

Projet 1: Un petit tableur

4 mai 2006

1 But du projet

Le but du projet est de réaliser un petit tableur. Un tableur est un programme permettant de manipuler des feuilles de calcul. Une feuille de calcul est composée de cellules organisées en ligne et en colonnes dans lesquelles on peut placer des données et effectuer des calculs.

2 Feuille de calcul

Le programme demandé ne gèrera qu'une seule feuille de calcul, composée de 10 lignes et de 6 colonnes.

Les cellules sont les éléments de la feuille. Elles sont référencées par leur numéro de ligne et de colonne. Par exemple (6,2) désigne la cellule au croisement de la sixième ligne et de la deuxième colonne.

Une cellule est soit vide, soit contient une expression qui peut être une valeur entière ou une formule.

Voici un exemple :

	1	2	3	4	5	6
1	5	(1,1)+5				
2			(1,2)/5			
3	2	12				
4	4	14				
5	6	10				
6	Somme((3,1),(5,2))	Moy((3,2),(5,2))				

Une formule est soit un opérateur arithmétique (+, -, *, /) appliqué à deux opérandes, soit les opérateurs *Somme* ou *Moy* appliqué à un bloc.

Les opérandes des opérateurs arithmétiques sont soit des valeurs entières, soit des références de cellules. Par exemple, le contenu de la cellule (1,2) peut être la formule (1, 1) + 5 c'est à dire la somme du contenu de la cellule (1,1) auquel on ajoute 5.

Les blocs, qui sont les arguments des opérations *Somme* et *Moy*, sont définis par la donnée de 2 références de cellules C1 et C2, désignant le bloc de cellules dont le coin supérieur gauche est la cellule C1 et le coin inférieur droit est la cellule C2. Par exemple, la cellule (6,1) contient la somme des cellules (3,1), (4,1), (5,1), (3,2), (4,2), (5,2).

Statut d'une cellule

Lorsqu'on définit le contenu d'une cellule par une formule, plusieurs cas de figure sont à envisager :

- La formule est syntaxiquement incorrecte. Par exemple `Moy(32)`. Votre programme doit rejeter de telles formules.
- La formule est syntaxiquement correcte mais non évaluable : cela se produit lorsqu'on divise par zéro, lorsque l'on référence une cellule qui n'a pas de valeur (vide ou non évaluable) ou lorsque l'on référence une cellule inexistante. Votre programme ne doit pas rejeter de telles formules, mais doit savoir que le statut de la cellule est "non évaluable" de façon à ne jamais chercher à calculer la valeur de la formule.
- La formule est correcte et l'expression évaluable. Le programme doit connaître la valeur correspondant à l'évaluation de la formule.

3 Travail à faire

Le travail qui vous est demandé consiste à faire un programme qui crée une table de 10 lignes et de 6 colonnes et qui propose un menu permettant :

1. d'afficher la table sous un format "numérique" : pour chaque cellule non vide on affichera soit sa valeur, soit un point d'interrogation (?) lorsque cette cellule correspond à une expression non évaluable ;
2. d'afficher la table sous un format "littéral" : pour chaque cellule non vide on affichera l'expression contenue dans la cellule ;
3. de modifier (ou de saisir) le contenu d'une cellule ;
4. de supprimer le contenu d'une cellule ;
5. d'initialiser la table avec des données provenant d'un fichier texte. Le format du fichier texte sera le suivant : Une ligne de la feuille correspondra à une ligne du fichier. Les cellules d'une même ligne seront séparées dans le fichier texte par le symbole &. Un fichier contenant

```
1 & 5 & ( 1 , 1 ) + 5 & & & &
2 & & & ( 1 , 2 ) / 5 & & &
3 & 2 & 12 & & & &
4 & 4 & 14 & & & &
5 & 6 & 10 & & & &
6 & Somme ( ( 3 , 1 ) , ( 5 , 2 ) & Moy ( 3 , 2 ) , ( 5 , 2 ) ) & & & &
```

engendrera une feuille de calcul correspondant à notre premier exemple.

