

Structures de données

Arbres

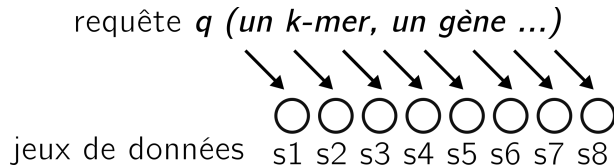
Master 1 MISO

Camille Marchet
CNRS, CRIStAL Lille, France

camille.marchet@univ-lille.fr

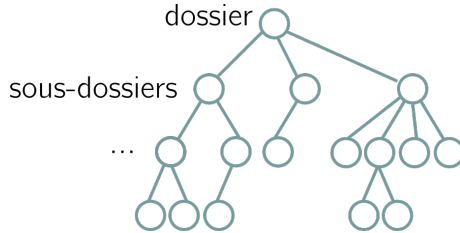


Introduction - Une question de bioinformatique



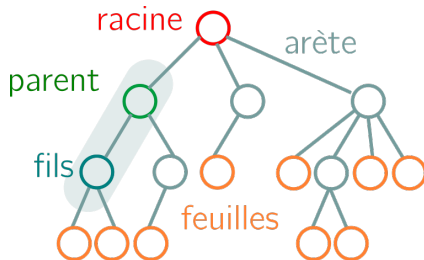
Complexité (hypothèse : la recherche est en temps constant dans chaque jeu de donnée) ?

Arbres



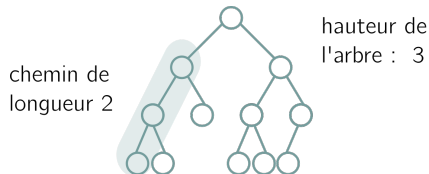
- structure non linéaire
- organisation, hiérarchisation de données
- exemple : arborescence de fichiers de votre pc

Arbres - définitions



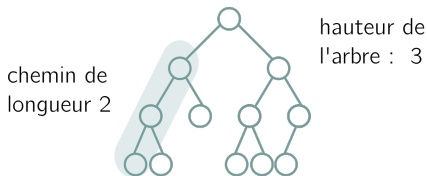
- noeud parent/ noeud fils
- ensemble de noeuds
 - noeud racine : unique noeud sans parent
 - feuilles : noeuds sans fils
 - noeuds internes : noeuds qui se sont pas des feuilles

Arbres - définitions



- Chemin : soit n_0, n_1, \dots, n_{p-1} une suite de noeuds tels que n_i est le parent de n_{i+1} pour $0 \leq i < p$
- la longueur de ce chemin est $p - 1$ (pour p noeuds)

Arbres - définitions



- la hauteur d'un noeud n : longueur maximale d'un chemin allant de n à une feuille
- la hauteur de l'arbre est la hauteur de sa racine
- la profondeur (ou le niveau) d'un noeud est la longueur du chemin de la racine à ce noeud

Arbres binaires

Un arbre binaire est :

- soit un arbre vide
- soit un arbre dans lequel chaque noeud possède 0, 1 ou 2 successeur(s)
- l'arbre vide ne possède aucun noeud

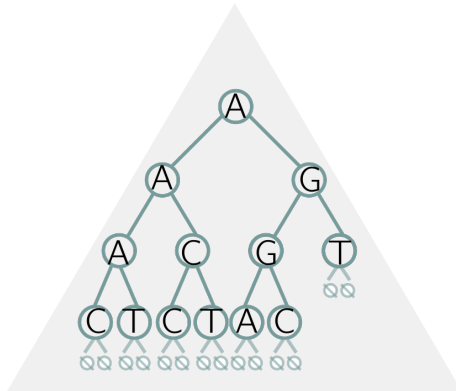


Une définition récursive d'un arbre binaire

Un arbre binaire est soit

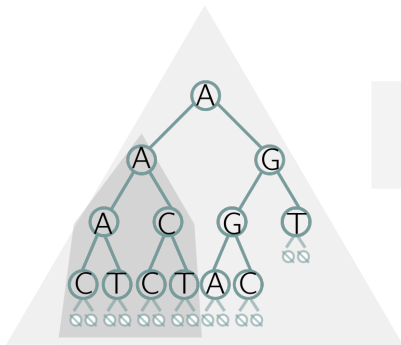
- un arbre vide
- un triplet (v, g, d) où
 - v est une valeur et
 - g et d sont les sous-arbres droit et gauche

