

Projets d'assembleur 2018

Le projet est à rendre avant le lundi 7 janvier 2019 . Il contiendra :

1) Un rapport au format PDF dans lequel on retrouvera

- En première page :

- les noms et prénoms des étudiants concernés,
- la présentation du sujet choisit,
- le nom du fichier qui contient le programme source,
- les consignes pour le fonctionnement de votre projet.
- les algorithmes mis en œuvre, les difficultés rencontrées,
- une conclusion dans laquelle vous évoquerez les améliorations que vous auriez pu ajouter,
- en annexe, le code commenté de votre programme.

2) Le programme lui-même (ainsi que les différents fichiers bibliothèque) sous format “zip” dans un dossier portant votre(vos) nom(s) à déposer sous moodle.

La soutenance du projet aura lieu en fin de première semaine de Janvier 2019.

Remarques : Pour tous les sujets qui suivent, il y a différents niveaux de difficultés. Vous essayerez donc pour les sujets « plus simples » de soigner la présentation et de l'améliorer.

Ainsi, vous pouvez ajouter une petite introduction de type « générique » de présentation du projet.

Pour les sujets de type « jeux » vous pouvez ajouter des notions de scores (introduire dans certains cas la notion de temps) sauvegarder les meilleurs scores etc...

Important : Avant de penser aux améliorations, n'oubliez pas de faire une version qui fonctionne sans artifices

« Jeux »

1. Drapeau anglais : Sur une grille 3x3, deux joueurs placent des symboles (X ou O). Le premier qui aligne 3 symboles a gagné.

- Amélioration : implanter un algorithme qui permet à la machine de jouer.

2. Le Jeu de la Vie : Le mathématicien CONWAY a proposé un jeu dit « jeu de la vie » en considérant des cellules susceptibles de se reproduire, de disparaître ou de survivre lorsqu'elles obéissent à certaines règles. Ces cellules sont représentées par des éléments sur une grille dont la taille est arbitraire. Chaque cellule est entourée de huit cases, susceptibles de contenir des cellules.

- Les règles sont les suivantes :

o La survie : Chaque cellule ayant deux ou trois cellules adjacentes survie jusqu'à la génération suivante.

o La mort : Chaque cellule ayant plus de trois cellules adjacentes meurt pour cause de surpopulation, celles ayant une ou aucune cellule adjacente meurt d'isolement.

o La Naissance : Chaque emplacement adjacent à exactement trois cellules fait naître une nouvelle cellule pour la génération suivante.

3. Le Mastermind : La machine « choisit » une composition de 5 symboles pris parmi 8. L'utilisateur fait des propositions de combinaisons, et à chaque fois, le programme indiquera le nombre de symboles bien placés et de symboles mal placés.

4. Jeu de Nim : 20 allumettes sont alignées. À tour de rôle, deux joueurs prennent 1, 2 ou 3 allumettes. Celui qui prend la dernière a gagné.

- Variante : Plusieurs lignes de plusieurs allumettes.

5. Échiquier : Placer huit reines sur un échiquier, sans qu'aucune d'entre elles ne puissent en prendre une autre.

6. Le moulin : Dans une grille 3x3, on labellise les cases du pourtour avec les lettres «LE MOULIN». Des pions portant les lettres « L,E,M,O,U,L,I,N » sont placés de façon arbitraire sur la grille (il y a donc une case vide). Le but du jeu est de replacer les lettres sur les cases portant les mêmes indications. Le programme doit afficher les différentes étapes. Pour cela, il faut respecter les règles de déplacement suivantes :

- Les pions en milieu de ligne ne peuvent se déplacer que dans les coins.
- Les pions en coin peuvent se déplacer dans toutes les cases adjacentes.
- Le pion du milieu ne peut se déplacer que dans les coins.

7. Sur une grille, deux joueurs jouent à la marelle (« morpion »). La machine compte les points.

8. Sur un damier, deux joueurs jouent aux dames. La machine vérifie qu'aucun d'entre eux ne trichent.

9. Réaliser un mini logiciel de DAO (permettre à l'utilisateur de dessiner avec la souris) avec les fonctionnalités : crayon et gomme

- Amélioration : cercle, rectangle, nouveau dessin, taille de crayon, pot de peinture, etc...

