TD n°1 d'algorithmique : Programmer un jeu sur la calculatrice

La question posée étant, comment programmer un jeu sur la calculatrice, on va essayer d'y répondre dans cette feuille. Choisir un des jeux proposés ici (ou un autre) sachant qu'ils sont classés par ordre de difficulté (à programmer) croissante. Ceux que la question ne concerne ou n'intéresse pas peuvent toujours finir, par exemple la feuille de révisions sur les équations.

1. Devine un nombre

Le plus simple des jeux : la calculatrice choisit un nombre entre deux bornes, par exemple entre 1 et 100, et le joueur doit deviner ce nombre. Pour cela, il propose un nombre et la calculatrice répond « plus » ou « moins ». Elle compte le nombre d'essais et, quand le joueur a trouvé la bonne réponse, elle affiche le nombre d'essais.

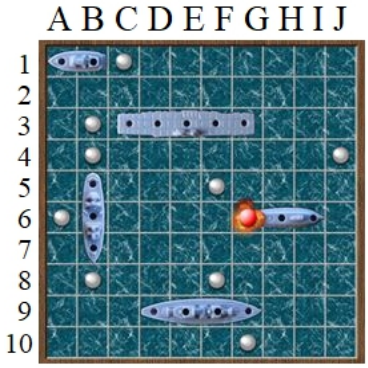
🗝 Pour programmer ce jeu, il suffit d'alterner les propositions du joueur et les affichages des réponses de la calculatrice « plus » ou « moins ». Le test d'arrêt de la boucle est évident puisque le jeu s'arrête quand le joueur a trouvé le nombre.

2. Entraînement aux tables

La calculatrice choisit deux nombres entre deux bornes, par exemple 2 et 10, et demande le produit de ces deux nombres. Si le résultat entré par le joueur est correct, elle propose un autre produit. Sinon, elle repose la question. Dans tous les cas, le nombre de bonnes réponses est enregistré ainsi que le nombre de question. On peut afficher le pourcentage de bonnes réponses avant chaque nouvelle question.

🗝 Pour programmer ce jeu, il suffit de prévoir l'interaction entre la calculatrice (qui pose les questions et affiche le score du joueur) et le joueur. La calculatrice formule sa question en affichant le produit des deux nombres choisis et attend la réponse. Cela peut se faire avec l'instruction *n*=*int*(*input*("{}x{}=".format(*a*,*b*))) qui va mettre votre réponse dans la mémoire *n* pour le traitement. Il faut prévoir un test d'arrêt pour éviter la boucle infinie : par exemple, tant que le nombre de questions est inférieur à 20 *ou* le tau de bonnes réponses inférieur à 90%.

3. Bataille navale



La calculatrice choisit des emplacements pour 5 bateaux dont les longueurs sont connues par le joueur (2, 3, 3, 4 et 5) et le joueur propose successivement le lieu à bombarder repéré par sa colonne (entre A et J) et sa ligne (entre 1 et 10). La calculatrice répond en affichant un point rouge si un bateau est touché, un point blanc si le bombardement a raté les cibles. Lorsqu'un bateau est coulé (tous ses points sont touchés), la calculatrice doit le signaler, par exemple en entourant les points touchés par un rectangle. Le but du jeu est de couler tous les bateaux ennemis.

🗝 Pour programmer ce jeu, il faut prévoir un affichage graphique (module kandinsky) ; les saisies du joueur sont des séquences de 2 caractères comme B7 ou H2 et les réponses de la calculatrice sont données par l'affichage qui est mis à jour. Au départ, il faut dessiner une grille 10×10 avec les indications des lettres et chiffres. Je rappelle que l'écran de la Numworks est composé de 320 pixels sur sa longueur et de 222 pixels sur sa hauteur. On peut fixer la couleur de chacun des pixels en exécutant l'instruction set_pixel(x,y,color), où color est une couleur du système RGB. Par exemple, c1=color(255,255,255) fixe la couleur c1 à blanc et c2=color(255,0,0) fixe la couleur c2 à rouge.

4. Master mind

La calculatrice choisit une combinaison de couleurs pour 4 emplacements, les couleurs étant choisies parmi 6 couleurs possibles : jaune, bleu, rouge, vert, blanc et noir. Des variantes plus difficiles existent (possibilité d'ajouter des couleurs, d'utiliser des espaces vides comme couleur, ajout d'un 5ème emplacement, etc .)
À chaque tour, le joueur propose une combinaison de pions de couleur et la calculatrice lui indique :

- le nombre de pions de la *bonne couleur bien placés* en utilisant le même nombre de *fiches noires*
- le nombre de pions de la *bonne couleur*, mais *mal placés*, avec les *fiches blanches*

Les tours successifs (dix coups maximum) restent affichés de manière à pouvoir servir de base à la réflexion du joueur, le but étant de trouver la combinaison en un minimum de coups.

🗝 Pour programmer ce jeu, il faut donc prévoir un affichage graphique (module kandinsky) ; les saisies du joueur doivent être codifiées, par exemple avec des chiffres : jaune=1, bleu=2, rouge=3, vert=4, blanc=5 et noir=6. Le premier coup du joueur de l'illustration est donc 3245. La calculatrice doit afficher des disques colorés aux bons emplacements ainsi que les fiches (noires ou blanches) sur le côté. Prévoir donc une fonction *disque*(*c*,*x*,*y*) qui affiche un disque de la couleur *c* centré en un point de coordonnées (*x*;*y*).