

Exemples de projets et conseils

La notation au bac ne se fait pas sur la difficulté des sujets réalisés, mais sur votre maîtrise de ce que vous avez fait (dans le respect du cahier des charges au minimum).

1. Le Jeu de la vie

Joli sujet, surtout avec une interface graphique. Peut se faire en programmation objet

Extrait de WIKIPEDIA http://fr.wikipedia.org/wiki/Jeu_de_la_vie

En préambule, il faut préciser que le jeu de la vie n'est pas vraiment un jeu au sens ludique, puisqu'il ne nécessite aucun joueur ; il s'agit d'un automate cellulaire, un modèle où chaque état conduit mécaniquement à l'état suivant à partir de règles préétablies.

Le jeu se déroule sur une grille à deux dimensions, théoriquement infinie (mais de longueur et de largeur finies et plus ou moins grandes dans la pratique), dont les cases — qu'on appelle des « cellules », par analogie avec les cellules vivantes — peuvent prendre deux états distincts : « vivantes » ou « mortes ».

À chaque étape, l'évolution d'une cellule est entièrement déterminée par l'état de ses huit voisines de la façon suivante :

- Une cellule morte possédant exactement trois voisines vivantes devient vivante (elle naît)
- Une cellule vivante possédant deux ou trois voisines vivantes le reste, sinon elle meurt

Le sujet du projet

Cahier des charges de base

- A partir d'un état donné de l'automate, calculer l'état suivant
- Programmation d'une interface
 - Saisir un état
 - Afficher un état
 - Afficher un certain nombre d'états successifs à partir d'un état initial

Pour aller plus loin

- Faire fonctionner une grille $n \times m$ à la façon d'un tore (on peut penser à une chambre à air comme représentation d'un tore)
 - Tout ce qui disparaît par le bas réapparaît par le haut et inversement
 - Tout ce qui disparaît par la gauche réapparaît par la droite et inversement

2. Le compte est bon

Sujet avec une interface graphique minimale au cas où. Sujet pour ceux qui aiment réfléchir, les algorithmes de résolution sont de ceux qui apportent le plus au niveau programmation, parmi tous les sujets proposés ici. Peut permettre un approfondissement sur la récursivité (non obligatoire).

Règles du jeu

- La donne :
 - Il y a 24 plaques comportant chacune un nombre entier positif
 - Deux plaques pour chaque entier de l'ensemble $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
 - Une plaque pour chaque entier de l'ensemble $\{25, 50, 75, 100\}$
 - On tire aléatoirement, avec équiprobabilité et sans remise, 6 des 28 plaques
 - On tire aléatoirement, avec équiprobabilité, un entier n dans l'intervalle $[100 ; 999]$
- L'objectif :
 - Obtenir n comme résultat d'une suite d'opérations arithmétiques, ou sinon approcher n au plus près, en combinant certaines des 6 plaques tirées et au moyen des 4 opérations $+$, $-$, \times et \div .
- Règles à observer :
 - On ne peut utiliser une plaque qu'une fois, mais on n'est pas obligé de les utiliser toutes
 - On ne peut utiliser la division, que lorsqu'elle "tombe juste", et le résultat à prendre en compte est le quotient
 - Les entiers négatifs sont interdits

Le sujet du projet

Cahier des charges de base

- Programmer les tirages aléatoires
 - Une équiprobabilité "approchée" suffirait
 - Mais tirer avec équiprobabilité dans un ensemble un certain nombre de ses éléments est un exercice amusant
- Programmer la recherche du bon compte
 - On peut utiliser un algorithme trouvé sur internet pour résoudre le problème arithmétique
- programmation d'une interface
 - Proposer un choix pour la détermination des données
 - Tirage aléatoire
 - Saisie console (utile pour la mise au point)
 - Afficher le résultat, accompagnée d'une suite valide d'opérations sur les données conduisant à ce résultat
 - Proposer un nouvel essai

Pour aller plus loin

- Concevoir un algorithme résolvant le problème en utilisant le plus petit nombre de plaques possible

