

Projet SNAKE

Projet informatique - IN3001

Ce projet consiste à réaliser un jeu Snake. Pour rappel, ce jeu 2D consiste à diriger un serpent en cherchant à collecter des objets qui apparaissent sur la zone de jeu afin de le faire grandir. A chaque fois que le joueur collecte un objet il gagne des points et le serpent s'allonge d'une unité.

Le serpent avance en continu, le joueur ne peut que lui indiquer une direction à suivre (haut, bas, gauche ou droite). Attention, le serpent ne doit pas toucher les murs délimitant la zone de jeu ni son propre corps sous peine de perdre la partie. Il est donc interdit de faire un demi-tour brutal. En figure 1 vous pouvez voir un exemple d'apparence de la zone de jeu d'un snake sous un terminal.

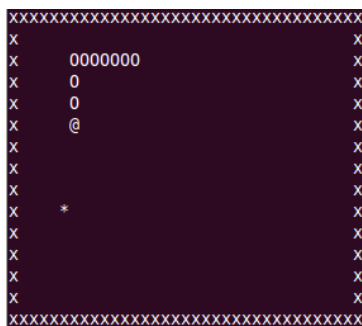


FIGURE 1 – Exemple de snake

Dans cet exemple la zone de jeu est délimitée par des caractères 'x', le caractère '@' représente la tête du serpent tandis que les caractères 'O' son corps. Le caractère '*' représente l'objet à collecter afin de faire grandir le serpent.

Certains d'entre vous ont peut être déjà identifié une difficulté pour créer un jeu de cette nature sur un terminal : lors de la récupération d'un ou plusieurs caractère(s) tapé(s) au clavier, l'utilisateur doit impérativement appuyer sur la touche entrée pour valider le ou les caractère(s) tapé(s) à l'écran. Ce qui est bien entendu incompatible avec un jeu qui se déroule en "temps-réel". Vous n'aurez pas à écrire la partie de code pour résoudre ce problème. Deux fichiers vous sont fournis "gestion_clavier.c" et "gestion_clavier.h", vous pourrez les trouver l'espace ecampus 2 du cours. Vous devrez cependant déclarer les macro-constantes définissant le choix de vos touches du clavier pour diriger le serpent (ne pas utiliser les touches directionnelles sauf si vous aimez les challenges bien sûr ;-).

Version minimale du jeu à fournir :

Le jeu devra être exécutable en "mode console" (pas d'utilisation de bibliothèque graphique). Un menu doit être proposé à l'utilisateur afin de : débuter une partie, consulter les 5 derniers meilleurs scores (pseudo + score) et mettre fin au programme. Les scores doivent être sauvegardés entre plusieurs parties (après un arrêt du programme). Pendant une partie, le joueur doit pouvoir voir son score (en cours) et la touche permettant de quitter sa partie prématurément pour revenir au menu.

Contraintes au niveau du programme source :

- Vos devez programmer de façon modulaire en regroupant les fonctions qui appartiennent à différents thèmes dans des fichiers sources différents ;
- La taille de la zone de jeu doit pouvoir être facilement paramétrable et de façon statique (i.e. imposé au moment de la compilation) au minimum. Pour ceux qui veulent aller plus loin le faire en dynamique (paramétrable dans les options / menu du jeu).

Quelques idées d'améliorations :

Voici quelques idées d'améliorations par rapport à la version minimale à fournir :

- Proposer des modes de difficulté différents en modulant la vitesse du serpent (macro-constante `DELAY_KEY_DETECTION` du fichier "gestion_clavier.h"). Augmenter le nombre de points gagnés suivant le mode de difficulté ;
- Placer des obstacles dans la zone de jeu. On pourrait imaginer un système de niveaux au cours d'une partie ;
- Insérer des objets malus/bonus qui apparaissent à interval régulier pendant une durée limitée (type trou de ver, téléporteur ou soustrayant d'une unité la taille du serpent) ;