

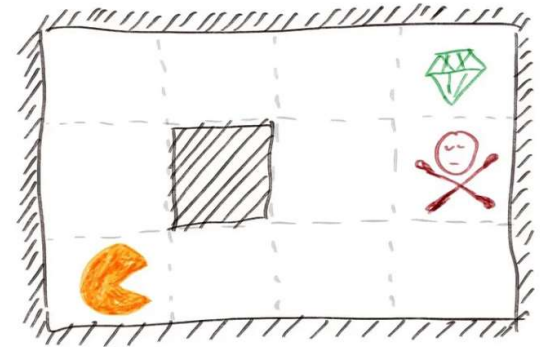
IFT 608 / IFT 702  
Planification en intelligence artificielle

Devoir 1

Présentation et spécification

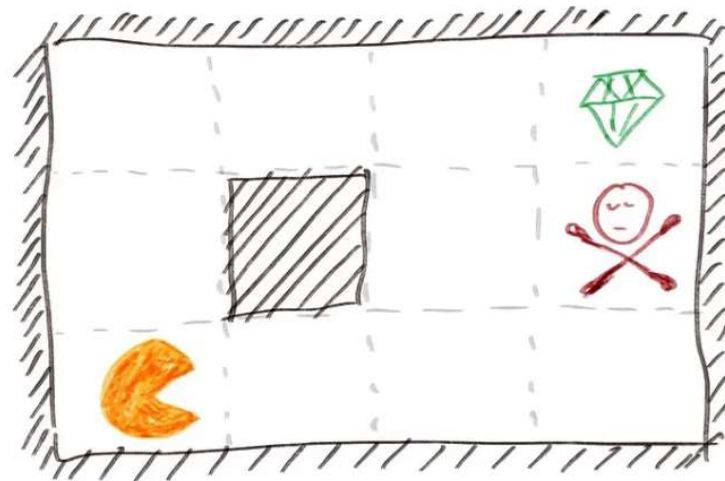
## Exercice 1 : (5 points)

- Choisir (1) un algorithme de planification à implémenter en Python, parmi ***value-iteration*** et ***policy-iteration*** (versions classiques, vues dans le cours), avec le domaine inspiré de *Pacman* simplifié (voir l'exemple, ci-après).



- Le programme doit prendre en entrée un fichier texte de spécification des conditions initiales et des paramètres. La grille est représentée par des codes (0, 1, 2 ou 3) pour les emplacements de la case vide, du but et du fantôme (voir l'exemple, ci-après). La position de départ de *Pacman* est constante (en bas, à gauche). Le fichier en entrée comporte aussi (sur une nouvelle ligne, pour chaque paramètre) les valeurs du facteur d'escompte (*gamma*), et s'il y a lieu, le seuil de tolérance (pour *value-iteration*).

- On suppose que le processus d'attribution des récompenses par état suit une fonction fixe de  $-0.04$ ,  $-1$  et  $1$  (voir l'illustration, ci-après). Pour simplifier, on suppose aussi que le fantôme ne bouge pas.



- Le nom du fichier doit absolument être : *value-iteration.txt* ou *policy-iteration.txt*, selon le choix adopté.
- Le programme fournit en sortie un fichier texte (appelé *log-file\_VI.txt* ou *log-file\_PI.txt*, selon le choix de l'algorithme implémenté) qui montre la trace complète de l'exécution, itération par itération, dans un format lisible et compréhensible par le correcteur humain (voir les exemples de traces de calcul, à chaque itération, vus dans le cours). Il n'y a pas de format spécifique pour la mise en forme. Toutefois, il faut que la trace soit détaillée et compréhensible.

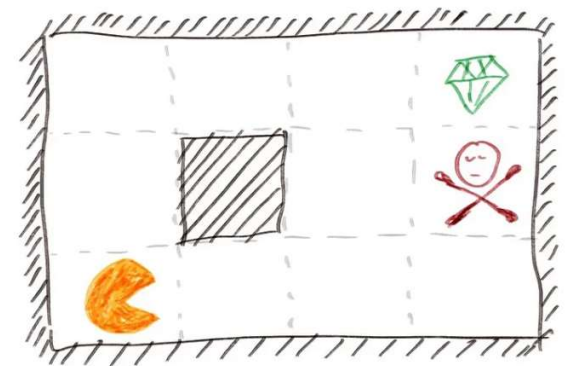
## **Exercice 2 : (5 points)**

- Reprendre le même domaine pour implémenter l'algorithme de planification par apprentissage actif ***Q-Learning*** (version classique vue dans le cours).

