

une classe simple : les polynomes

- ↪ Définir une classe `Polynome` avec deux attributs : un entier `deg` et un tableau de `deg` entiers `coeff` et les constructeurs et accesseurs vous semblant utiles.
- ↪ Ecrire une méthode d'instance `afficher(Polynome p)` qui affiche le polynome.
- ↪ Ecrire une méthode d'instance `add(Polynome p)` qui renvoie le polynome somme de l'objet et du paramètre `p`.
- ↪ Ecrire une méthode d'instance `derive` qui renvoie le polynome dérivé de l'objet.
- ↪ Tester votre classe dans une classe `TestPolynomes` avec une méthode `main` qui manipulera des polynomes (création, addition, dérivation, affichage)

le jeu de 421

Le jeu de 421 est un jeu de dés qui se joue avec 3 dés. Le 1er joueur démarre la partie en lançant les trois dés il peut alors choisir de relancer ou non. S'il relance, il peut choisir de relancer tout ou une partie des 3 dés; il fera 3 lancers maximum. Les joueurs suivants devront effectuer le même nombre de lancers que le 1er joueur.

Le but est de marquer le plus de points selon le barème suivant

figure	points
421	10
111	7
xxx ou 11x ($x \neq 1$)	x
tierce	2
221	2
autre	1

les classes pour le 421

Pour chaque classe vous écrirez les constructeurs et accesseurs nécessaires.

↪ Définir une classe `De` avec 1 attribut de type `byte` (Utiliser la classe `Random` pour créer ou changer la valeur d'un dé)

↪ Définir une classe `TroisDes` avec 1 attribut de type tableau de `De` de longueur 3

↪ Définir une classe `Joueur` avec un attribut de type `String` `nom`, 1 attribut de type `TroisDes` et d'autres attributs à définir pour que le joueur puisse conserver son score et sa décision de relancer ou pas les dés issus d'un précédent lancer.

↪ Définir une classe `Jeu` avec 2 joueurs en attributs, un constructeur à deux paramètres `String` pour les noms des 2 joueurs, une méthode `jouer` qui gère une partie en 10 tours et qui renvoie le vainqueur.

↪ Dans une classe `JouerAu421`, écrire la méthode `main` qui saisit les noms de deux joueurs et qui leur fait faire une partie en 10 tours.