Une définition récursive d'un arbre binaire



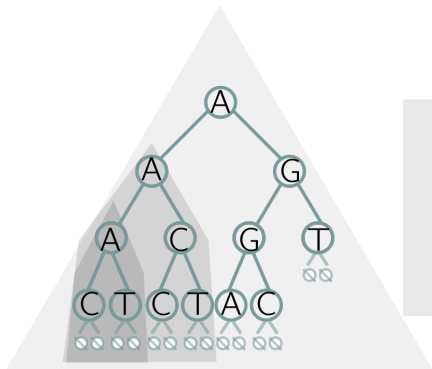
arbre

Une définition récursive d'un arbre binaire



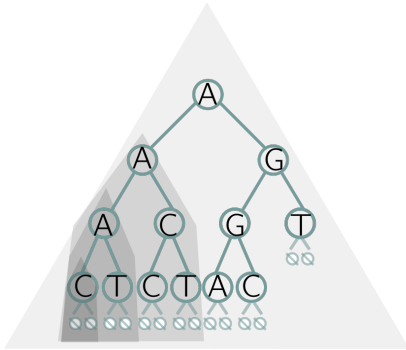
```
arbre(  
  "A", arbre_g, arbre_d  
)
```

Une définition récursive d'un arbre binaire



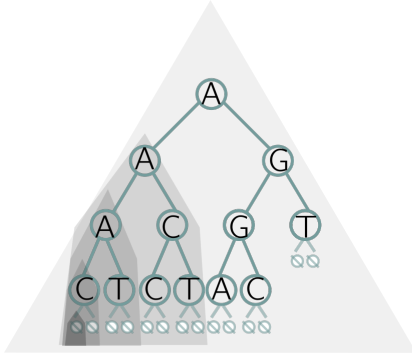
```
arbre(  
    "A",  
    arbre("A", arbre_g, arbre_d),  
    arbre_d  
)
```

Une définition récursive d'un arbre binaire



```
arbre(  
  "A",  
  arbre(  
    "A",  
    arbre("A", arbre_g, arbre_d),  
    arbre_d  
  ),  
  arbre_d  
)
```

Une définition récursive d'un arbre binaire



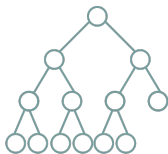
```
arbre(  
  "A",  
  arbre(  
    "A",  
    arbre(  
      "A",  
      arbre("C", arbre(), arbre())  
      arbre_d),  
      arbre_d  
    ),  
    arbre_d  
  )  
)
```

Arbres binaires - définitions

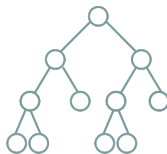
- arbre binaire complet
toutes les feuilles sont à la même distance de la racine
- arbre binaire localement complet
tous les noeuds possèdent zéro ou deux fils
- arbre binaire presque complet
tous les niveaux sont remplis à l'exception du dernier, rempli partiellement en commençant par la gauche



complet



presque complet



localement complet

Parcours d'un arbre

Parcours en profondeur :

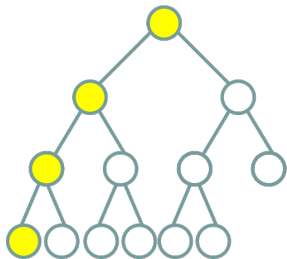
- à chaque noeud rencontré on descend dans le sous-arbre

Parcours en largeur

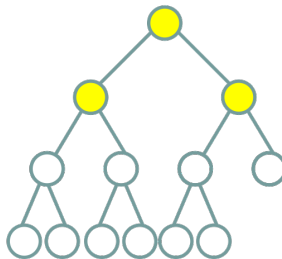
- chaque noeud d'un même niveau est visité avant de descendre au niveau inférieur