3. Le tic-tac-toe

4. Sujet très classique. L'intelligence artificielle est obligatoire.

Règles du jeu

- Deux joueurs impriment à tour de rôle leur marque, en général un rond pour l'un et une croix pour l'autre, dans les cases au départ vierges d'un plateau 3×3 , c'est-à-dire de 3 lignes et 3 colonnes
- A gagné le premier joueur qui réussit à aligner 3 de ses marques, en ligne, en colonne ou en diagonale
- Le jeu peut se terminer sur un match nul

Le sujet du projet

Cahier des charges de base

- Vérifier que la pose d'une marque est valide
- Déterminer la fin de partie
- Faire jouer contre l'ordinateur, l'ordinateur joue aléatoirement
- Faire jouer contre l'ordinateur, avec une intelligence artificielle
- Programmation d'une interface
 - Saisir un coup
 - Afficher le plateau

Pour aller plus loin

- Permettre à deux joueurs de jouer plusieurs parties d'affilée l'un contre l'autre en mémorisant les points gagnés par chacun des adversaires

5. Bataille navale

6. Sujet très classique, plus sympa avec une interface. Moins simple qu'il n'y paraît.

Règles du jeu

- Classiques, à choisir parmi les diverses variantes usuelles, voir sur internet si vous n'avez jamais joué à la bataille navale ☺

Le sujet du projet

Cahier des charges de base

- Générer aléatoirement un plateau initial, c'est-à-dire une disposition initiale des navires d'une flotte
- Vérifier qu'un plateau initial saisi est valide (nombre de navires de chaque catégorie et disposition disjointe des navires). Les navires ne doivent pas se toucher.
- Mémoriser les tirs successifs en tenant à jour le plateau courant
- Déterminer l'impact d'un tir sur le plateau courant
 - Coup dans l'eau
 - Navire touché
 - Navire coulé
- Calculer un tir aléatoire sur une des cases vierges du plateau courant

- Déterminer la fin de la partie
- Programmation d'une interface
 - Saisir un plateau initial
 - Saisir un tir
 - Afficher un plateau
- Faire jouer une partie humain contre ordinateur idiot

Pour aller plus loin

- Faire jouer une partie humain contre ordinateur intelligent

7. Algorithme de Dijkstra : calcul d'un trajet le plus court possible par un GPS

Facile mais pas fascinant.

Présentation du problème

On se donne un graphe orienté dont les arcs sont valués, c'est-à-dire qu'à chaque arc est affectée une valeur numérique. L'exemple le plus simple est celui d'une carte sur laquelle figurent des distances entre les villes. Le but de l'algorithme de Dijkstra est de trouver le chemin le plus court entre le sommet dit « de départ » et le sommet dit « d'arrivée ».

Cet algorithme sert de base pour les algorithmes de calcul de temps de trajet avec les GPS (en théorie, je pense qu'en pratique c'est plus sophistiqué)

Le sujet du projet

Cahier des charges de base

- Définir une structure de donnée permettant de représenter un graphe valué aux arcs
- Implémenter l'algorithme de Dijkstra pour trouver le chemin le plus court entre le sommet de départ et le sommet d'arrivée.
- Programmation d'une interface
 - Saisir un graphe valué aux arcs
 - Afficher un graphe valué aux arcs
 - Afficher les distances d'un sommet à tous ses descendants
- Modifier l'algorithme afin de calculer un plus long chemin d'un sommet à un autre, donner un ou plusieurs exemples de l'intérêt de trouver le chemin le plus long.

8. Générateur d'exercices de calcul mental

Plus sympa avec une interface graphique bien sûr.

Présentation du problème.

Le but est de créer un programme permettant de tester les capacités en calcul mental d'un utilisateur, sur des entiers naturels. La difficulté est donnée par la taille des entiers et la nature des opérations.

Le sujet du projet

Cahier des charges de base

- Ecrire un programme qui génère aléatoirement un opérateur (les quatre opérations usuelles) et deux entiers. Les divisions doivent « tomber juste », les soustractions doivent donner un résultat positif.
- Ecrire un programme vérifiant que la réponse donnée par l'utilisateur est juste.
- Ecrire un programme permettant de choisir la bonne réponse parmi 4, une seule étant juste, les autres étant aléatoires (version QCM)
- Ecrire un programme permettant de faire plusieurs calculs et de donner un score en fonction de la difficulté
- Programmation d'une interface :
 - Choix de la difficulté (ou absence de choix)
 - Choix de la version : QCM ou non.
 - Affichage des questions, des réponses et du score final

Pour aller plus loin

- Dans la version QCM, donner la réponse juste et trois réponses plausibles.