10. Pousse Pousse (ou "puzzle"): dans une grille comportant un trou on doit déplacer les pièces pour refaire une figure (ou ordre) pré-établi. Exemple : la grille 3x3 contient les chiffres de 1 à 8 et on doit remettre les chiffres dans l'ordre.

- Amélioration : grille plus grande, mélange, dessin

5	3	1
2	7	
8	6	4

1	2	3
4	5	6
7	8	

11. Puissance 4 : Dans une grille de hauteur 6 et de largeur 7, deux joueurs font tomber alternativement des jetons (un symbole par joueur). Celui qui réussit à en aligner 4 a gagné.

12. “Claquer“ la bulle : Réaliser un “jeu“ qui consiste à faire disparaître une bulle (cercle) qui apparaît de façon aléatoire à l’écran

- Amélioration : score, vitesse, plusieurs bulles.

13. Faire la simulation d’un jeu de squash

- Amélioration : score, obstacle, vitesse, “casse-briques“

14. Afficher un labyrinthe 2D à l’écran et permettre de se déplacer à l’intérieur pour trouver la sortie

- Amélioration : différents niveaux, temps, génération de labyrinthe,...

15. Faire la simulation d’un jeu de dé (type 421) Lancer de 3 dés, possibilité de relancer une partie de ceux-ci.

- Amélioration : distributions de jetons, animation des dés, autres jeux de dés

16. Le jeu démineur : Dans le jeu Démineur il faut déminer tout un territoire en cliquant sur les cases. Pour la première case c'est du hasard mais pour les autres on peut deviner ce qui se cache sous les cases.

- Amélioration : temps, score, aide

17. Le jeu de la bataille Navale : Le but de la bataille navale est de couler tous les bateaux de son adversaire. Dans un premier temps, on place ses bateaux sur le plateau. Votre adversaire sans voir votre placement place aussi ses propres bateaux. Ensuite, chacun à son tour, vous essayez de trouver et couler les bateaux de l'adversaire.

- Amélioration : L’adversaire peut être l’ordinateur, stratégie.

« Gestion de nombres »

1. Afficher une montre avec des chiffres 7 segments (similaire à ceux de la «console hardware»).

- Amélioration : mise à l’heure, jour/mois, fonctionnalités d’un réveil

2. Écrire un programme qui permet à l’utilisateur de demander le résultat de la somme ou de la différence de deux horaires (exprimés en heures, minutes, secondes).

- Amélioration : prévoir différents formats d'écriture, ajouter les jours (mois ! !- années ! !)

3. Écrire un programme permettant la gestion de nombres au format de 64 bits

4. Écrire un programme permettant l'apprentissage des opérations classiques (addition, soustraction, multiplication, division). L'utilisateur donne les informations et le programme affiche les opérations (comme vous les avez vues en primaire)

5. Écrire un programme qui permet d'effectuer des conversions de nombres dans des bases données par l'utilisateur.

« Gestion de chaînes »

1. L'utilisateur entre un texte, puis il demande de remplacer une lettre par une autre (on vérifiera la substitution lors de l'affichage)

- Amélioration : remplacer un mot (portion de texte) par un autre.

2. Écrire un nombre entier donné par l'utilisateur, en chiffres romains

- Afficher la valeur décimale d'un nombre donné en chiffres romains.

3. Permettre à l'utilisateur de saisir un nombre compris entre -10 et +10 millions et afficher ce nombre au format monétaire (ex : -524,15 € — 1 145 786,53 €) les nombres seront toujours arrondis à deux décimales (ex : 6.256 s'affichera 6,26 €).

- Amélioration : prévoir les calculs de budgets (somme, différence)

4. Le programme permet à la machine d'afficher, à chaque fois, un nombre entier qui contient de plus en plus de chiffres, puis l'efface. L'utilisateur doit retaper le nombre. Le programme s'arrêtera à la première erreur.

5. Faire la simulation d'un message publicitaire avec défilement.

6. Écrire un nombre donné par l'utilisateur en toutes lettres.

7. Recherche d'un mot dans une phrase : L'utilisateur saisie un mot. Si ce mot existe dans la phrase alors on affiche "trouvé" sinon on affiche "non trouvé". On peut améliorer ce projet en recherchant un mot dans un texte.