Parcours d'un arbre

profondeur

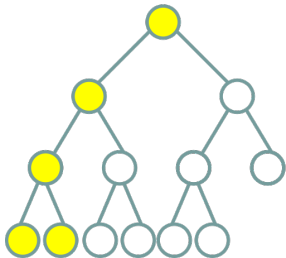


largeur

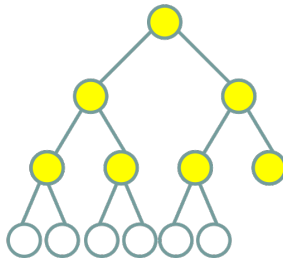


Parcours d'un arbre

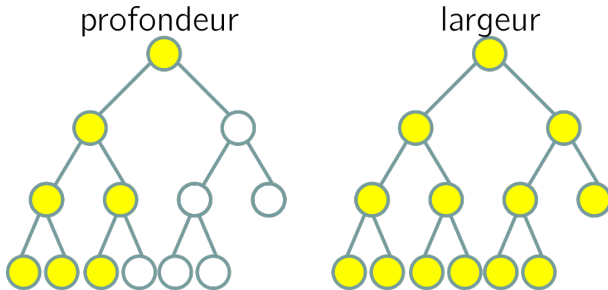
profondeur



largeur

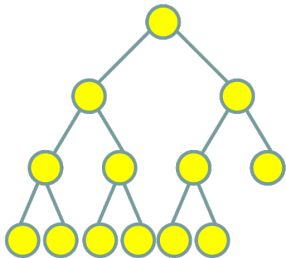


Parcours d'un arbre

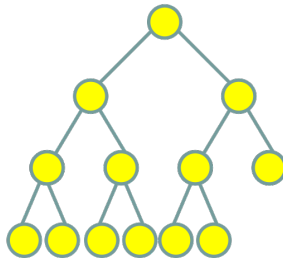


Parcours d'un arbre

profondeur



largeur



Parcours en profondeur d'un arbre

3 types de parcours en profondeur :

- Parcours préfixe

1. visiter la racine
2. parcourir le sous-arbre gauche suivant l'ordre préfixe
3. parcourir le sous-arbre droit suivant l'ordre préfixe

- Parcours infixé

1. parcourir le sous-arbre gauche suivant l'ordre infixé
2. visiter la racine
3. parcourir le sous-arbre droit suivant l'ordre infixé

- Parcours suffixe

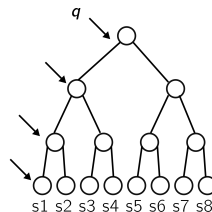
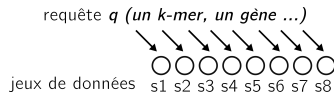
1. parcourir le sous-arbre gauche suivant l'ordre suffixe
2. parcourir le sous-arbre droit suivant l'ordre suffixe
3. visiter la racine

Parcours en largeur d'un arbre

1. Ajouter la racine dans une liste des nœuds à traiter
2. Tant que la liste des noeuds à traiter n'est pas vide
3. Retirer le premier élément de la liste des noeuds à traiter et le prendre comme noeud courant
4. Ajouter les enfants du noeud courant à la fin de la liste des noeuds à traiter
5. Ajouter le noeud courant à la liste représentant le parcours
6. Retourner la liste représentant le parcours

Des arbres en bioinformatique

- Modélisation/représentation des données : phylogénie
- Structures de données : Sequence Bloom trees



Une remarque sur la complexité ?

Topologies

- Arbre filiforme : chaque noeud a au plus un fils
- Arbre équilibré : profondeurs de toutes les feuilles égales à ± 1 près

Arbres binaire de recherche (ABR)

Arbre binaire dans lequel la valeur à la racine (si elle existe) est supérieure ou égale à toutes les valeurs du sous-ABR gauche et strictement inférieure à toutes les valeurs du sous-ABR droit.

Autrement dit, le parcours infixe d'un ABR donne la liste des valeurs triées dans l'ordre croissant.