

Jeu de Catégorie 2 « DéplacerPièces »

1. Description du jeu

On dispose d'un tableau de 10 cases, numérotées de 0 à 9.

Ces cases peuvent contenir des pièces, éventuellement empilées (cf. Figure 1).

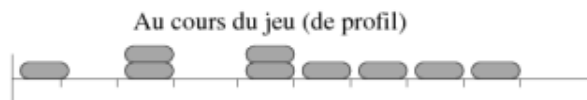


Figure 1 : Vue profil du tableau en cours de jeu : chaque case peut contenir 0 ou plusieurs pièces.

Début et fin de partie (cf Figure 2) :

- Au départ, le tableau ne contient qu'une seule pièce, en case d'indice 1
- En fin de partie, le tableau contient une seule pièce, en case d'indice 7

Etat initial	Tableau	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>
	Indices	<div>0123456789</div>
Etat final	Tableau	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>
	Indices	<div>0123456789</div>

Figure 2 : Etats initial et final du jeu

Actions possibles (cf Figure 3) :

- De l'état **A** vers l'état **B** : l'action Ajouter (des pièces) à partir de la case d'indice **i** a pour effet :
 - D'enlever 1 pièce à la case d'indice **i**
 - D'ajouter 1 pièce à chacune des cases **i-1** et **i+1**
 Les conditions de réalisation sont :
 - La case **i** a bien deux cases adjacentes **i-1** et **i+1**
 - La case **i** contient au moins 1 pièce
- De l'état **B** vers l'état **A** : l'action Supprimer (des pièces) à partir de la case d'indice **i** a pour effet :
 - D'enlever 1 pièce à chacune des cases **i-1** et **i+1**
 - D'ajouter 1 pièce à la case **i**
 Les conditions de réalisation sont :
 - La case **i** a bien deux cases adjacentes **i-1** et **i+1**
 - Les cases **i-1** et **i+1** contiennent chacune au moins 1 pièce

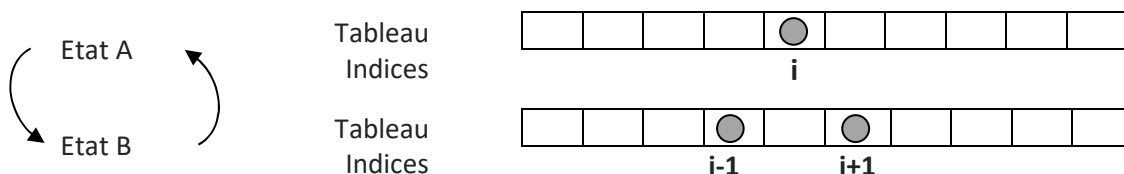


Figure 3 : Actions : Ajouter, **i** (**A** → **B**) et Supprimer, **i** (**B** → **A**)

Le Jeu

Le jeu se joue à un seul joueur.

Le tableau est présenté au joueur dans son état initial.

Le joueur doit faire passer la pièce de la case d'indice 1 à la case d'indice 7 en un minimum d'actions (Ajouter/Supprimer).

Le jeu se termine lorsque le tableau atteint l'état final, ou bien par Abandon du joueur.

2. Spécification du besoin

On demande d'afficher

- Les règles du jeu avant de commencer
- Le tableau à l'état initial
- L'état du tableau et le nombre de coups joués à chaque coup
- En fin de jeu :
 - Un message de félicitations en cas de réussite
 - Un message indiquant l'abandon en cas d'abandonDans les 2 cas, le nombre total de coups joués, et le nombre minimal de coups nécessaires (18) à la réalisation de cette réussite

