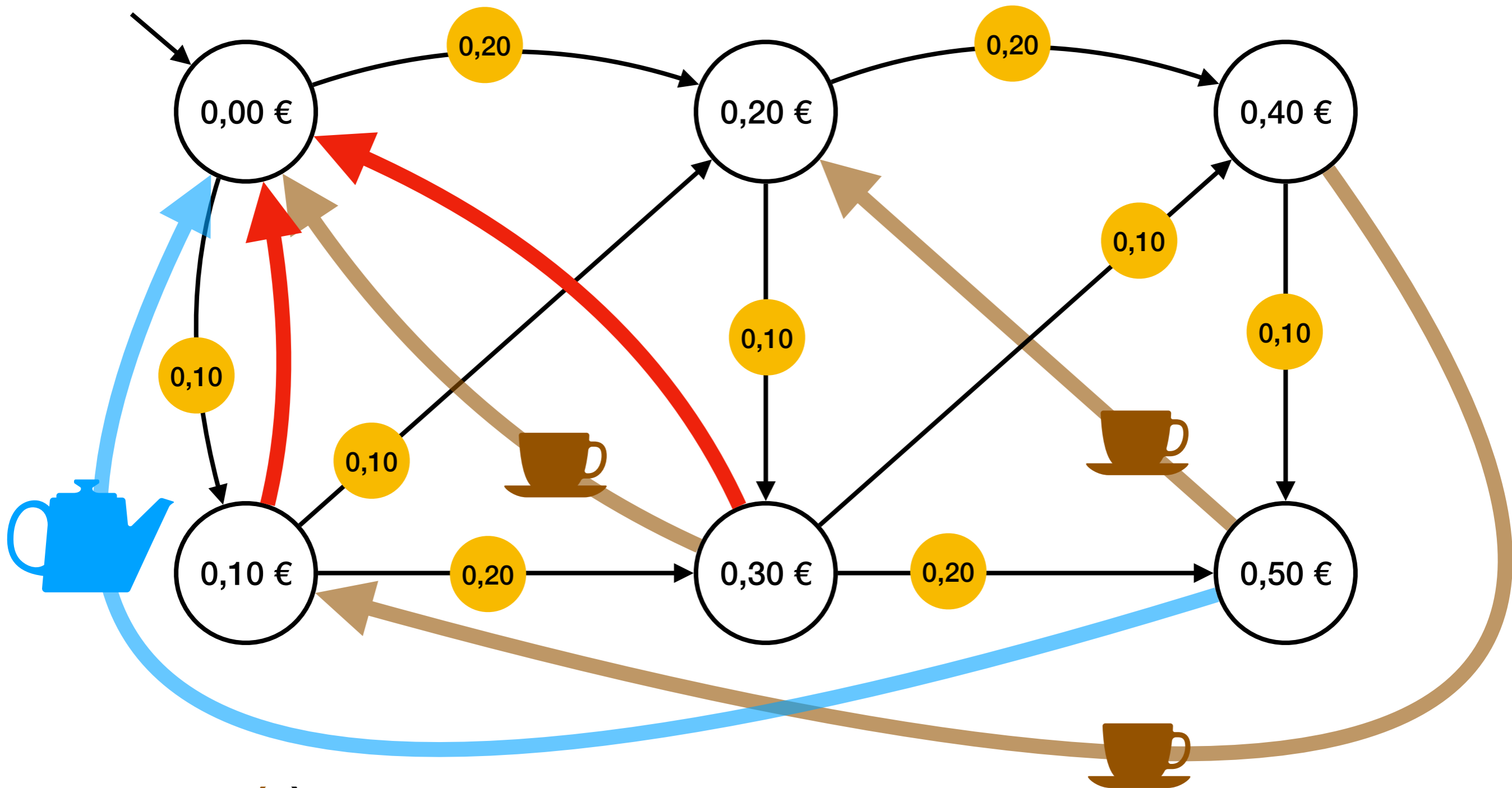
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



