

## TME SOLO - Groupe 1

### Consignes :

1. Créez une branche dans votre **github** pour le TME solo, nommée `tmesolo`, et basculez dessus : `git branch tmesolo`, puis `git checkout tmesolo`.
2. En fin d'épreuve, **créez un fichier** `explications.txt`, **ajoutez-le** au dépôt (`git add explications.txt`) et indiquez-y :
  - toute explication utile pour comprendre les stratégies implémentées ;
  - les modifications notables faites vis à vis de votre travail sur le projet foot.Enfin, **mettez à jour** votre dépôt distant.
3. Renseignez le nom de votre compte **github** dans ce formulaire.
4. Vous êtes encouragés à utiliser les outils implémentés lors des semaines passées sur le projet foot (`superstate`, `actions`, etc) et à les étoffer pour les adapter au sujet.

---

### Exercice 1 (20 points) – Volleyball

---

Le jeu a été modifié pour simuler un jeu de volleyball. Les joueurs ne peuvent se déplacer que sur leur moitié de terrain car ils ne peuvent pas franchir le filet.

Une équipe marque des points si :

- le ballon s'arrête sur la moitié de terrain adverse (`vitesse < settings.BALL_STOP`) ;
- l'équipe adverse fait sortir la balle des limites du terrain ;
- l'équipe adverse touche le ballon plus de 3 fois d'affilée.

A l'engagement, la balle est donnée aléatoirement au joueur 0 d'une des deux équipes. Celui-ci doit immédiatement tirer dans la balle ou on considère que celle-ci tombe au sol et le point va à l'autre équipe.

Un exemple de jeu est donné dans `examples/volleyball_example.py`.

**Q 1.1** Implémenter une stratégie `Echauffement` qui renvoie la balle à un adversaire de l'autre côté du terrain.

Créer, à la racine de votre dépôt, un fichier `volleyball_question1.py` pour y faire jouer deux joueurs avec cette stratégie.

**Q 1.2** Implémenter une stratégie `Attaque` qui envoie la balle le plus loin possible de l'adversaire (attention à ne pas la faire sortir du terrain). Il est possible pour le joueur de se faire la passe à lui-même.

Créer, à la racine de votre dépôt, un fichier `volleyball_question2.py` pour illustrer le fonctionnement de cette stratégie et expliquer ce dernier dans `explications.txt`.

**Q 1.3** Implémenter une stratégie `Defense` qui réagit adéquatement à l'attaque précédente. Créer, à la racine de votre dépôt, un fichier `volleyball_question3.py` pour y faire jouer deux joueurs, l'un en attaque, l'autre en défense, et expliquer cette dernière stratégie dans `explications.txt`.

**Q 1.4** Implémenter une stratégie 1 vs 1 à partir des questions précédentes.

Créer, à la racine de votre dépôt, un fichier `volleyball_question4.py` pour illustrer le fonctionnement de cette stratégie et expliquer ce dernier dans `explications.txt`.

**Q 1.5** Implémenter une (ou plusieurs) stratégies 2 vs 2 de votre choix.

Créer, à la racine de votre dépôt, un fichier `volleyball_question5.py` pour illustrer votre travail et l'expliquer dans `explications.txt`.

**Q 1.6 BONUS** Faire une recherche sur des paramètres d'une de vos stratégies grâce à un algorithme de recherche au choix parmi ceux implémentés pendant les séances.

Créer, à la racine de votre dépôt, un fichier `volleyball_question6.py` pour illustrer votre travail et l'expliquer dans `explications.txt`.