- Le fichier texte en entrée doit s'intituler *Q-Learning.txt* et doit contenir les conditions initiales et les paramètres, de la même façon que dans l'exercice 1. Les facteurs d'escompte (*gamma*) et d'apprentissage (*alpha*) sont précisés (dans cet ordre) chacun sur une ligne. Le dernier paramètre à figurer dans ce fichier est le nombre (entier) d'essais.
- Le programme fournit en sortie un fichier texte (appelé *logfile\_QL.txt*) qui montre la trace complète de l'exécution, itération par itération, essai par essai, dans un format lisible et compréhensible par le correcteur humain.

## Critères de correction :

- Clarté et lisibilité du code-source et des traces.
- Conformité des implémentations aux algorithmes théoriques.
- Respect des consignes, ci-dessus, pour le rendu.





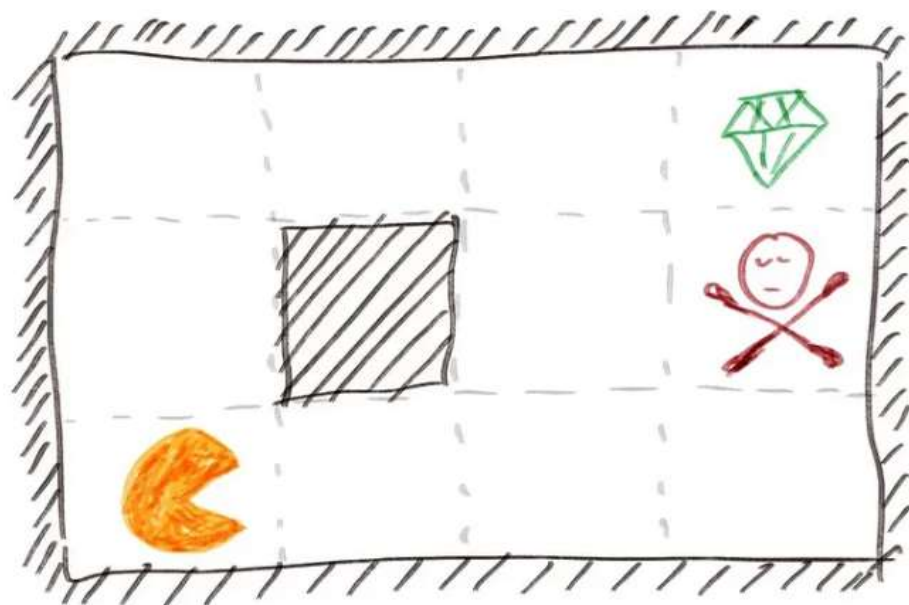
State and Reward

$s = 0$ $r = -0.04$	$s = 1$ $r = -0.04$	$s = 2$ $r = -0.04$	$s = 3$ $r = 1.0$
$s = 4$ $r = -0.04$	$s = 5$ $r = \text{nan}$	$s = 6$ $r = -0.04$	$s = 7$ $r = -1.0$
$s = 8$ $r = -0.04$	$s = 9$ $r = -0.04$	$s = 10$ $r = -0.04$	$s = 11$ $r = -0.04$

