Reflexion

Les Classes :

* Quete : permet de séparer les différentes données des quetes dans des champs

@champ : chNumero, chPos, chPrecond, chDuree, chExp, chIntitule

@methode : Quete, extraitPrecond, extraitPos, toString, getNumero, getPos,

getPrecond, getDuree, getExp, getIntitule

* Scenario : TreeMap stockant les quête que l’on ajoute via méthode ajout

@champ : chTreeMapQuete

@methode : ajoutQuete

* LectureFichierQuete : Permet de lire les différents fichiers contenant les quêtes

@methode : lecture

* Solution : Permet de trouver l’ordre des quêtes en fonction de la solution voulue

@champ: chScenario, chJoueur

@methode: efficace, exhaustive, speedrun

* Joueur: Commence à 0, et stocke l’xp accumulé, la position actuelle, les quete déjà

accompli et le temps de jeu accumulé par le joueur pour un scénario

@champ : xp, position, temps de jeu, queteAccompli

@méthode : ajoutXp, ajoutDuree, majPosition, ajoutQuete, getPosJ, getExpJ,

getDureeJ, getQueteAcc

* Client : Permet de lancer le programme

@méthode : main

Precondition :

* [1,0,0,0] il faut faire la quete 1
* [0,0,0,0] pas de precondition
* [1,0,5,0] il faut faire la quete 1 et 5
* [1,2,5,0] il faut faire la quete (1 ou 2) et la quete 5
* [1,2,5,6] il faut faire la quete (1 ou 2) et la quete (5 ou 6)