Légende

S_j	état contrôlé par J_2	S_i	état contrôlé par J_1
S_j	victoire du joueur J_2	S_i	victoire du joueur J_1

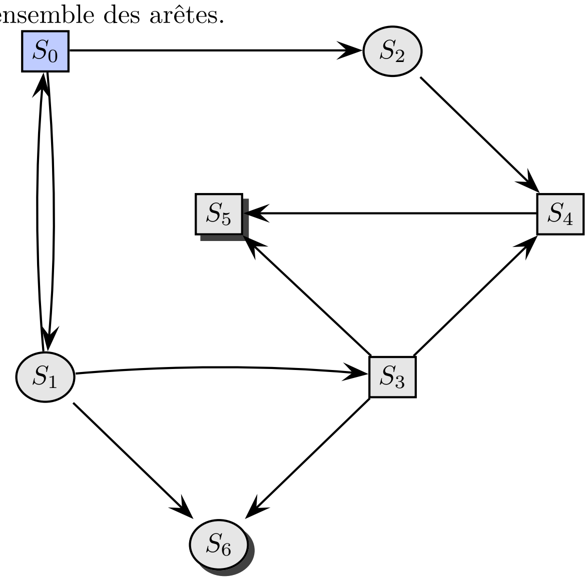
I. Calcul de positions gagnantes sur un graphe

On considère un jeu d'accessibilité, où les états victorieux des joueurs sont représentés pour le joueur J_1 par un rectangle ombré, pour le joueur J_2 par un cercle ombré. Le joueur 1 joue les noeuds carrés, le joueur 2 joue les noeuds ronds.

Quand un joueur contrôle un noeud, c'est lui qui choisit dans lequel des noeuds accessibles on va au coup suivant.

Initialement le joueur 2 démarre la partie sur le noeud S_0.

On note $G = (V, E)$ où V est l'ensemble des sommets, E l'ensemble des arêtes.



1. Justifiez que l'arène du jeu est bipartie.
2. Justifier que l'attracteur pour le joueur J_1 doit être initialisé à $A_0 = \{S_5\}$.
3. Calculer A_1, A_2.
   on rappelle que A_{n+1} s'obtient en ajoutant à A_n les noeuds appartenant à J_1 qui permettent de rejoindre un noeud de A_n ou les noeuds appartenant à J_2 qui conduisent obligatoirement à rejoindre un noeud de A_n
4. En déduire les positions gagnantes du joueur 1.
5. En notant V_1^+ l'ensemble des sommets de degré sortant non nul contrôlés par le joueur 1, Proposer une stratégie $\varphi : V_1^+ \to V$, telle que $\forall s \in V_1^+$, $(s, \varphi(s)) \in E$, pour le joueur 1.
6. Proposez une stratégie $\psi : V_2^+ \to V$ pour le joueur 2.

II. Python et dictionnaires pour déterminer les positions gagnantes et l'attracteur

Connectez-vous via Toutatice à l'adresse
`https://capytale2.ac-paris.fr/web/c/dd3a-1142026`

Le graphe précédent peut être représenté par le dictionnaire suivant :
`g1='S0':['S1','S2'],'S1':['S3'],'S2':['S4'] ,'S3':['S4','S5','S6'] ,'S4':['S5'],'S5':[],'S6':[]`

1. Que représente les valeurs associées à chaque clé dans ce dictionnaire ?
2. `https://capytale2.ac-paris.fr/web/c/dd3a-1142026`
3. Ecrire une fonction `transpose(g)` qui prend en paramètre un graphe orienté (dictionnaire sommet :liste de voisins) et retourne le graphe transposé obtenu en inversant les flèches du graphe.
4. Tester votre programme sur le graphe `g1='s0':['s1','s2'],'s1':['s0','s3','s6'],'s2':['s4'] ,'s3':['s4','s5'] ,'s4':['s5'],'s5':[],'s6':[]`
5. Dans la suite pour tenir compte de l'appartenance des sommets aux joueurs, on codera les sommets sous la forme d'un couple (numéro du joueur propriétaire, numéro du sommet), et on utilisera le graphe
   `g2=(1,0):[(2,1),(2,2)],(2,1):[(1,0),(1,3),(2,6)],(2,2):[(1,4)] ,(1,3):[(1,4),(1,5)] ,(1,4):[(1,5)],(1,5):[],(2,6):[]`
6. On va ensuite construire l'attracteur pour le joueur j à partir du graphe g, de son graphe transposé tg, de la liste W1 des sommets victorieux pour le joueur j.