0,0,0,1

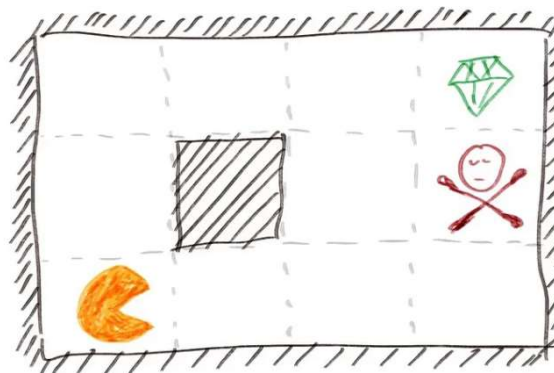
0,3,0,2

0,0,0,0

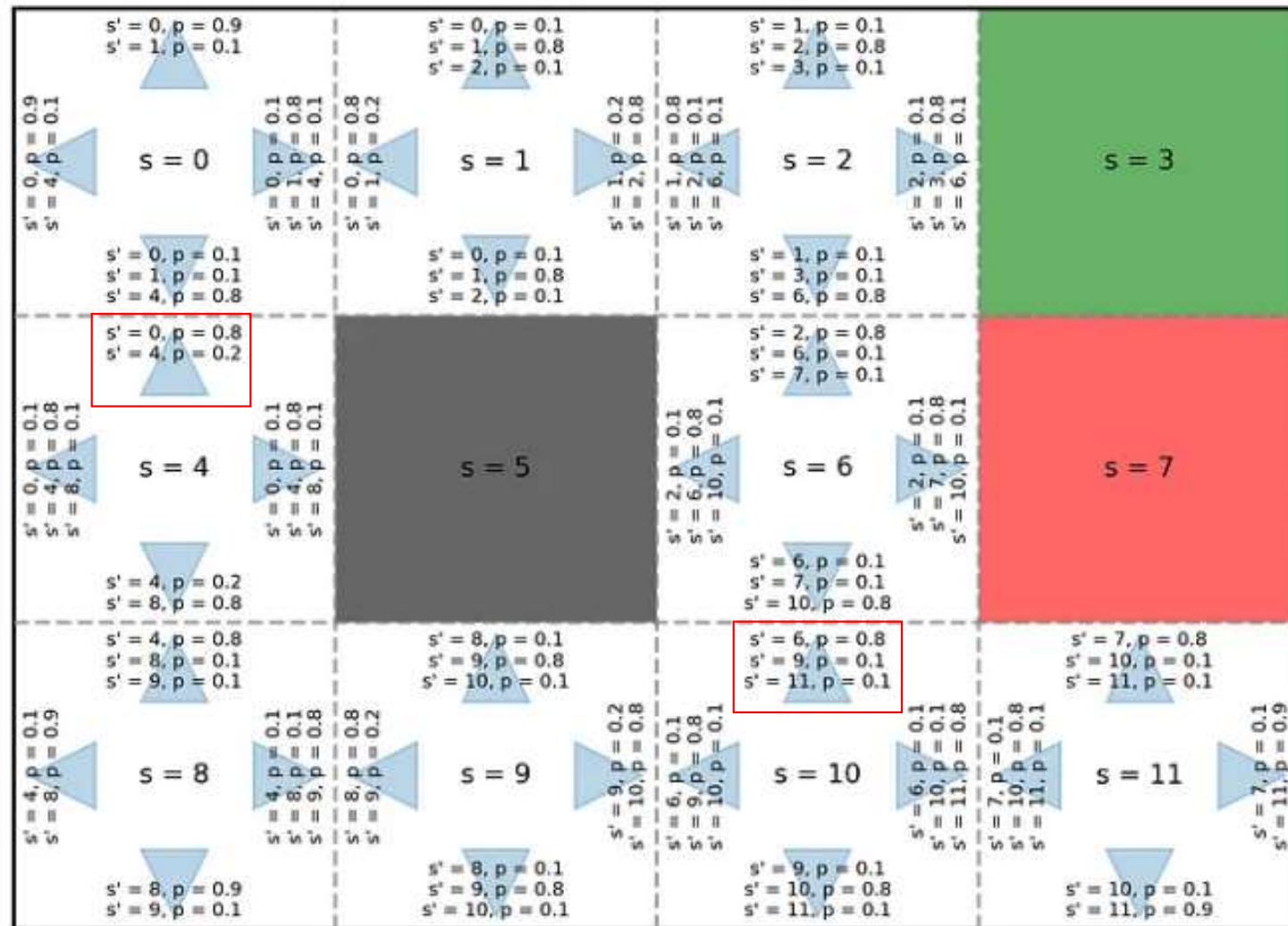


### **Transitional model**

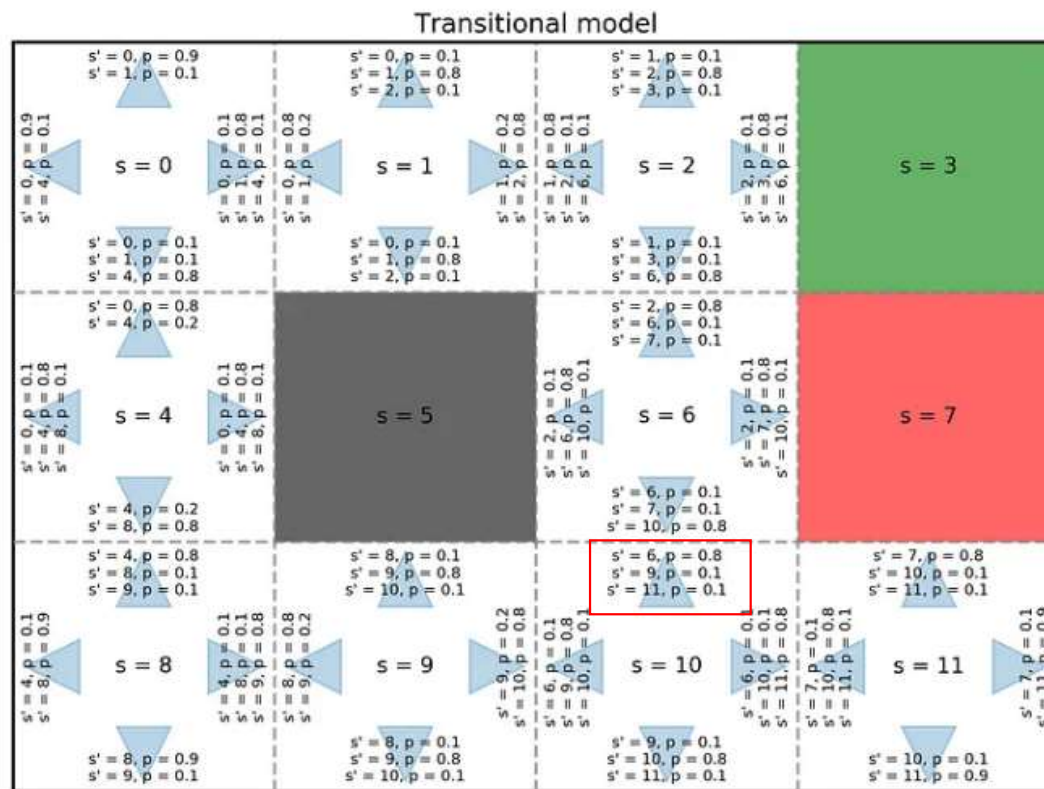
If we bump into a wall, we are bounced back and stay where we were. If we want to move up or down, we have 0.8 probability to actually go up or down, 0.1 probability to go left, and 0.1 probability to go right. Similarly, if we want to move left or right, we have 0.8 probability to actually go left or right, 0.1 probability to go up, and 0.1 probability to go down.



# Transitional model



transition model when  $s = 10$  and  $a = \text{UP}$ .



$$P(s' = 0 | s = 10, a = \text{UP}) = 0$$

$$P(s' = 1 | s = 10, a = \text{UP}) = 0$$

$$P(s' = 2 | s = 10, a = \text{UP}) = 0$$

...

