

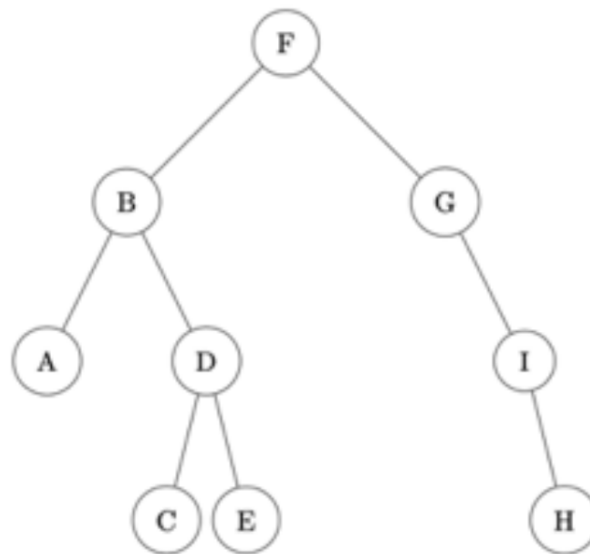
Exercice 1 :

1. Écrire une fonction **recherche** qui prend en paramètre un tableau de nombres entiers **tab**, et qui renvoie la liste (éventuellement vide) des couples d'entiers consécutifs successifs qu'il peut y avoir dans **tab**.
2. Proposer un jeu de tests permettant de vérifier plusieurs cas de figures.

Exercice 2 :

- On appelle « mot » une chaîne de caractères composée avec des caractères choisis parmi les 26 lettres minuscules ou majuscules de l'alphabet.
 - On appelle « phrase » une chaîne de caractères : composée avec un ou plusieurs « mots » séparés entre eux par un seul caractère espace ' ', se finissant :
 - soit par un point '.' qui est alors collé au dernier mot,
 - soit par un point d'exclamation '!' ou d'interrogation '?' qui est alors séparé du dernier mot par un seul caractère espace ' '.
1. Construire plusieurs exemples de phrase et observer le lien entre le nombre d'espaces et le nombre de mots.
 2. Écrire la fonction **nombre_de_mots(phrase: str) → int** qui renvoie le nombre de mots dans **phrase**.

Exercice 3 : Un arbre binaire de caractères est stocké sous la forme d'un dictionnaire où les clefs sont les caractères des nœuds de l'arbre et les valeurs, pour chaque clef, la liste des caractères des fils gauche et droit du nœud.



1. Construire le dictionnaire **arbre** représentant l'arbre binaire ci-dessus.
2. Écrire la fonction récursive **taille(arbre: dict) → int** qui renvoie la taille de l'**arbre**.