Travail 1 (10%) – Équipe 2-3

Serpents et échelles

# Description

Vous allez programmer un jeu de type Serpents et Échelles.

Il existe plusieurs types de cases :

* Case de base : lorsqu’un joueur arrive sur une case de base, sa position ne change pas.
* Case échelle : lorsqu’un joueur arrive sur une case « échelle », sa position doit être augmentée. (1 pt)
* Case serpent : lorsqu’un joueur arrive sur une case « serpent », sa position doit être diminuée. (1 pt)
* Case trappe : lorsqu’un joueur arrive sur une case « trappe », sa position doit être diminuée d’un nombre de cases au hasard entre 5 et 10. (1 pt)
* Case saut : lorsqu’un joueur arrive sur une case « saut », sa position doit être augmentée d’un nombre de cases au hasard entre 1 et 3. (1 pt)
* Chaque type de case doit s’afficher différemment. (1 pt)
* Chaque case possède un attribut de type Points.

Il existe plusieurs types de points :

* Points de base : donne un nombre de points égal à sa valeur.
* Points négatifs : retire un nombre de points égal à sa valeur. (1 pt)
* Point quitte ou double : donne aléatoirement 0 ou 2 fois sa valeur en points. (1 pt)

À gauche de la planche de jeu, vous devez :

* Afficher le joueur courant. (1 pt)
* Afficher les 2 joueurs ainsi que leurs points à l’aide d’un contrôle utilisateur. (2 pts)
* Avoir un bouton pour lancer un dé 6. (1 pt)
  + Lorsqu’un joueur lance le dé, sa position doit être augmentée de la valeur du dé.

Événement :

* Toute mise à jour de l’affichage doit être déclenchée par un événement. (3 pts)

Planche de jeu :

* Vous devez initialiser la planche de jeu au hasard avec un « [seed](https://learn.microsoft.com/en-us/dotnet/api/system.random.-ctor?view=net-6.0) ». (1 pt)
  + Vous devez avoir au moins 4 cases de chaque type non basique.
  + Donner une valeur de déplacement aléatoire aux cases échelle et serpent.
* Lorsqu’ un joueur arrive sur une case, il gagne les points associés à la case. (1 pt)
* Lorsqu’ un joueur arrive sur une case, il se déplace selon le type de case. (1 pt)

Fin du jeu :

* Le jeu se termine lorsqu’un joueur arrive sur la dernière case ou la dépasse. (1 pt)
* Lorsque le jeu se termine, le joueur qui a le plus de points gagne. (1 pt)
* Le gagnant est affiché dans un MessageBox. (1 pt)
* Le programme doit se fermer à la fermeture du MessageBox. (1 pt)

Début du jeu (3 pts) :

* L’application doit ouvrir sur une Form menu qui
  + Permet d’entrer les noms des 2 joueurs.
  + Possède un bouton « Jouer » qui démarre la partie.

GitHub (3 pts):

* Vous devez avoir au moins une « Pull request » regroupant plusieurs « commits » par membre de l’équipe.
  + Dans un cas extrême où un membre n’aurait aucune « Pull request » ou que ses « Pull request » ne fassent rien, la note 0 pourrait être obtenue pour cette personne.
* Tout changement apporté à la branche master/main autre que le « commit » initial doit passer par une « Pull request » et être approuvé par les autres membres de l’équipe.

# Modalité de remise

* La remise se fait par Github. Vous devez inviter **elacassecegepba** comme collaborateur à votre répertoire. Je vais prendre le code se trouvant dans la branche master/main.
* Les modalités d’intégrité intellectuelle et plagiat s’appliquent.
* Votre programme doit compiler (30% de pénalité)