9. Calcul en valeur exacte sur des fractions

La partie « pour aller plus loin » est la bienvenue, sinon le projet est un peu juste. Le professeur vous expliquera ce que sont les fractions continues.

Présentation du problème.

Le but est de créer un programme permettant de faire du calcul sur des fractions, en donnant les résultats sous forme de fraction réduite.

Le sujet du projet

Cahier des charges de base

- Proposer un type de donnée pour représenter une fraction.
- Ecrire un programme qui étant donné deux fractions rentrées sous la forme ci-dessus, et un opérateur, renvoie le résultat.
- Ecrire un programme utilisant la méthode de Héron pour calculer la valeur approchée d'une racine carrée sous forme de fraction, la précision étant donnée.
http://fr.wikipedia.org/wiki/M%C3%A9thode_de_H%C3%A9ron
- Programmation d'une interface :
 - Rentrer au clavier une opération à faire, composée soit de deux fractions séparées par un opérateur (sans espaces), soit d'un formalisme à définir pour le calcul d'une valeur approchée de la racine carrée.
 - Affichage du résultat

Pour aller plus loin

- Compléter par un programme donnant l'écriture en fraction continue d'un nombre rationnel, ainsi que l'interface correspondante

10. Algorithme de Pledge (version virtuelle)

L'algorithme est simple à programmer. Par contre l'interface graphique donne plusieurs challenges

Présentation du problème.

L'algorithme de Pledge permet de sortir d'un labyrinthe, lorsque l'on part d'un point quelconque à l'intérieur. Un labyrinthe peut toujours être représenté sous forme de graphe, il est donc assez simple de programmer cet algorithme.

http://interstices.info/jcms/c_46065/l-algorithme-de-pledge

http://en.wikipedia.org/wiki/Maze_solving_algorithm#Pledge_algorithm

Le sujet du projet

Cahier des charges de base

- Programmer l'algorithme de Pledge étant donné un graphe représentant un labyrinthe, on considérera que tous les murs sont à angle droit.
- Interface :
 - Etant donné un graphe représentant un labyrinthe, donner le sommet représentant la position initiale ainsi que le sommet représentant la sortie
 - Afficher la suite des directions permettant de sortir du labyrinthe

Pour aller plus loin

- Interface graphique :
 - Afficher le labyrinthe et la trajectoire
- Programmer l'algorithme « suivre le mur »
- Interface :
 - Rentrer un graphe donnant un labyrinthe

11. Stéganographie et traitement des images

12. L'algorithme est assez simple à programmer, une fois qu'on l'a compris... Très beau sujet.

Présentation du problème.

La stéganographie est « l'art de la dissimulation ». Il s'agit de cacher une image dans une autre (ou un son dans un autre etc...). Soit pour jouer aux espions, soit pour mettre un copyright sur une image

Le sujet du projet

Cahier des charges de base

- proposer un algorithme qui, étant donné deux images dans un format simple non compressé, l'une étant « lourde » et l'autre « légère » (en taille), cache l'image légère dans la lourde. On utilisera la technique du bit de poids faible (cf <http://fr.wikipedia.org/wiki/St%C3%A9ganographie>). On réfléchira sur la taille relative des images.
- Il est impératif de travailler à partir des outils proposés par le professeurs (programme « utilitaires images » et format ppm de préférence). Sinon cela devient vite compliqué. Par ailleurs, travailler

dans un format compressé (comme le jpg) détruit l'image cachée.

- Proposer un autre algorithme qui extrait une image cachée dans une autre (ie l'opération inverse du précédent)

Pour aller plus loin

- compléter par d'autres effets sur une image (par exemple le contour avec des masques sophistiqués).

13. Cryptanalyse du chiffre de Vigenère

Pas d'interface graphique, sinon minimale

Présentation du problème.

Le chiffre de Vigenère est une méthode pour rendre un message secret. Elle a été popularisée par Blaise de Vigenère, diplomate, cryptographe, alchimiste et astrologue français du 16^{ème} siècle. Le véritable inventeur en est Giovan Battista Bellaso, cryptographe du 16^{ème} siècle également, dont on ne sait presque rien. Cette méthode a été utilisée pendant plus de trois siècles.

