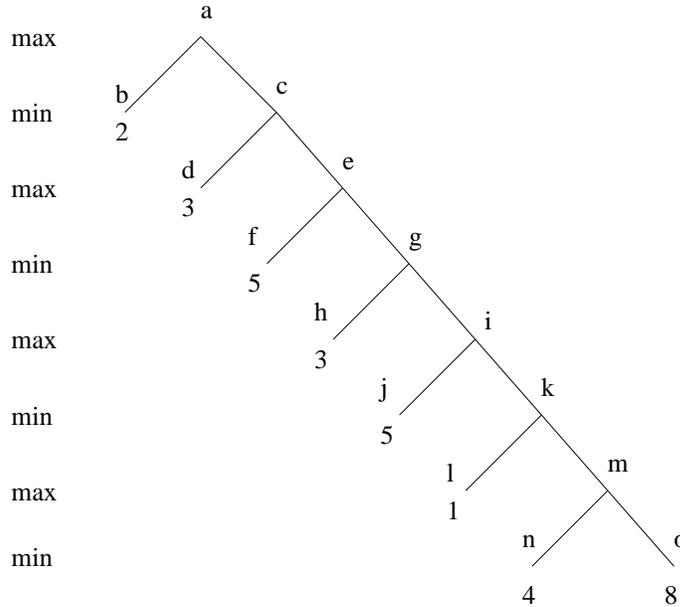


**Informations :** Tous les documents sont autorisés. Le barème est donné à titre indicatif et pourra être modifié.

**Exercice 1 Jeux (5 points)**

Considérez l'arbre de jeu suivant. La racine est un nœud max.



- Appliquez l'algorithme minimax sur cet arbre.
- Appliquez l'algorithme  $\alpha$ - $\beta$  sur cet arbre en le parcourant de gauche à droite et en commençant avec les valeurs initiales  $\alpha = -\infty, \beta = +\infty$ .
- Est-ce qu'on peut donner un arbre de jeu (pas forcément avec toutes les feuilles au même niveau) avec 1000 feuilles, pour lequel l'algorithme  $\alpha$ - $\beta$  (avec valeurs initiales:  $\alpha = -\infty, \beta = +\infty$ ) ne considère que 2 feuilles ? Donnez schématiquement un tel arbre ou montrez que ce n'est pas possible.

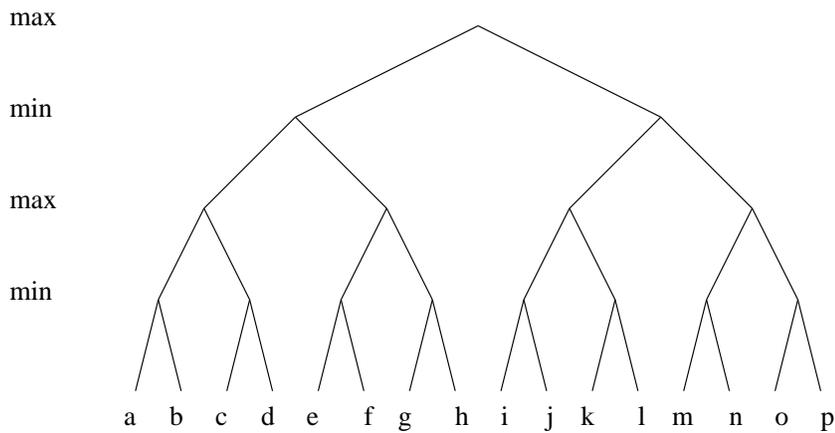
**Exercice 2 Algorithme de recherche (6 points)**

Nous considérons un monde avec 4 pions (A,B,C,D) non superposables. Ils peuvent être arrangés dans n'importe quel ordre, sauf A qui ne peut pas être plus à droite que D. Par exemple, ABCD et CBAD sont deux états possibles du monde, tandis que DCBA et CDAB ne sont pas possibles. Le monde peut être manipulé par une action de la forme *échange*( $x, y$ ) qui échange les pions des positions  $x$  et  $y$ . Par exemple *échange*(1, 2) transforme BCAD dans CBAD. Seules les actions *échange*(1, 2), *échange*(2, 3) et *échange*(2, 4) sont autorisées. Ils donnent un successeur uniquement si la situation atteinte est possible.

- Dessinez le graphe d'états.
- On suppose que l'état de départ est ADBC et l'état que l'on veut atteindre est CBAD. On suppose que chaque action coûte 1. Donnez une "bonne" heuristique  $h$  **admissible** (mais aussi **différente de 0** pour les nœuds non-finaux) pour ce problème. Le principe de l'heuristique devrait être suffisamment général pour pouvoir s'appliquer à des problèmes similaires.
- Appliquez la recherche gloutonne avec votre heuristique. Si vous n'avez pas trouvé d'heuristique, utilisez l'heuristique  $h = 0$ . Ne considérez pas les nœuds déjà développés. En cas d'égalité choisissez un nœud à développer au hasard.
- Appliquez la recherche A\* avec votre heuristique. Si vous n'avez pas trouvé d'heuristique, utilisez l'heuristique  $h = 0$ . Ne considérez pas les nœuds déjà développés. En cas d'égalité choisissez un nœud à développer au hasard.

**Exercice 3** Algorithme  $\alpha$ - $\beta$  (5 points)

Considérez l'arbre de jeu suivant:



Donnez des valeurs aux feuilles  $a$  à  $p$  de sorte que l'algorithme  $\alpha$ - $\beta$  (avec valeurs initiales:  $\alpha = -\infty, \beta = +\infty$ ) **ne coupe aucune** branche avec un parcours de gauche à droite. Utilisez le moins de valeurs différentes que possible !

**Exercice 4** Algorithme de recherche (4 points)

On considère l'algorithme de recherche à faisceau. Cet algorithme est basé sur  $A^*$ . Il utilise la même valeur  $f = g + h$ . La différence est que dans chaque pas on met uniquement les  $k$  meilleurs enfants (c-à-d. les enfants avec les  $k$  plus petites valeurs  $f = g + h$ ) du nœud considéré dans la liste des nœuds encore à traiter.

- Donnez un exemple d'un espace d'états fini (et une valeur de  $k \geq 2$ ) avec heuristique admissible où cet algorithme ne trouve pas la solution alors que  $A^*$  la trouve.
- Donnez un exemple d'un espace d'états fini (et une valeur de  $k \geq 2$ ) avec heuristique admissible où cet algorithme ne trouve pas la solution optimale.