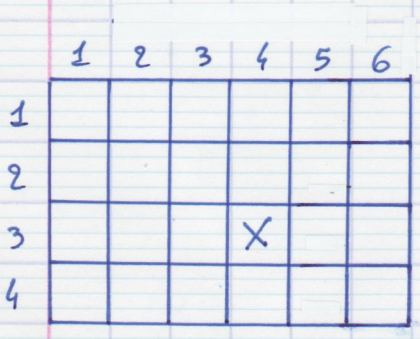
Chasse au Trésor

Voici une démarche possible de conception d'un programme afin de résoudre un problème posé.

I) Schématiser le problème



① On "cache" un trésor dans une grille

② Le joueur donne les coordonnées d'une case :

* Ou bien c'est la case du trésor et c'est gagné.
* Ou bien le programme renvoie une des 8 directions :

haut/gauche - haut - haut/droit - droit
bas/droit - bas - bas/gauche - gauche

Le joueur a ainsi une indication pour son prochain coup.

II) Représenter les informations

① Une matrice nommée grille de nb_lignes lignes et nb_colonnes colonnes. Cette matrice représente le plateau de jeu et ne contient que des zéros.

② Une case contient un "1" (la case du trésor)

Cette case est repérée par un couple (ligne, colonne) d'entiers générés aléatoirement

③ Deux variables entières x et y qui représentent l'abscisse et l'ordonnée proposées par le joueur.

④ Une variable entière score qui représente le nombre de "coups" (de propositions) du joueur.

III) Affichage à l'écran

Seules deux choses s'afficheront à l'écran :

① Le plateau de jeu

Prévoir une fonction d'affichage du plateau qui permette de visualiser les numéros de lignes et de colonnes

② Les messages à l'utilisateur

Fonctions print() et input(). Attention au type de données renvoyées par input().

IV) Rôle des fonctions.

① verif_coor(x, y) qui vérifie si les coordonnées proposées par l'utilisateur sont possibles.
x et y sont des entiers ; verif_coor(x, y) renvoie un booléen.

② compare_coor(x, y) qui compare les coordonnées proposées par l'utilisateur à celles du trésor.
x et y sont des entiers ; compar_coor(x, y) renvoie une chaine de caractères (haut, bas, trouve, ...)

③ jeu() lance et execute le jeu.
Cette fonction execute les autres fonctions, interagit avec l'utilisateur et affiche le plateau jusqu'à ce que le joueur trouve le trésor.

IV) Répartition du travail.

A faire !