6. de sauvegarder la table dans un fichier texte conforme au format précédemment défini.

4 Détails techniques

4.1 Représentation graphique de la table

Une librairie graphique permet d'afficher une table à l'écran et d'afficher des chaînes de caractères dans les cellules de cette table. Elle sera prochainement disponible. Vous pouvez en attendant vous contenter d'un affichage textuel standard.

4.2 Dépendances entre cellules

Une formule définit le contenu d'une cellule en fonction du contenu d'autres cellules. Ceci à 2 conséquences :

1. Lorsque l'on modifie le contenu d'une cellule, il est possible de créer un cycle dans les références de cellules. Supposons que
 - la cellule (1,1) contienne l'expression $(1,3) - 5$
 - la cellule (1,3) contienne l'expression $(2,4) / 5$
 - la cellule (2,4) contienne l'expression 100.

A ce point, il n'y a pas de problème. Supposons maintenant que l'on modifie le contenu de la cellule (2,4) par l'expression $(1,1) + 3$. Cette modification crée un cycle puisque (2,4) référence (1,1) qui référence (1,3) qui référence (2,4) : on ne peut évaluer correctement cette expression. Il faut donc détecter cette création de cycle et interdire la modification.

2. Lorsque l'on modifie ou supprime le contenu d'une cellule, les cellules qui dépendent de cette cellule voient aussi leur contenu modifié. Supposons que la cellule (3,2) contienne l'expression $5 + (1,1)$. Une modification de la cellule (1,1) pourra entraîner une modification de la cellule (3,2) ; on dit que la cellule (3,2) dépend de la cellule (1,1). Il faut donc mettre à jour l'ensemble de ces cellules. Cela peut se faire de 2 façons :
 - Dès qu'une cellule est modifiée, on recalcule l'ensemble de la table (cellule par cellule). Ce calcul est répété jusqu'à ce qu'aucun changement n'ait eu lieu entre deux calculs de table (il s'agit d'un algorithme de point fixe). Le problème de cet algorithme est qu'il nécessite de nombreux calculs inutiles.
 - On associe à chaque cellule C la liste des coordonnées des cellules qui dépendent de C. Ainsi, lors d'une modification de C, il suffira de parcourir cette liste pour connaître les calculs à faire. Ces calculs pourront néanmoins entraîner d'autres modifications et d'autres calculs. Cependant, on ne fera que les calculs nécessaires : ni plus, ni moins. Avec cet algorithme, lors de la modification d'une cellule C, il faut donc :
 - (a) mettre à jour la liste des dépendances des cellules référencées dans l'expression de C. Deux cas sont à prendre en compte : une cellule était référencée et ne l'est plus, il faut alors supprimer dans la liste de cette cellule la référence à C. Une cellule devient référencée, il faut alors ajouter à la liste de dépendances de cette cellule la référence à C.
 - (b) calculer la nouvelle valeur de la cellule ;
 - (c) parcourir la liste des cellules dépendantes et recalculer (récursivement) la valeur de ces cellules.

Cet algorithme est bien plus efficace que le précédent.

4.3 Représentation des données

Nous ne voulons pas présumer de la représentation des données et de la structuration en classe que vous choisirez. Voici cependant quelques indications pouvant vous aider.

4.3.1 Caractéristiques d'une cellule

Pour chaque cellule de la table, il serait sans doute judicieux de disposer des informations suivantes :

- un critère permettant de savoir s’il s’agit d’une cellule vide, d’une cellule contenant une expression non évaluable, d’une cellule contenant une simple valeur entière, d’une cellule contenant une opération.
- la valeur entière définissant la valeur de l’expression, si elle existe (cellule contenant une valeur simple ou une expression évaluable).
- l’opérateur utilisé lorsqu’il s’agit d’une formule.
- la première opérande si elle existe (formule).
- la deuxième opérande si elle existe (formule).
- la liste des références de cellules qui dépendent de cette cellule si vous implantez le second algorithme.

