

**Partiel PROG5**

(Documents non autorisés - durée 2h00)

(Le barème est donné à titre indicatif, il est susceptible d'évoluer)

✕ **Exercice 1 : (Questions de Cours) [3pts]**

- ✎ Donnez la définition formelle d'un graphe réciproque et d'un graphe symétrisé.
- ✎ Soient  $G = (S, A)$  un graphe orienté et  $x$  un sommet de  $S$ . Donnez les définitions formelles des descendants et des ascendants du sommet  $x$  dans  $G$ .

**Exercice 2 : (Tris – 4pts)**

1. Rappeler le principe de fonctionnement du tri fusion.
2. Après avoir donné le type en C permettant de représenter une suite d'entiers, écrire une procédure en C permettant de la trier en utilisant le tri fusion.

✕ **Exercice 3 : (Arbres – 4pts)**

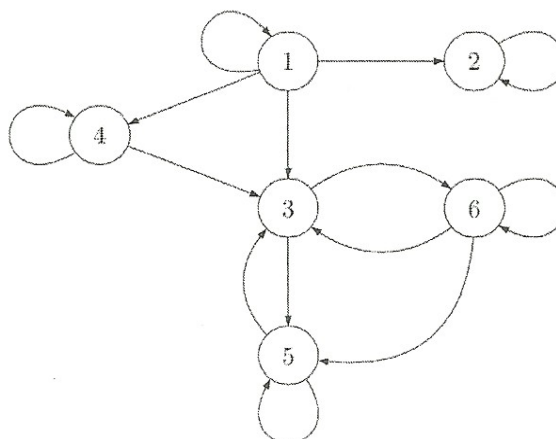
Soit un arbre binaire de recherche.

1. Définir le type arbre binaire de recherche en C
2. Deux arbres binaires sont dits similaires, s'ils ont la même structure (pas forcément les mêmes éléments). Ecrire une fonction permettant de tester si deux arbres binaires sont similaires.
3. Ecrire une procédure permettant de construire l'arbre miroir d'un arbre binaire i.e. en tout nœud, échanger les sous arbres gauche et droit.

✕ **Exercice 4 : (Graphes – 5 pts)**

Soit  $G = (S, A)$  un graphe orienté.

1. Donnez les deux types (en C) permettant de représenter un tel graphe (matrice d'adjacence et liste d'adjacence).
2. En utilisant le type liste d'adjacence, écrire une procédure permettant de calculer le degré intérieur de chacun des sommets du graphe  $G$ .
3. En utilisant le type liste d'adjacence, donnez l'algorithme de parcours en profondeur d'un graphe ainsi que sa complexité.
4. En utilisant le graphe suivant, montrez l'ordre de visite des sommets par un tel parcours.



✕ **Exercice 5 : (Plus courts chemins) [4 pts]**

1. Rappeler les conditions d'application de l'algorithme Dijkstra
2. Appliquer cet algorithme au graphe suivant (pour les plus courts chemins depuis 0) :

