

## Introduction

Dans ce TP, vous allez implémenter une calculette en C#. L'utilisateur doit pouvoir saisir des nombres (entiers non signés dans une première phase du développement) et des opérateurs arithmétiques (+, -, \*, /) ; il doit pouvoir lire le résultat de l'opération.

## Exercice 1 Interface graphique

Construisez une calculette qui, provisoirement... ne calcule rien du tout. Il s'agit donc d'utiliser convenablement les composants graphiques disponibles. Vous n'avez pas de code à écrire : utilisez l'E.D.I. (programmation graphique d'une interface graphique...)

## Exercice 2 Modification du code

Allez voir – si vous ne l'avez déjà fait – le code généré ; modifiez les noms (projet, fichiers, variables, etc...) afin d'améliorer la lisibilité de l'ensemble (par exemple `boutonDivision` est plus explicite que `button11`, etc.).

## Exercice 3 Gestion des événements : conception

A ce stade, nous nous contenterons d'une calculette qui traite des entiers non signés. Dessinez l'automate avec sorties qui modélise le comportement de la calculette.

## Exercice 4 Gestion des événements : implémentation de l'automate.

## Exercice 5 Traitement des exceptions

Essayez de diviser un nombre par 0. Vous constatez que le système a levé une exception. Gérez cette exception.

## Exercice 6 Evolution de la calculette

La calculette doit maintenant traiter les nombre flottants positifs. Faites en sorte que la division par 0 lève une exception (le système ne le fait pas lorsque les opérandes sont des flottants !).

## Exercice 7 Compléments

7.1. Faites en sorte que l'on puisse itérer l'appui de la touche = , et aussi réutiliser le résultat dans la nouvelle opération :

```
a
+
b
=
(a+b)
=
(a+2b) etc...
*
c
=
((a+2b)*c)
7.2. On veut maintenant traiter des nombres signés.
7.3. Rajoutez une touche "clear" qui réinitialise la calculette, et une touche "backspace".
```