

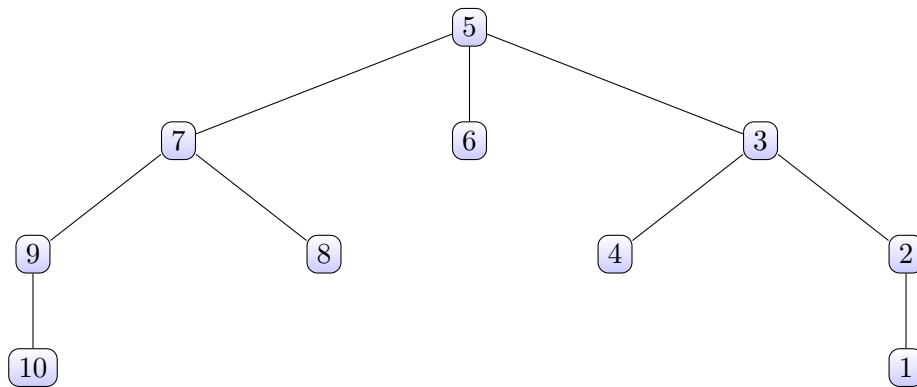
(*) Exercices à faire à la maison.

Exercice 1. *Parcours.*

Dessiner l'arbre général dont le résultat des parcours préfixe et suffixe sont respectivement $[1, 2, 4, 6, 3, 5]$ et $[4, 6, 2, 5, 3, 1]$.

Exercice 2. *Codage de Prüfer.*

Donner le codage de Prüfer de l'arbre suivant



Exercice 3. *Encore codage de Prüfer.*

On considère un arbre général dont les sommets sont numérotés bijectivement sur $\{1, 2, \dots, 16\}$ et dont le code de Prüfer est $[5, 4, 5, 9, 12, 4, 1, 14, 5, 9, 10, 14, 1, 1, 9]$.

1. Quelle est l'étiquette du sommet de l'arbre ? Quelles sont les étiquettes de ses feuilles ?
2. Construire l'arbre général correspondant à ce codage.
3. * Coder l'algorithme (dans votre langage favori) qui prend en entrée un entier n , et un codage de Prüfer, et renvoie l'arbre correspondant à ce codage qui contient les sommets $\{1, \dots, n\}$.

Exercice 4. *Vers les arbres binaires.*

1. Donner l'image de l'arbre de l'exercice du codage de Prüfer par la fonction f ("arbres généraux \rightarrow arbres binaires sans sous-arbre droit") vue en cours.
2. Donner un exemple d'arbre binaire qui n'a pas d'antécédent par cette fonction.
3. Prouver par récurrence sur la hauteur d'un arbre général, que l'ordre préfixe est conservé pour tout arbre général. On peut pour ça étendre la fonction f aux forêts
4. * Prouver de même que l'ordre suffixe (ou postfixe), devient l'ordre infixé.

Soit t un arbre binaire. La *canopée* de t est un mot binaire codant l'orientation des feuilles de t considérées de gauche à droite, tel que si la feuille est orientée vers la gauche (c'est-à-dire est fille gauche d'un nœud) on la code par 0 et si la feuille est orientée vers la droite (c'est-à-dire est fille droite d'un nœud) on la code par 1.

Écrire une fonction *canop* paramétrée par un arbre binaire *t* qui renvoie la liste de 0 et de 1 correspondant à la canopée de *t*.

Un arbre binaire t est *équilibré* si et seulement si pour tout nœud interne x de t ,

Écrire une fonction `est_equilibre` paramétrée par un arbre binaire `t` qui teste si `t` est équilibré.