

Une coupe du monde aléatoire

Le but de ce premier travail est de simuler à l'aide de Python une coupe du monde de football qui serait totalement aléatoire.

Le fichier `liste_équipes.csv` contient la liste des équipes qui participent à cette coupe du monde.

Vous aurez à choisir comment répartir votre code Python dans différents fichiers pour que l'utilisation de vos programmes soit la plus commode possible.

1. Dans la première phase de la coupe, les équipes sont réparties en 8 groupes de 4 équipes, numérotés de la lettre A à la lettre H. À partir du fichier de la liste des équipes, effectuer un tirage qui donne la composition des différents groupes dans 8 fichiers :

```
équipes_groupe_A.csv
équipes_groupe_B.csv
```

```
:
```

```
équipes_groupe_H.csv
```

On pourra utiliser la fonction `shuffle` du module `random` qui permet de mélanger une liste.

2. Dans chaque groupe, chaque équipe doit affronter une fois toutes les autres. Pour simuler un match entre deux équipes, on procède de manière simpliste : on tire pour chaque équipe un entier aléatoire entre 0 et 3 qui correspond au nombre de buts marqués par l'équipe. Écrire le code permettant de créer 8 fichiers contenant les résultats des matchs de chaque groupe :

```
matchs_groupe_A.csv
matchs_groupe_B.csv
```

```
:
```

```
matchs_groupe_H.csv
```

Dans chaque fichier, une ligne doit correspondre à un résultat de match. Par exemple, la ligne :

```
Allemagne,2,Angleterre,1
```

indique que dans le match entre l'Allemagne et l'Angleterre, l'Allemagne a marqué 2 buts et l'Angleterre 1.

3. Une fois les matchs de groupes simulés, on peut compter les points de chaque équipe. Chaque match donne un certain nombre de points : 3 pour une victoire, 1 pour un match nul, 0 pour une défaite. Écrire un code qui lit les fichiers de résultats des matchs et qui calcule un nombre de points pour chaque équipe dans 8 fichiers :

```
points_groupe_A.csv
points_groupe_B.csv
```

```
:
```

```
points_groupe_H.csv
```

4. Les équipes qualifiées pour la suite de la compétition sont les deux premières de chaque groupe au classement par points. En cas d'égalité de points entre deux équipes, les règles pour les départager sont très faciles à retrouver sur le web. Écrire un code permettant d'obtenir dans un fichier `.csv` la liste des 16 équipes qualifiées, avec leur groupe initial et leur classement.

5. On souhaite ensuite simuler les matchs de la phase finale de la compétition, depuis les huitièmes jusqu'à la finale. Les règles pour déterminer quels seront les matchs, comme par exemple « premier du groupe A contre deuxième du groupe B », devront être cherchées sur le web. Un match de la phase finale ne peut pas se terminer par un nul, donc on procède de manière encore plus simpliste que précédemment pour le simuler : on tire le vainqueur à pile ou face. Écrire un code permettant de simuler tous les matchs de la phase finale, en sauvegardant les résultats de chaque tour dans des fichiers `.csv` :

```
résultats_huitièmes.csv
```

```
résultats_quarts.csv
```

```
résultats_demi.csv
```

```
résultats_3e_place.csv
```

```
résultats_finale.csv
```

Dans chaque fichier, une ligne doit correspondre à un résultat de match. Par exemple, la ligne :

```
Allemagne,perdu,Angleterre,gagné
```

indique que c'est l'Angleterre qui a gagné le match.

6. Écrire le code Python qui « englobe » tout le travail effectué, permettant de passer directement de la liste des équipes de départ au vainqueur de la coupe.

7. Question ouverte : comment pourrait-on améliorer la simulation des matchs pour obtenir des résultats plus réalistes ? Proposer des idées et les implémenter en Python.