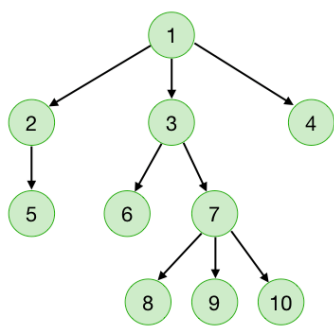
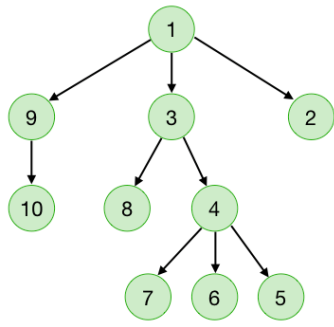


les prochains nœuds à visiter. Cet algorithme suppose que le coût de toutes les actions est égal à une constante $c \geq 0$.



□ **Parcours en profondeur (DFS)** – L’algorithme de parcours en profondeur (en anglais *depth-first search* ou *DFS*) est un algorithme de parcours de graphe traversant chaque chemin qu’il emprunte aussi loin que possible. On peut le coder de manière récursive, ou itérative à l’aide d’une pile qui stocke à chaque étape les prochains nœuds à visiter. Cet algorithme suppose que le coût de toutes les actions est égal à 0.



□ **Approfondissement itératif** – L’astuce de l’approfondissement itératif (en anglais *iterative deepening*) est une modification de l’algorithme de DFS qui l’arrête après avoir atteint une certaine profondeur, garantissant l’optimalité de la solution trouvée quand toutes les actions ont un même coût constant $c \geq 0$.

□ **Récapitulatif des algorithmes de parcours d’arbre** – En notant b le nombre d’actions par état, d la profondeur de la solution et D la profondeur maximale, on a :

Algorithme	Coût des actions	Espace	Temps
Retour sur trace	peu importe	$\mathcal{O}(D)$	$\mathcal{O}(b^D)$
Parcours en largeur	$c \geq 0$	$\mathcal{O}(b^d)$	$\mathcal{O}(b^d)$
Parcours en profondeur	0	$\mathcal{O}(D)$	$\mathcal{O}(b^D)$
DFS-approfondissement itératif	$c \geq 0$	$\mathcal{O}(d)$	$\mathcal{O}(b^d)$