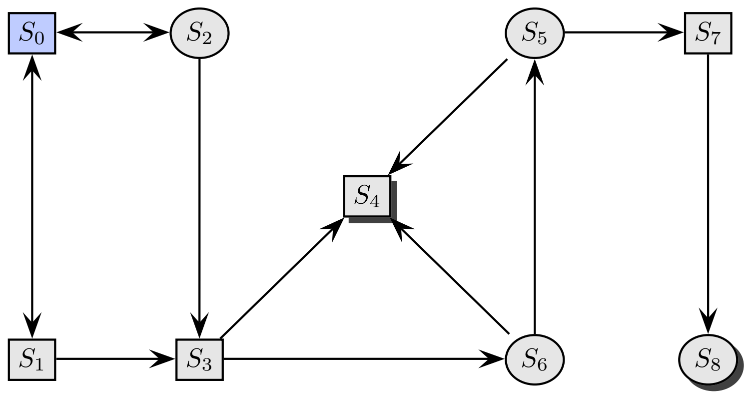
   Pour cela on va fixer un joueur j, initialiser une liste `aj` contenant les sommets ajoutés à l'attracteur, créer un dictionnaire `att` représentant son attracteur et le mettre à jour par itération en maintenant une liste `newaj` des sommets ajoutés à chaque étape.

   Nous allons implémenter une fonction next(g,tg,j,aj,att) qui retourne une nouvelle liste `newaj` sans doublons des nouveaux sommets ajoutés à l'attracteur à cette étape. Cette liste peut être vide. De plus cette fonction met à jour l'attracteur `att`.

   Vérifier l'initialisation de l'attracteur.
7. Compléter le code pour la fonction next()
8. Exécuter la première étape
   newaj=next(g2,tg2,1,newaj,att)
9. Exécuter les étapes suvantes jusqu'à avoir terminé le calcul de l'attracteur.
   newaj=next(g2,tg2,1,newaj,att)
10. En déduire une fonction attracteur(g,j,w) qui prend en arguments en un graphe g, un joueur j et ses conditions de victoires w et qui renvoie son attracteur.

III. Calcul de positions gagnantes sur un graphe

On reprend l'exercice précédent avec le nouveau graphe :



Déterminer les positions gagnantes de chaque joueur, l'attracteur pour le joueur 1 et élaborer une stratégie pour ce joueur.

IV. Jeu de Chomp classique

IV.1 Principe

on dispose d'une tablette de chocolat de $N \times P$ carrés, dont le carré $(0, 0)$ est empoisonné.

```
P # # # # #
# # # # # #
# # # # # #
# # # # # #
```

à tour de rôle chaque joueur choisit une case (a, b) et mange tout le quadrant inférieur droit et contenant cette case.
Le joueur qui se retrouve avec le dernier carré empoisonné à perdu.
exemple, le joueur choisit $(3, 4)$, il reste

```
P # # # # #
# # # # # #
# # #
# # #
```

IV.2 Pratique du jeu contre un ordinateur à stratégie aléatoire

Le code partagé est en ligne avec les identifiants toutatice, télécharger le lien (trombonne) depuis :

`https://capytale2.ac-paris.fr/web/c/dd3a-1142026`

Télécharger le fichier `TP1chomp`, puis jouez trois parties contre l'ordinateur.

IV.3 Existence d'une stratégie gagnante pour le premier joueur, non explicite

Théorème 1.

Si un jeu de Chomp débute par un rectangle A de dimensions $N \times P$ (N et $P \geq 1$), alors il existe un mouvement gagnant pour le premier joueur.

Considérons le mouvement consistant à choisir le carré en haut à droite.
Si ce mouvement n'est pas gagnant pour le joueur 1, alors le joueur 2 a un mouvement gagnant.
Mais tout mouvement choisi par le joueur 2 peut être joué par le joueur 1 au premier tour. Le joueur 1 a donc un mouvement gagnant. □

Remarque 1. Une telle preuve est appelée vol de stratégie. Il s'agit d'une preuve non constructive.