4.3.2 Caractéristiques des opérandes

Les opérandes de nos formules sont soit des valeurs entières, soit des références de cellules. Pour chaque opérande il serait judicieux d’avoir les informations suivantes :

- un critère permettant de savoir s’il s’agit d’une valeur simple ou pas.
- la valeur de l’opérande, si elle existe (valeur simple)
- le numéro de ligne et le numéro de colonne constituant la référence de la cellule (opérande non simple).

5 Travail optionnel

Pour ceux qui seront bien avancés dans leur projet, nous proposons d’ajouter :

1. la possibilité de désigner les cellules de façon relative dans l’écriture des formules
2. la possibilité de recopier le contenu d’une cellule source vers une cellule destination.

5.1 Désignation relative des cellules

Les références de cellule peuvent être relatives ou absolues. Dans notre exemple, et par défaut, les cellules sont référencées de façon absolue. La référence absolue est la place de cette cellule dans la table (ligne,colonne).

La référence relative à une cellule se fait en précédant le couple (numéro de ligne, numéro de colonne) par le signe \$. Par exemple \$(3, 2) est une référence relative de cellule. Lorsque la référence est relative, le couple (numéro de ligne, numéro de colonne) ne désigne pas la place dans la table mais la position de cette cellule relativement à la cellule y faisant référence. Ainsi, une référence relative donne le décalage (en terme de nombre de lignes et de colonnes) entre la cellule (dite de référence) et une cellule pointée (dite cellule référencée).

Par exemple, définir le contenu de la cellule (2,3) par la formule \$(1, 2)/5 revient à définir son contenu comme la division par 5 du contenu de la cellule se situant 1 ligne plus haut et une colonne plus à gauche. En effet, relativement à (2,3) , la cellule (1,2) se situe une ligne plus haut et une colonne plus à gauche.

5.2 copie du contenu d’une cellule source vers une cellule destination

Lorsque le contenu de la cellule source est une valeur, on recopie cette valeur dans la destination. Lorsque le contenu de la source est une formule, la recopie produira un effet différent suivant que la

formule contient des références absolues ou relatives.

Prenons un exemple : supposons que la cellule (2,1) soit définie par la formule : $\$(1, 3) + (1, 2)$.

La recopie de cette cellule vers la cellule (3,4) produira la formule : $\$(2, 5) + (1, 2)$.

En effet, (1,2) est une référence relative et doit donc être recopiée telle quelle ; en revanche, $\$(1, 3)$ est une référence relative, on doit donc la remplacer par la référence relative de la cellule qui est en même place relativement à (3,4) que (1,3) l'est par rapport à (2,1).

Or la cellule qui se trouve une ligne au dessus et une colonne à gauche de (3,4) est bien la cellule (2,5).

6 Dates et remarques importantes

Ce projet est individuel. Même si vous échangez des idées avec vos collègues, cela ne pourra pas justifier des parties de programme identiques ou très similaires entre deux projets différents.

Pour pouvoir être évalué, un programme doit compiler. Il est donc préférable d'avoir un programme qui compile, mais qui ne fait pas tout, qu'un programme complet qui ne compile pas.

Le projet sera à soutenir du 5 au 17 juin pour les cours du soir et de jour aux horaires habituels de TP (et dans les mêmes salles). Il sera à soutenir le 14 juin pour la FOD. Aucun report ne sera accordé.

A l'occasion de la soutenance, vous devrez :

- Rendre un listing de votre programme.
- Rendre une documentation engendrée par Javadoc de votre programme contenant un commentaire pour chaque fonctionnalité.
- Montrer le fonctionnement du programme à votre enseignant, qui le testera et vous posera des questions. La capacité à expliquer, anticiper et justifier le comportement et le code du programme est un élément très important de l'évaluation.