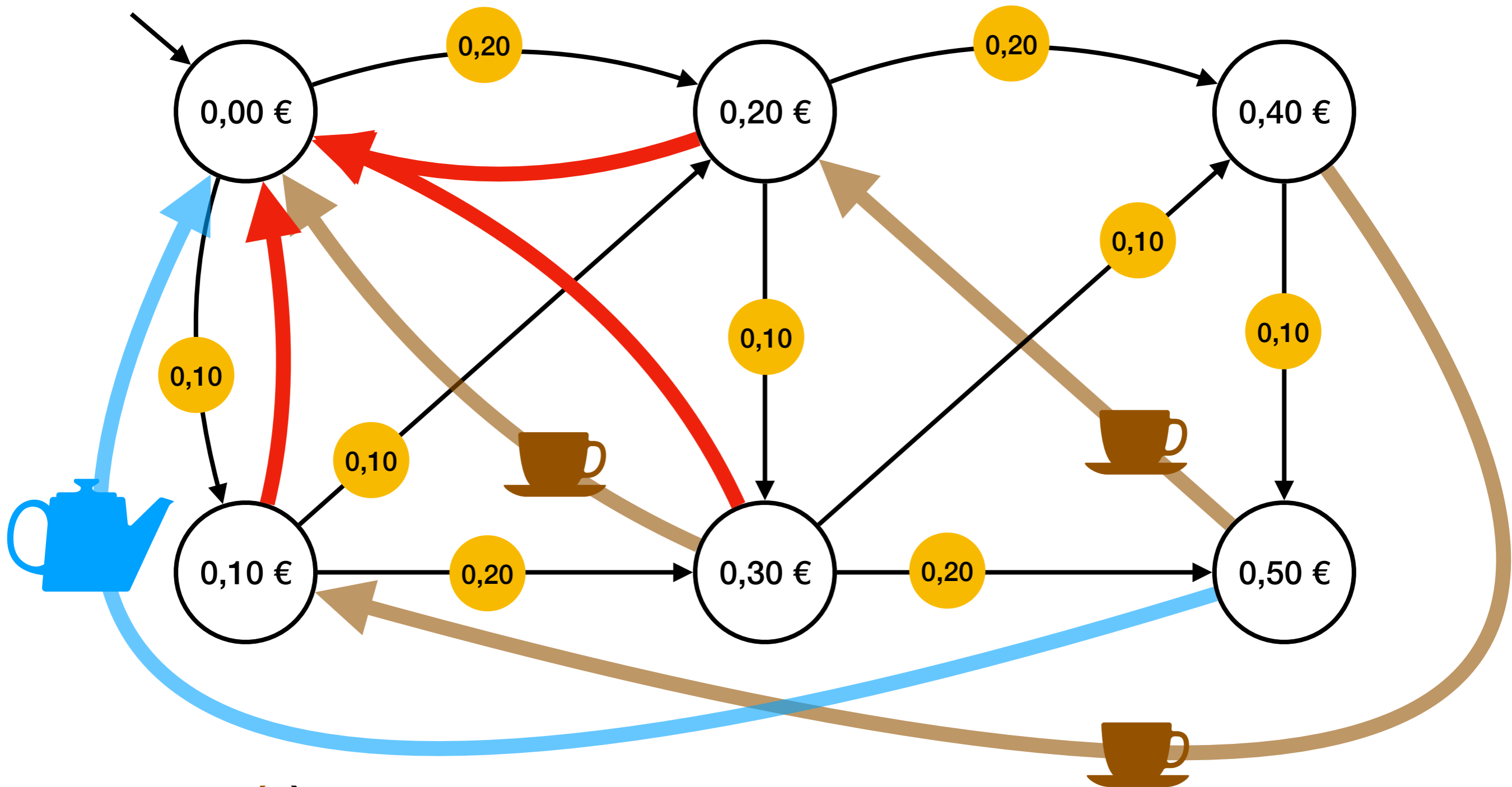
- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

