

Commentaires DM - Belote en OCaml

- Eviter de compliquer le code pour le plaisir
- Bien respecter les signatures des fonctions à écrire
- Ne pas oublier de tester vos fonctions avec assert (comme indiqué par l'énoncé)

Q1 - `est_valide` : le typage assure que la couleur est Pique, Coeur ... et que la hauteur est As, Roi ... ou Petite of int. Il suffit de vérifier que, dans le cas Petite, la valeur est entre 7 et 10

Q2 - `print_carte` : la fonction doit afficher (et non renvoyer une chaîne de caractères)

Q3 - `points` : pour le Valet et le 9, il y a 2 cas

Q4 - `compte_points` : on utilise une référence pour sommer le nombre total de points

Q5 - `prend_la_main` :

- lorsque les couleurs sont identiques, comparer le nombre de points ne suffit pas.

Par exemple le 7 et le 8 valent tous les deux 0 points. Pourtant le 8 doit l'emporter.

Voir le corrigé, en utilisant un "rang" d'une carte (selon qu'il s'agit d'un atout ou non)

Q6 - `tour` : on appelle 3 fois la fonction précédente, en se souvenant de l'indice de la carte qui l'emporte

Q7 - `genere_jeu` : il faut prévoir la taille du tableau résultat à l'avance. Il faut renvoyer le tableau.

Q8 - `melange_jeu` : on peut utiliser `Random.self_init` pour initialiser aléatoirement le générateur de nombres pseudo-aléatoires (HP)

Q9 - affichage final : on appelle successivement : `genere_jeu`, `melange_jeu` et `print_carte`