

cdm2018.zip sur site

1 Problématique

La France a (brillamment) gagné la coupe du Monde de football 2018. Pour stocker le tableau des phases finales (figure 1) le site web *lateam.fr* utilise une structure simple de données.

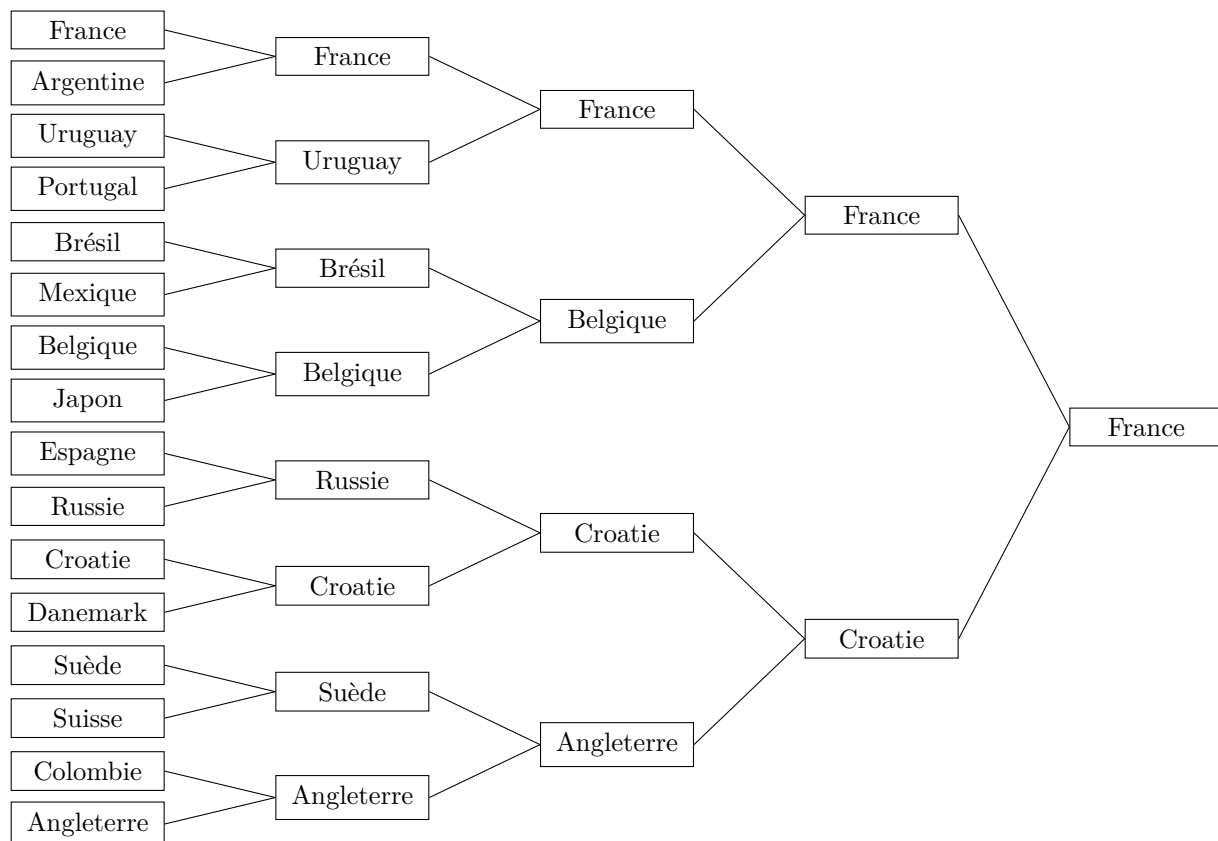


FIGURE 1 – Coupe du Monde 2018 - phases finales

Comment représenter un arbre binaire en mémoire en utilisant une structure de données native ?

arbre binaire parfait de hauteur 4 ; *feuilles* = 2^4 ; *taille* = $2^{4+1} - 1 = 31$; sous-arbre gauche en haut

2 Propriété des indices

L'arbre binaire (figure 2) est numéroté en utilisant un parcours en largeur.

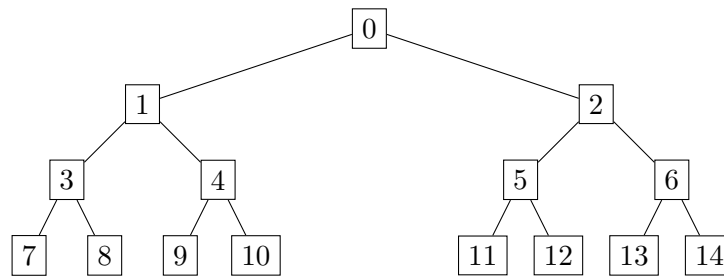
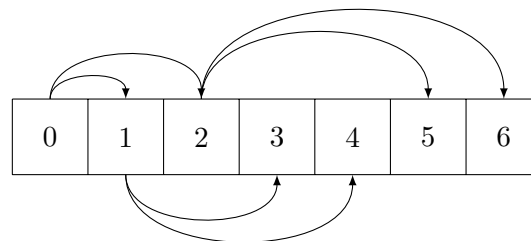


FIGURE 2 – Relation entre les indices

Pour chaque nœud i qui a des fils, on peut remarquer que :

- l'indice du fils gauche est $2 * i + 1$,
- l'indice du fils droit est $2 * i + 2$.

Un arbre binaire peut alors être stocké dans un simple tableau.



Code 1 – Un arbre binaire dans un tableau

si arbre pas parfait certaines cases restent vides.

Activité 1 :

1. Télécharger le fichier `cdm2018.zip` sur le site <https://cviroulaud.github.io>.
2. Ouvrir le fichier `cdm2018.json` et vérifier à la main que le tableau représente bien l'arbre binaire des phases finales de la coupe du Monde 2018.
3. Importer le `json` dans un programme Python.
4. Écrire la fonction `affichage_BFS(tab : list) → None` qui affiche les éléments de `tab` dans l'ordre d'un parcours en largeur de l'arbre binaire correspondant.