$$P(s' = 6 | s = 10, a = \text{UP}) = 0.8$$

$$P(s' = 7 | s = 10, a = \text{UP}) = 0$$

$$P(s' = 8 | s = 10, a = \text{UP}) = 0$$

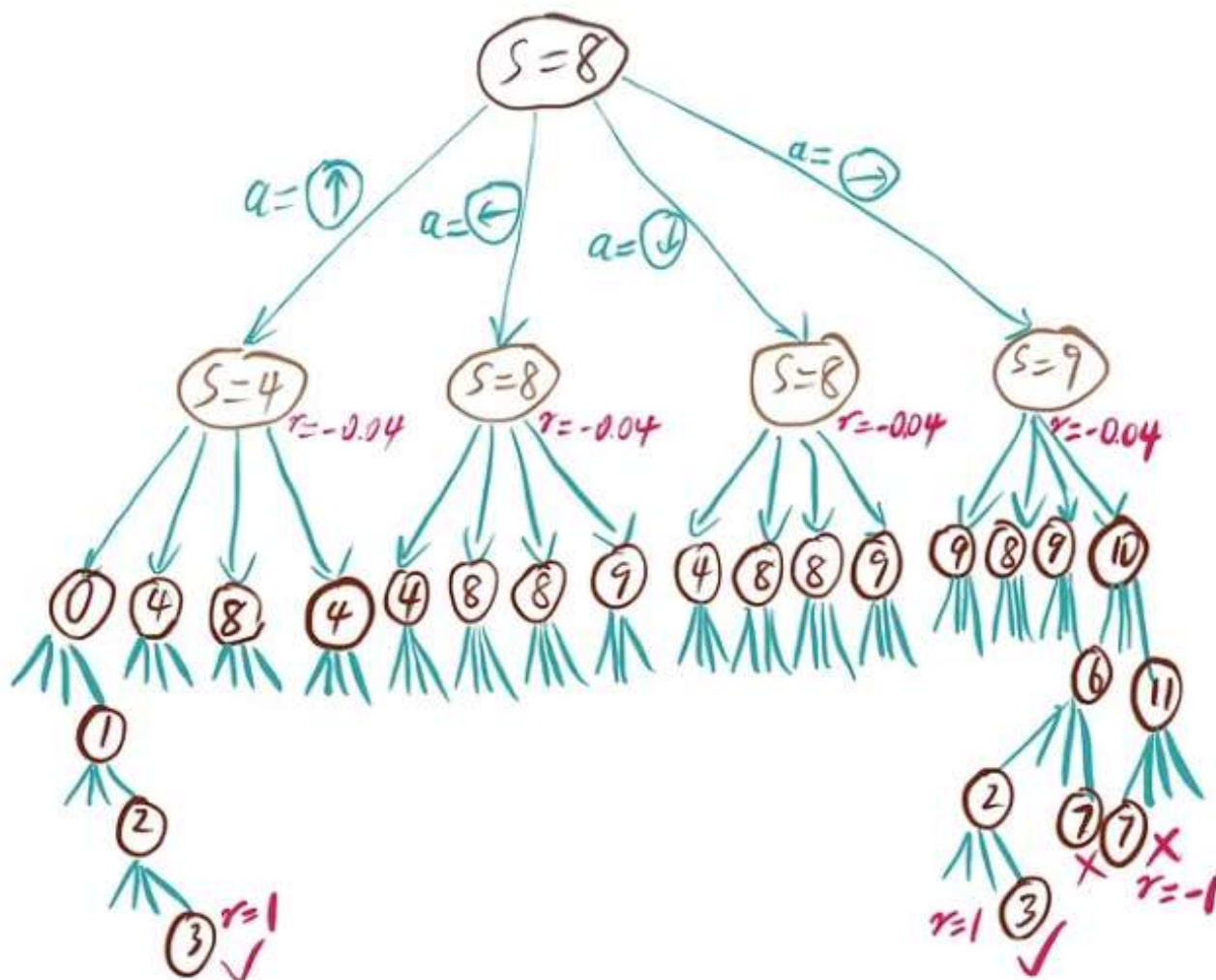
...

$$P(s' = 9 | s = 10, a = \text{UP}) = 0.1$$

$$P(s' = 10 | s = 10, a = \text{UP}) = 0$$

$$P(s' = 11 | s = 10, a = \text{UP}) = 0.1$$





State and Reward

$s=0$ $r=-0.04$	$s=1$ $r=-0.04$	$s=2$ $r=-0.04$	$s=3$ $r=1.0$
$s=4$ $r=-0.04$	$s=5$ $r=\text{nan}$	$s=6$ $r=-0.04$	$s=7$ $r=-1.0$
$s=8$ $r=-0.04$	$s=9$ $r=-0.04$	$s=10$ $r=-0.04$	$s=11$ $r=-0.04$