Voir http://fr.wikipedia.org/wiki/Chiffre_de_Vigen%C3%A8re

Le sujet du projet

Cahier des charges de base

- Ecrire un premier programme réalisant le chiffrement de Vigenère d'un texte en langue française écrit en majuscules, sans signe diacritique ni espace (c'est-à-dire n'utilisant que les lettres majuscules non accentuées de l'alphabet, et rien d'autre), en utilisant une clef donnée par l'utilisateur sous la même forme
- Ecrire un second programme réalisant le déchiffrement d'un texte crypté par le premier programme, en utilisant la clef ayant servi au chiffrement
- Ecrire un troisième programme effectuant la cryptanalyse d'un texte crypté par le premier programme, sans la connaissance de la clef de cryptage, en utilisant la méthode de l'indice de coïncidence décrite ici : http://fr.wikipedia.org/wiki/Indice_de_co%C3%AFncidence , ainsi que dans l'article sur le chiffre de Vigenère.
- Programmation d'une interface pour chacun des 3 programmes :
 - Saisir un texte et sa clef de chiffrement pour les 2 premiers programmes et sans clef pour le troisième
 - Afficher le texte résultant pour les 2 premiers programmes et un ensemble de textes possibles pour le troisième
 - Réaliser une interface unique pour les 3 programmes

Pour aller plus loin :

- Ecrire un troisième programme effectuant la cryptanalyse d'un texte crypté par le premier programme, sans la connaissance de la clef de cryptage, en utilisant la méthode de Kasiski et Babbage décrite ici : http://fr.wikipedia.org/wiki/Cryptanalyse_du_chiffre_de_Vigen%C3%A8re
- Ecrire un programme permettant transformant un texte standard en texte sans signes diacritiques.

14. Snake (interface graphique obligatoire)

Se prête bien à la programmation objet. Difficulté de l'interface graphique et de la gestion du temps

Présentation du problème.

Le but est de programmer le jeu classique du serpent. Ce projet me paraît un peu plus difficile que d'autres.

Le sujet du projet

Cahier des charges de base

- Écrire le jeu du serpent avec une interface graphique, et les fruits de base, si nécessaire en programmation objet.

Pour aller plus loin

- Rajouter les fruits qui font changer de sens, ainsi que d'autres effets amusants (vitesse par exemple)

15. Casse-brique (interface graphique obligatoire)

Se prête bien à la programmation objet. Difficulté de l'interface graphique et de la gestion du temps

Présentation du problème.

Le but est de programmer le jeu classique du casse-brique. Ce projet me paraît un peu plus difficile que

d'autres.

Le sujet du projet

Cahier des charges de base

- Écrire le jeu du casse-brique avec une interface graphique, si nécessaire en programmation objet.

Pour aller plus loin

- Rajouter des effets sur certaines briques (taille de la raquette, accélération, ralentissement de la balle)

16. 2048

Nettement plus joli avec une interface graphique, mais ici ce n'est pas la principale difficulté, qui vient de spécificités du jeu.

Présentation du problème.

Le but est de programmer le jeu « 2048 ».

Le sujet du projet

Cahier des charges de base

- Écrire le jeu 2048... Pour ceux qui ne connaissent pas, tester le jeu sur téléphone portable avant. Les règles sont très simples

Pour aller plus loin

- Jouer avec des images plutôt que des nombres, sur interface graphique.

17. Tétris

Tout est dur dans ce projet : l'algorithme, la gestion du temps et donc l'interface graphique. A commencer de préférence directement en programmation objet

18. Tower défense

Du style minimaliste, avec un seul type de tours et une seule vague d'ennemis. A commencer de préférence directement en programmation objet. Peut-être approfondi avec plusieurs vagues/types d'ennemis/tours. Interface graphique obligatoire bien sûr.

19. Robotique : les applaudissements aux concerts

Très beau sujet, original.

Présentation du problème.

Lors d'un concert ou d'un spectacle, vous avez tous constaté qu'une foule peut se coordonner pour applaudir en rythme. L'idée de ce projet est de modéliser le phénomène avec arduino, et des signaux lumineux plutôt que sonores. On pourra utiliser une dizaine de modules arduino, chacun représentant une personne. Ce projet fait partie des problèmes dits « d'intelligence collective ».

