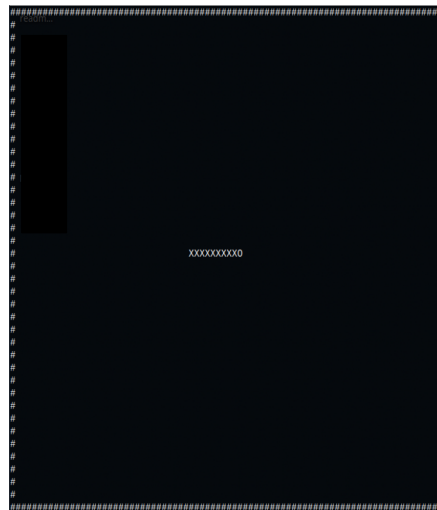


## SAÉ S1.01 : implémentation d'un besoin client

### Le jeu du Snake : version 3

#### Votre mission

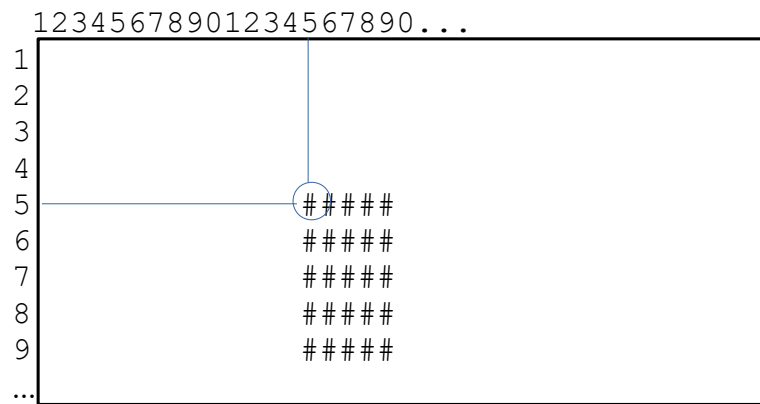
Dans cette troisième version du programme, le serpent se déplacera à l'intérieur d'un plateau délimité par une bordure constituée de caractères `#`.



**Note** : le coin supérieur gauche de la bordure a pour position (1,1).

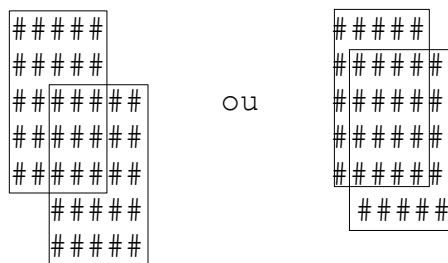
- Le plateau aura une de taille 80 caractères de large sur 40 caractères de haut. Le serpent ne pourra pas sortir de ce cadre. S'il entre en collision avec un élément de la bordure, le jeu se terminera (fin d'exécution).
- Le serpent a toujours une taille de 10, mais sa tête sera positionnée initialement en (40,20). Les autres éléments seront positionnés à sa gauche car le serpent se dirigera initialement vers la droite.
- À l'intérieur du cadre, vous placerez 4 pavés. Un pavé est un carré de taille 5 composé de caractères `X`. Ces pavés seront placés à des positions définies de manière aléatoire.  
**Attention** : un pavé ne devra jamais être "collé" à une bordure.

Exemple de pavé positionné en (15,5), qui est la position de son coin supérieur gauche :



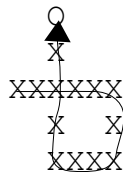
Bien sûr, si le serpent rentre en collision avec un élément de pavé, le jeu se terminera (fin d'exécution).

Les pavés pourront éventuellement se chevaucher, voire se superposer. Exemples :



Mais en aucun cas les pavés ne devront "sortir" du plateau. Un pavé doit intégralement être contenu à l'intérieur du plateau, et non collé à une bordure.

- Dorénavant, le serpent ne pourra plus se croiser. Si sa tête entre en collision avec une partie de son corps (un de ses anneaux), le jeu se terminera (fin d'exécution).



Ce mouvement sera désormais interdit.

Adaptez votre programme de manière à respecter ces nouvelles exigences. Vous déposerez sur Moodle, avant la date convenue, votre code source "version3.c" ainsi qu'un cahier de tests complet que vous établirez.

## Consignes

- Vous écrirez une procédure `initPlateau(...)` qui initialise le plateau de jeu (cf. "détails techniques").
- Vous écrirez une procédure `dessinerPlateau(...)` qui affiche le plateau de jeu sur l'écran préalablement effacé.
- Votre procédure `progresser(...)` devra posséder un paramètre supplémentaire en sortie : un booléen dont la valeur sera `true` si une collision a été détectée, contre un élément de la bordure du plateau ou contre un pavé ou bien contre un anneau du serpent.

## Détails techniques

- Le plateau sera représenté par un tableau à deux dimensions de caractères. Il sera initialisé avec des caractères ``#'` pour la bordure et les pavés, et avec des caractères `ESPACE` pour tout le reste.
- Les tableaux à deux dimensions en langage C (ou "matrices")

Exemple : tableau de taille 5x5

	0	1	2	3	4
0					
1					
2					
3					
4					

### *Déclarations*

- déclaration d'une variable de type tableau de 5 fois 5 entiers :  
`int tab[5][5];`
- déclaration d'un tableau de 50 fois 30 réels :  
`float tab[50][30];`

### *Accès aux éléments d'un tableau*

- Chaque élément du tableau est accessible via deux indices, son indice de ligne et son indice de colonne :  
`int element;`  
`element = tab[0][3];`