IV.4 Explicitation d'une stratégie gagnante pour le premier joueur

1. Proposer une stratégie gagnante pour le premier joueur pour un damier $1 \times P$, avec $P \geq 2$.
2. Une stratégie pour le premier joueur pour un damier $2 \times P$, avec $P \geq 2$ est la suivante : *obtenir un damier est de la forme* $2 \times Q$ *privé de son coin inférieur* $(1, Q-1)$ *à l'issue d'un tour de* J_1.

   • Si le mouvement de J_2 est sur la première ligne,

   ```
   P####
   #####
   ```

   on le force à la configuration

   ```
   P####
   ####
   ```

   en enlevant le carré en bas à droite.

   • Si le mouvement de J_2 est sur la seconde ligne,

   ```
   P############
   ####
   ```

   on le force à la configuration

   ```
   P####
   ####
   ```

   en enlevant les carrés les plus à droites dans la première ligne.

   Justifier qu'en répétant cela ,on parvient à l'état

   ```
   P#
   #
   ```

   qui conduit à une victoire de J_1 à l'issue du choix $(1, 0)$ ou $(0, 1)$ de J_2.
   Cette stratégie est-elle gagnante pour J_1 ?
3. Sur un carré (N, N), avec $N \geq 2$ initialement.
   Le premier joueur choisit la case $(1, 1)$, puis pour chaque choix (p, q) de son adversaire J_2, J_1 choisit ensuite toujours la case symétrique (q, p) par rapport à la diagonale.
   Expliquez pourquoi la stratégie de J_1 est gagnante.

IV.5 Stratégie gagnante pour le premier joueur sur un damier 2×3

Enumerez les états du système, vous devez en trouver 9.
Sur un graphe biparti cela fait donc 18 sommets.
Représenter le graphe et les arêtes pour ces 18 positions.
Identifiez l'attracteur pour le joueur J_1 qui démarre la partie, en coloriant en rouge sur votre graphe.

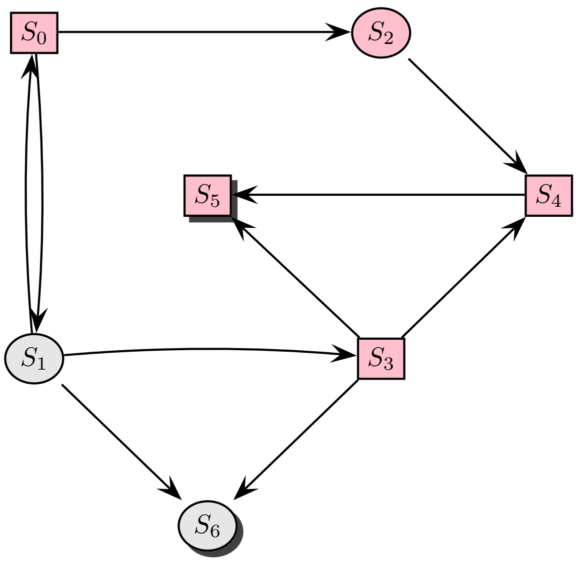
Correction : $V = V_1 \bigsqcup V_2$

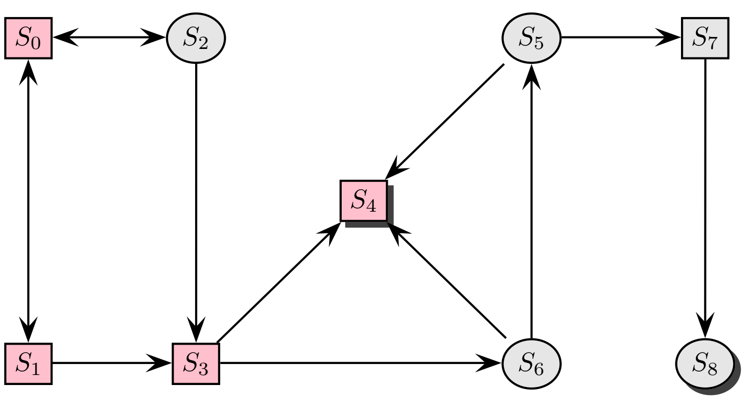
$A_0 = \{S_5\}$ on initialise l'attracteur avec condition de gain.

