

TP 4 : Traduction systématique d'exemples complexes

Le but de cette séance est de consolider les acquis des séances précédentes.

Les exercices sont organisés comme dans les TP précédents : dans le répertoire `tp4/` de votre dépôt, pour chaque fonction `exo`, vous avez un fichier `exo.c` qui contient le programme principal, un fichier `fct_exo.s` à remplir, et une règle de génération dans le `Makefile` (`make exo`) pour générer l'exécutable. Pour chaque fonction demandée, il convient de vérifier l'exécution pas à pas du programme avec le débogueur (`gdb`) et le simulateur (`qemu`).

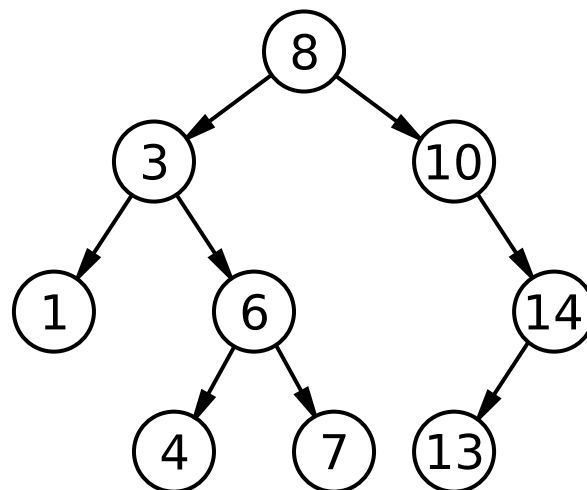
Ex. 1 : Parcours d'arbres binaires de recherche

On va travailler dans cet exercice sur des arbres binaires de recherche (ABR) dont les nœuds contiennent une valeur représentable sur un entier naturel de 32 bits. On rappelle les principales propriétés de cette structure de données :

- chaque nœud de l'arbre a au plus 2 fils ;
- le sous-arbre gauche d'un nœud N donné ne contient que des nœuds dont la valeur est strictement inférieure à la valeur de N ;
- le sous-arbre droit d'un nœud N donné ne contient que des nœuds dont la valeur est strictement supérieure à la valeur de N ;
- les sous-arbres gauche et droit d'un nœud donné sont aussi des arbres binaires de recherche.

Il vient naturellement des propriétés ci-dessus que chaque valeur est unique (*i.e.* il n'existe pas plusieurs nœuds de même valeur). Autrement dit, la structure de données ABR met en œuvre le type abstrait *ensemble*.

Le schéma ci-dessous est un exemple d'ABR qu'on va utiliser dans cet exercice :



On représente le type des nœuds d'un ABR par la structure C suivante :