

SAÉ S1.02 : comparaison d'approches algorithmiques

Le Snake autonome : version 4

Votre mission

Voici les nouvelles consignes pour votre quatrième version du programme de snake autonome.

Objectif

Il s'agit maintenant de faire cohabiter 2 serpents sur le plateau : *serpent1* et *serpent2*. Chaque serpent devra essayer d'être le plus rapide pour manger chaque pomme avant l'autre serpent. Les deux serpents avanceront d'une position à tour de rôle. Enfin, au cours de ses déplacements un serpent devra éviter l'autre (pas de collision entre serpents, pas de « croisement » de serpents).

Positions initiales

Les deux serpents seront initialement positionnés comme ceci :

- *serpent1* : X=40 et Y=13 et orienté vers la droite,
- *serpent2* : X=40 et Y=27 et orienté vers la gauche.

Nouvelles positions des pommes

```
int lesPommesX[NB_POMMES] = {40, 75, 78, 2, 9, 78, 74, 2, 72, 5};  
int lesPommesY[NB_POMMES] = {20, 38, 2, 2, 5, 38, 32, 38, 32, 2};
```

Nouvelles positions des pavés

```
int lesPavesX[NB_PAVES] = { 4, 73, 4, 73, 38, 38};  
int lesPavesY[NB_PAVES] = { 4, 4, 33, 33, 14, 22};
```

Représentation graphique

Afin de bien distinguer les deux serpents à l'exécution, la tête du *serpent1* sera représentée par le caractère '1', et celle du *serpent2* sera représentée par le caractère '2'.

Affichage des résultats

En fin d'exécution, votre programme devra afficher pour chaque serpent le nombre de déplacements effectués et le nombre de pommes mangées. Exemple :

```
Serpent1 : 365 déplacements et 6 pommes mangées  
Serpent2 : 348 déplacements et 4 pommes mangées
```

Consignes de programmation

Vous duplierez votre procédure `progresser()` en deux procédures :

- `progresser1()`, qui calculera la prochaine position du *serpent1*,
- et `progresser2()` qui calculera celle du *serpent2*.

Note : vous duplierez de la même façon toute fonction personnelle qui interviendrait dans le calcul des déplacements du serpent.

Travail demandé

Complétez et adaptez votre programme de manière à ce qu'il permette à deux serpents de manger les pommes de manière concurrentielle.

Déposer sur Moodle, avant la date convenue, votre code source "version4.c".

Note : n'oubliez pas de respecter les **conventions de codage** vues en R1.01 (voir document sur Moodle) : commentaires, indentation, utilisation de constantes, nommage des variables, etc.