

CROQUE CAROTTE

Logique du jeu :

- Initialiser plateau (tableau)
- Piocher une carte (nombre aléatoire qui choisit une carte)
- faire avancer le lapin (changer l'état du lapin dans la structure)
- faire tourner la carotte (Faire une rotation aléatoire de l'état des trous)
- Mélanger les cartes si besoin
- vérifier s'il y'a un gagnant
- Changer de joueur
- Premier arrivé en haut

Dans toutes les fonctions doivent figurer des "sécurités" ne permettant pas au joueur de saisir n'importe quelle valeur et de faire planter le programme...

Fonctions d'affichage :

Plateau de 25 cases (tableau prenant diff valeurs) :

Trou = -1 → ooo

Case vide = 0 → _

Lapin 1 = 1 → L1



Lapin 2 = 2 → L2

On affiche ensuite en fonction de la valeur de la case un caractère

Première approche de la fonction main (modèle) :

Type int (Renvoie qlq chose)

Type void (Ne renvoie rien)

Besoin de pointeurs

-Void Init affichage + plateau (mettre en place les 3 trous)

While (nbr de lapins des 2 j >= 4 ET que on a pas WIN){

-int Select_card : (Pioche aléatoirement ds le tab/Retire la carte tirée du tab/Renvoie la valeur d'une carte)

-move_rabbit (modifier la case du lapin et/ou la structure lapin/Choisir lapin/Saut de lapin?/)

-turn_carrot (modifier rotation cases (trous))

-delete_rabbit(Remplacer la case par un trou et "éliminer" le lapin)

-shuffle_cards (mélanger cartes si besoin)

-int win_check (Renvoie True ou False dans win)

-next_player (change de joueur, y'a t-il besoin d'une fonction ?)

On quitte le while }

On regarde quel lapin est sur la 25e case et on détermine le gagnant et le perdant

Types :

Structure → Lapin

Tableau d'entiers → Plateau

Tableau d'entiers → 6x4(1,2,3,clic) = 24 cartes

Liens utiles :

Booléens →

<https://openclassrooms.com/en/courses/900279-apprenez-a-programmer-avec-ada/893006-les-conditions-ii-les-booleens>

POINT AU 7/03/2020:

-Initialiser structures

-Faire avancer les lapins

-Faire un menu (switch); ok

-Boucle par tour + vérifier et afficher gagnant dans le main();(si pos = arrivée -->gagnant ok

-Sauvegarde;

BORDEL dans les cases rotatives qui varient au cours du jeu ???-OK

Retirer les lignes inutiles remplacées par maj plateau → Pas de maj plateau...bug

POINT AU 25/04/2020:

-Sauvegarde (récup pioche)

-Pb ds tourner carotte avec case 0-OK

-Revoir la pertinence et l'utilité des commentaires pour un code compréhensible

-Boucle infinie ap le mélange de la pioche- PRESQUE OK

-Bcp trop de cartes carotte (0 réattribué à la dernière case pq ?)

-Lorsque tous les lapins sont morts, l'affichage du trou ne s'est pas fait-OK

-Remplacer init_structures par init_lapins ???-OK

-Case Carotte s'est ouverte → Modulo cases rotatives

-Si plus qu'un lapin et qu'on pioche une carte le faisant avancer plus loin que le plateau, on ne pourra pas sélectionner de lapin (la règle n'est pas précisée)

-Revoir sélection lapin avec nouvelle fonction avancer

-OUBLI sauvegarde cases rotatives dans la sauvegarde

~~-Structure pour cases rotatives ?~~

JOUR J :

Rendre → Code + Slides + Agenda/Hiérarchie/Document explicatif

Présenter ce qui a été fait :

-Cahier des charges

-Suppléments

-Architecture projet

Les bugs

Difficultés (structures) / (pointeurs) / Modulos / Config

RÉORGANISATION DE AVANCER :

CAS 1 : case vide/carotte

- avance
- case compte

CAS 2 : trou

- avance
- case compte

CAS 3 : joueur

- avance
- case compte pas

30/04 :

- sécurité sauvegarde (on laisse tomber pas très utile au final)
- Tous les lapins meurent d'un coup-OK

COULEURS ?

structures dans les prototypes de fonction ?

On peut piocher 25 cartes ??

+++ : Affichage/règles du jeu/demander noms/sécu sauvegarde?/