PROGRAMMATION EN C

NGA BESSALA

PREPAS INTERNATIONAL

GROUPE IV

**Dossier de Conception - Jeu Snake**

**Sommaire**

1. **Introduction**
   * Présentation du jeu
   * Objectif du projet
   * Public cible
2. **Architecture du projet**
   * Principe du jeu
   * Objectif du joueur
   * Règles et mécanique
   * Gestion des collisions
   * Conditions de victoire et défaite
3. **Interfaces utilisateur**
   * Écran d’accueil
   * Écran de jeu
   * Affichage du score
   * Écran de fin de partie
4. **Modules et Algorithmes**
   * Description des modules
   * Gestion des fichiers
5. **Sécurité et Bonnes Pratiques**
   * Sécurisation des données
   * Gestion des erreurs
   * Tests et validation
6. **Conclusion**

**1. Introduction**

**1.1 Présentation du Jeu**

Le jeu **Snake** est un classique des jeux d'arcade où le joueur contrôle un serpent qui se déplace sur une grille. L'objectif est de manger des objets (nourriture) pour grandir tout en évitant de toucher les murs ou son propre corps. Plus le joueur mange, plus le serpent s'allonge et le jeu devient difficile.

**1.2 Objectif du Projet**

L'objectif de ce projet est de développer une application en langage C qui permet aux utilisateurs de jouer au jeu Snake. L'application doit permettre :

* L'authentification des joueurs
* L'ajout, la suppression, la modification et la consultation des informations sur les parties jouées
* L'enregistrement des scores et des données de jeu dans un fichier externe
* La gestion sécurisée des mots de passe
* La gestion des niveaux de difficulté

**1.3 Public Cible**

Ce jeu est destiné à toute personne intéressée par les jeux rétro et souhaitant tester ses réflexes. Il peut aussi servir de projet pédagogique pour les étudiants apprenant le langage C.

**1.4 Technologies et Bibliothèques Utilisées**

* **Langage** : C
* **Compilateur** : GCC (via l'invite de commande ou un IDE comme Code::Blocks)
* **Bibliothèques** :
  + stdio.h : Entrée/sortie standard
  + stdlib.h : Fonctions standards (ex: génération de nombres aléatoires)
  + conio.h : Gestion des entrées clavier (Windows uniquement)
  + windows.h : Fonctions spécifiques à Windows pour l'affichage et les délais
  + time.h : Gestion du temps (délais et randomisation)
* **Gestion des fichiers** :
  + fopen, fprintf, fscanf pour l'enregistrement des scores
* **Sécurité** : Cryptage basique des mots de passe

**2. Architecture du Projet**

**2.1 Principe du Jeu**

Le joueur déplace un serpent qui doit manger de la nourriture pour grandir. À chaque fois qu'il mange, le score augmente. Le jeu se termine si le serpent touche un mur ou son propre corps.

**2.2 Objectif du Joueur**

* Manger le plus de nourriture possible pour atteindre le score le plus élevé.
* Éviter de toucher les murs ou son propre corps.

**2.3 Règles et Mécanique**

* Le serpent se déplace en ligne droite dans la direction choisie par le joueur.
* La nourriture apparaît à une position aléatoire.
* Lorsque le serpent mange, il grandit et la vitesse du jeu peut augmenter.
* Le joueur ne peut pas faire de demi-tour immédiat (ex: aller à gauche s'il allait à droite).

**2.4 Gestion des Collisions**

* Collision avec les murs : Game Over
* Collision avec le propre corps du serpent : Game Over
* Collision avec la nourriture : Augmentation du score et du corps du serpent

**2.5 Conditions de Victoire et Défaite**

* Pas de condition de victoire (jeu infini)
* Défaite si collision avec un mur ou soi-même

**3. Interfaces Utilisateur**

**3.1 Écran d’Accueil**

* Affichage du titre
* Options : Jouer, Voir les scores, Quitter

**3.2 Écran de Jeu**

* Affichage du plateau
* Position du serpent et de la nourriture
* Score en temps réel

**3.3 Affichage du Score**

* Score mis à jour en temps réel
* Sauvegarde en fin de partie

**3.4 Écran de Fin de Partie**

* Message "Game Over"
* Score final
* Option : Rejouer ou Quitter

**4. Modules et Algorithmes**

**4.1 Description des Modules**

| **Module** | **Description** |
| --- | --- |
| **Main** | Initialise le jeu, gère la boucle principale, coordonne les autres modules |
| **Snake** | Gère la position et la direction du serpent, vérifie les collisions |
| **Affichage** | Affiche le plateau, le serpent, la nourriture et le score |
| **Entrées Clavier** | Capture les touches du joueur et met à jour la direction |
| **Gestion des Scores** | Enregistre et charge les scores à partir d'un fichier |

**5. Sécurité et Bonnes Pratiques**

**5.1 Sécurisation des Données**

* Stockage des mots de passe sous forme de hash
* Validation des entrées utilisateurs

**5.2 Gestion des Erreurs**

* Vérification des fichiers avant lecture
* Gestion des erreurs d'entrée utilisateur

**5.3 Tests et Validation**

* Tests unitaires sur chaque module
* Simulation de différents scénarios de jeu

**6. Conclusion**

Ce document présente la conception détaillée du projet Snake. Chaque module est clairement défini pour assurer un développement efficace. Le jeu respectera les bonnes pratiques de programmation en C pour garantir une expérience fluide et stable.