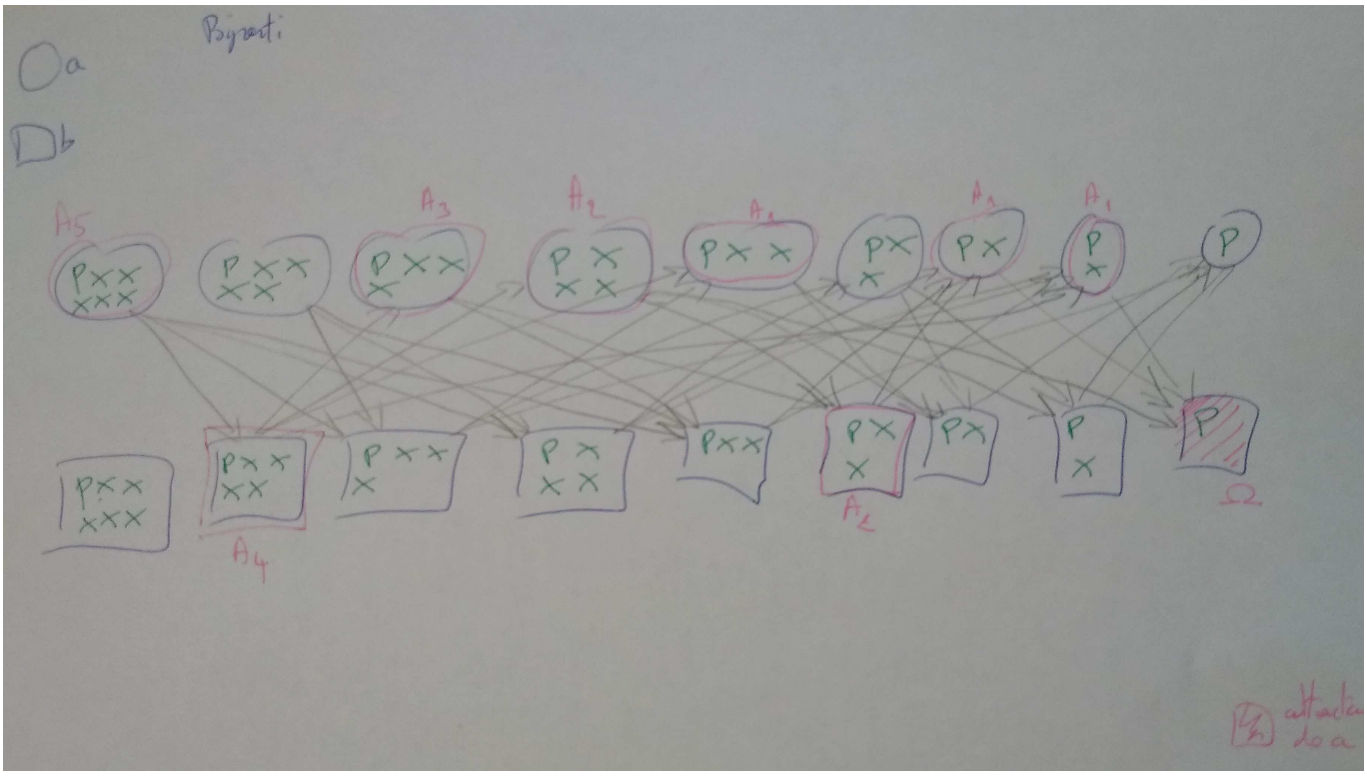
$A_1 = \{S_5, S_3, S_4\}$, car le joueur 1 doit rejoindre S_5 lorsqu'il est en S_4.

$A_2 = \{S_5, S_3, S_4, S_2\}$, car S_2 appartient au joueur 2 et ne permet que de rejoindre S_4 qui est une position gagnante.

$A_3 = \{S_5, S_3, S_4, S_2, S_0\}$ est stable, c'est l'attracteur pour le joueur 1






a
b
A5
A3
A2
A4
Ω