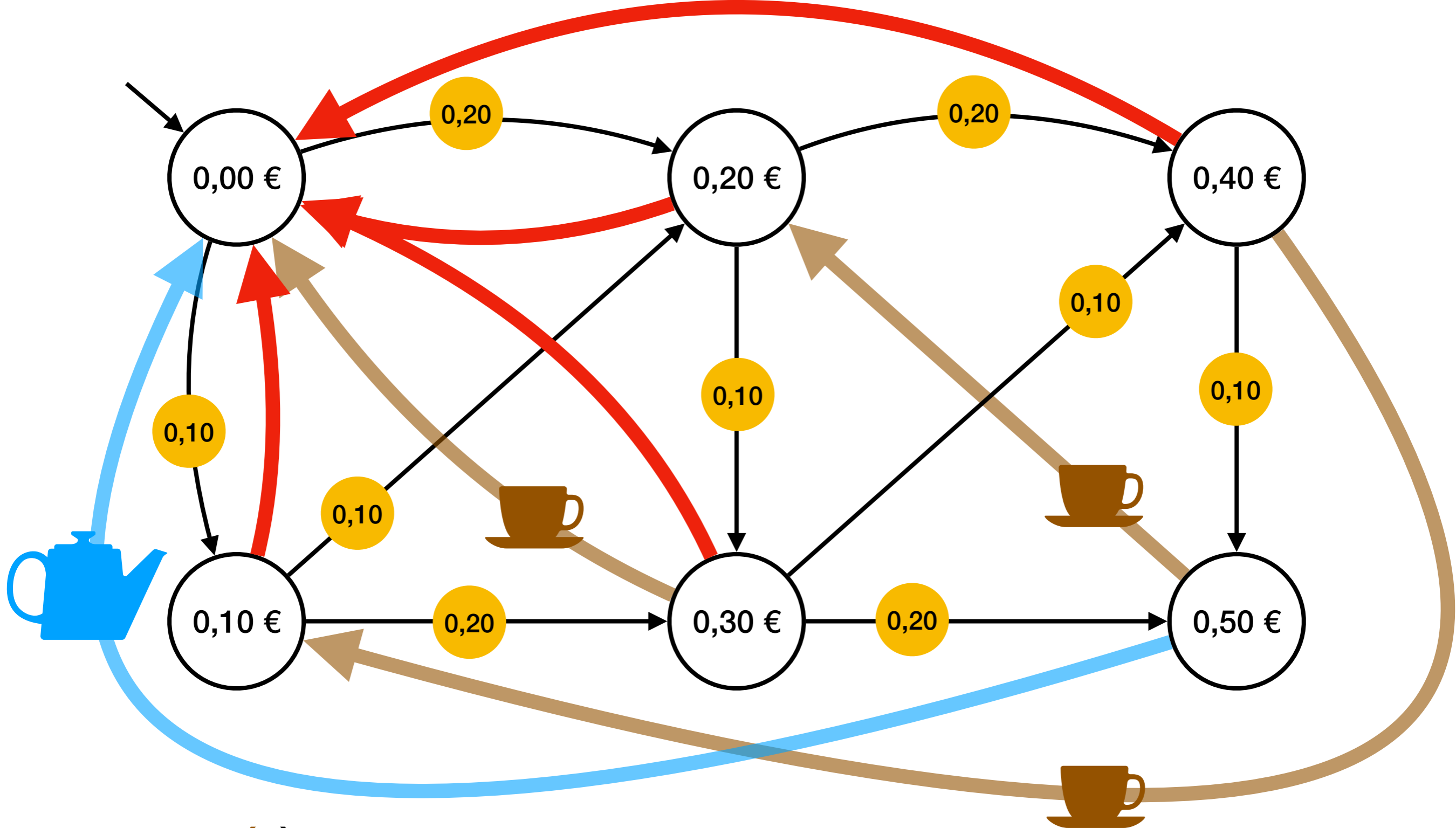


- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

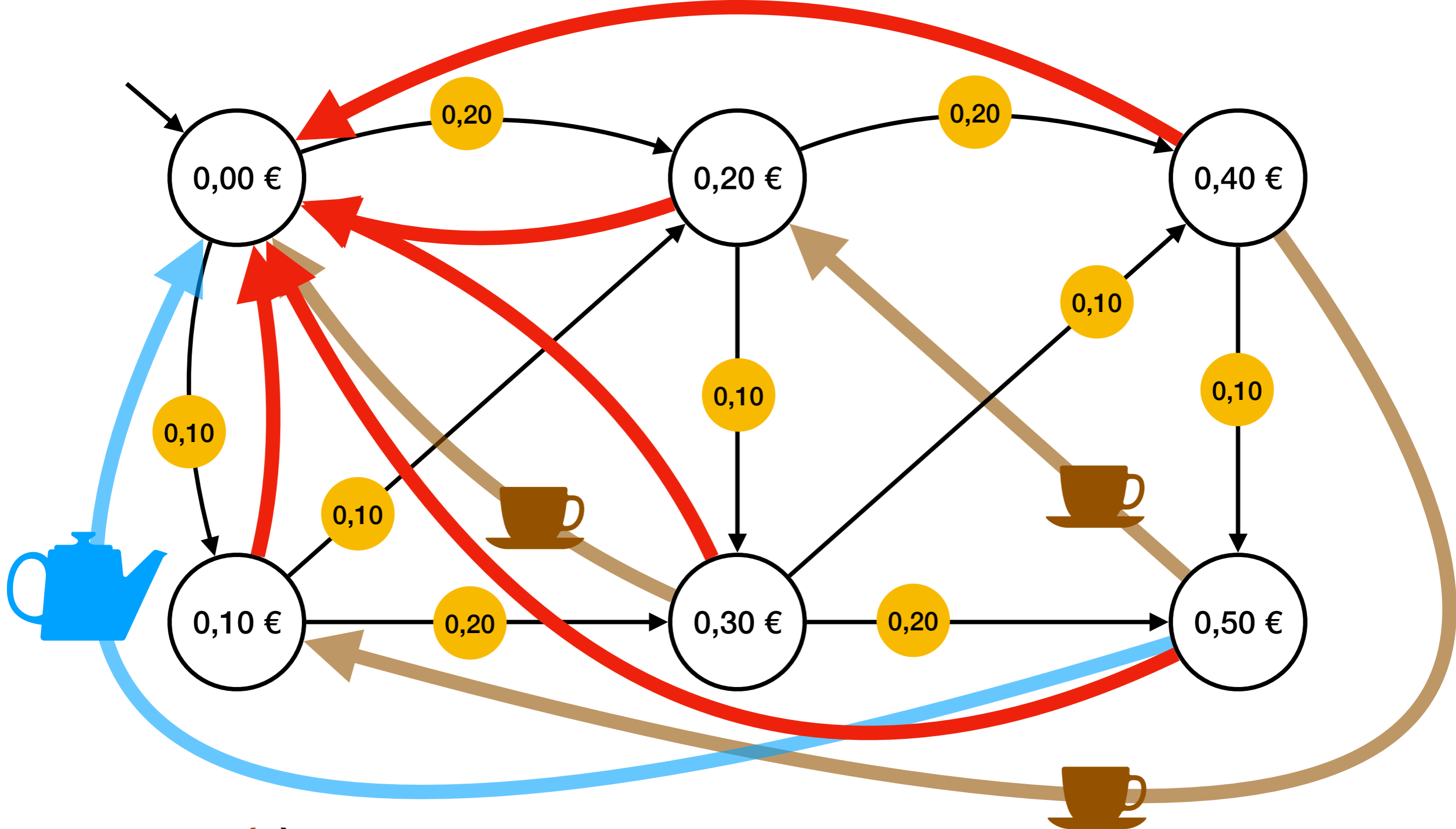


- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)





- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)



- **Café** à 0,30 €
- **The** à 0,50 €
- La machine n'accepte que les **pièces** de 0,10 € et 0,20 €
- On peut insérer dans la machine un maximum de 0,50 €
- On peut **annuler** à tout moment (et récupérer la monnaie)

# Limites des automates finis

Il n'existe pas un automate fini qui accepte exactement l'ensemble infini de mot suivant :

01  
0011  
000111  
00001111  
0000011111  
000000111111  
⋮



*The End*