Le sujet du projet

Cahier des charges de base

- Chaque arduino doit avoir le même programme.
- Programmer une génération d'un signal lumineux aléatoire périodique.
- Acquérir les intensités émises régulièrement (par exemple 10 fois par seconde), sur un temps assez long.
- Extraire les deux signaux les plus « proches », en tenant compte des possibles variations de lecture de la lumière, et de leur période inconnue a priori
- Coordonner le signal émis avec ces deux signaux reçus, en faisant la moyenne.
- Recommencer indéfiniment les trois étapes précédentes

Pour aller plus loin

- Rajouter une perturbation aléatoire après un certain temps, lorsque tous les signaux sont coordonnés

20. Robotique : algorithme de Pledge

Le labyrinthe sera représenté par les chemins et non par les murs. En effet il existe déjà un programme permettant de suivre des traces noires sur un fond blanc.

Présentation du problème.

L'algorithme de Pledge permet de sortir d'un labyrinthe, lorsque l'on part d'un point quelconque à l'intérieur.

http://interstices.info/jcms/c_46065/l-algorithme-de-pledge
http://en.wikipedia.org/wiki/Maze_solving_algorithm#Pledge_algorithm

Pour que ce projet soit plus simple, les murs seront représentés par des surfaces noires au sol. Le robot roulera sur le bord de ces surfaces, ce qui est bien plus facile que de rester à distance constante d'un mur.

Le sujet du projet

Cahier des charges de base

- Comprendre le fonctionnement du robot arduino : servomoteurs, boussole, comment suivre une ligne.
<http://arduino.cc/en/Guide/Robot> pour les moteurs et la boussole, ainsi que le tutoriel « line following ». Voir aussi la commande « updateIR »
- Programmer l'algorithme de Pledge sur le robot arduino (votre honoré professeur ou bien le lycée vous fournira les planches pour faire le sol du labyrinthe, ainsi que du scotch noir)

Pour aller plus loin

- Choisir aléatoirement si l'on « longe » les murs par la droite ou la gauche
- Rajouter des indicateurs (diodes) pour les actions en cours

21. Robot télécommandé

Présentation du problème.

Comme son nom l'indique, le but est de commander un robot à distance.

Le sujet du projet

Cahier des charges de base

- Comprendre le fonctionnement du robot parallax/arduino : servomoteurs, et éventuellement compléments.
<http://learn.parallax.com/ShieldRobot> pour la partie servomoteurs, en comprendre l'essentiel dans le chapitre 2
- Comprendre le protocole de communication xbee et le logiciel X-CTU (livre du professeur, en particulier p 467)
- Réaliser un « mappage » qui permet de passer des deux axes x/y du joystick Tinkerkit, aux deux moteurs droite/gauche du robot.
- Programmer le robot et la télécommande afin qu'il communiquent

Pour aller plus loin

- Rajouter une sécurité avec les détecteurs de contact, pour que le robot ne heurte pas les obstacles.

22. Logisim

Présentation du problème.

Le but est la création d'une UAL 2 ou 3 bits, 4 opérations, sur papier, à l'aide du logiciel Logisim.

Le sujet du projet

Sera détaillé si un groupe désire faire ce projet

23. Robotique et réseaux : afficheur météo : problème de commande du shield lcd.

Présentation du problème.

Le but est de créer un gadget connecté qui donne des informations récupérées sur le web. Ce genre d'objet est sans fil, mais on ne transmettra pas les données sans fil dans la version de base. Attention sur ce projet, l'aide du professeur sera minimale, vu les faibles compétences de celui-ci en réseaux...

Le sujet du projet

Cahier des charges de base

- Comprendre le fonctionnement du module lcd <http://www.tinkerkit.com/lcd/>
- Comprendre le shield ethernet (livre du professeur)
- Comprendre comment on récupère des données sur le web (livre du professeur)
- Afficher des données météo d'un endroit spécifié.

Pour aller plus loin

- Rajouter une alarme « vous avez reçu un message » en surveillant une boîte e-mail.
- Rajouter une connexion xbee sans fil