Exemples de comportements possibles

Scénario : abandon en cours de jeu, avec erreurs de saisie

```
Le plateau contient un tableau de 10 cases, numerotees de 0 a 9.
La case d'indice 1 contient 1 piece de monnaie.
La reussite consiste a deplacer cette piece de monnaie dans la case d'indice 7,
sans qu'il ne reste a la fin d'autre piece dans le tableau que celle de la case 7.

Les operations disponibles pour le faire sont :
- Ajouter-i : Ajoute 1 piece en cases i-1 et i+1 et supprime 1 piece de la case i
  si la case i a bien 2 cases voisines i-1 et i+1 et si elle contient au moins 1 piece
- Supprimer-i : Supprime 1 piece de la case i et ajoute 1 piece dans chacune des cases i-1 et i+1,
  si la case i contient au moins 1 piece et si elle a bien 2 cases voisines i-1 et i+1
Le jeu se termine lorsque le tableau atteint son etat final ou par abandon du joueur.
Le nombre minimal d'actions necessaires est 18.

Chaque action de l'utilisateur est comptabilisee dPs lors que :
- l'indice fourni correspond à un indice du tableau
  et
- l'action demandee est bien un Ajout ou une Suppression

pieces : +---+---+---+---+---+---+---+---+---+
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9
, nombre de Coups = 0

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : a1

pieces : +---+---+---+---+---+---+---+---+---+
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9
, nombre de Coups = 1

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : A2

pieces : +---+---+---+---+---+---+---+---+---+
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9
, nombre de Coups = 2

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : f1
Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : a100
Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : a3

pieces : +---+---+---+---+---+---+---+---+---+
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9
, nombre de Coups = 3

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : a3

pieces : +---+---+---+---+---+---+---+---+---+
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9
, nombre de Coups = 4

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : s1

pieces : +---+---+---+---+---+---+---+---+---+
| 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9
, nombre de Coups = 5

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : q

----- Fin de partie -----

pieces : +---+---+---+---+---+---+---+---+---+
| 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
+---+---+---+---+---+---+---+---+---+
      0 1 2 3 4 5 6 7 8 9

Abandon apres 5 coups.
Le nombre minimal de coups pour deplacer la piece de la case 1 a la case 7 est 18

Process returned 0 (0x0)   execution time : 66.470 s
Press any key to continue.
```

Scénario (extrait) : fin de partie par réussite du joueur

```
Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : s5
pieces : | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
          0  1  2  3  4  5  6  7  8  9
, nombre de Coups = 16

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : s6
pieces : | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
          0  1  2  3  4  5  6  7  8  9
, nombre de Coups = 17

Taper A ou S (Ajouter ou Supprimer) + indice choisi OU BIEN Q(uitter) : s7

----- Fin de partie -----
pieces : | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
          0  1  2  3  4  5  6  7  8  9
, nombre de Coups = 18

B R A V O !
Vous avez réussi en 18 coups.
Le nombre minimal de coups pour déplacer la piece de la case 1 a la case 7 est 18

Process returned 0 (0x0)   execution time : 43.269 s
Press any key to continue.
```

3. Ressources à disposition

Fonctionnalités graphiques

Vous pouvez utiliser des fonctionnalités proposées dans le module game-tools :

<https://github.com/patrick-etcheverry/game-tools>

Ce module met à disposition des fonctionnalités simples permettant d'effacer le terminal, de mettre le programme en pause, d'afficher des éléments en couleur ou encore de générer un nombre entier aléatoire. Les fonctionnalités de la bibliothèque game-tools doivent être utilisées :

- soit pour répondre aux besoins du problème ;
- soit pour améliorer l'interface du jeu, c'est-à-dire sans dénaturer le comportement attendu et décrit dans les scénarios.

En cas de doute, n'hésitez pas à en discuter avec votre enseignant.

Idées pour la saisie de la demande utilisateur

On pourra lire une chaîne de caractères et en extraire les éléments de réponse attendus.

Transformations entre valeurs de type char et int

- int → char

La fonction **char()** retourne le caractère de la table ASCII dont la position est fournie en paramètre.

```
int i = 65; // Déclare un entier i et l'initialise avec la valeur 65
char c;    // Déclare un caractère c
c = char (i); // Retourne dans c le ième caractère de la table ASCII
cout << c ; // Affiche 'A', le 65ème caractère de la table ASCII
```

- char → int

La fonction **int()** retourne la position de la table ASCII à laquelle se trouve le caractère fourni en paramètre.

```
char c = 'B'; // Déclare le caractère c et l'initialise avec la valeur 'B'
int i ;      // Déclare l'entier i
i = int (c); // Retourne dans i la position de 'B' dans table ASCII
```

4. Extensions possibles

- L'utilisateur peut choisir si le déplacement se fait de la case 1 à la case 7 ou l'inverse (de la case 7 vers la case 1).