

---

## **TD 12- La calculatrice**

Inès de Courchelle, Thierry Garcia



2023-2024

**Objectifs :**

- Manipuler des arguments
- Gérer l'interprétation des caractères
- Réaliser des calculs en script

**Durée** 1h30**Format** papier/ordinateur**Attention** Durant ce TD l'utilisation de la souris, Touchpad et/ou Trackpoint sont strictement interdits. Seul le clavier est autorisé.**Exercice 1**

Écrire un script nommé question1.sh permettant d'effectuer un calcul simple.

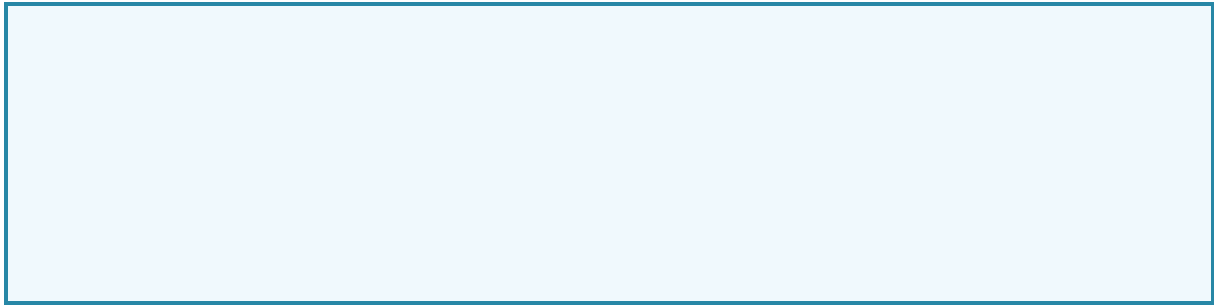
- N'oubliez pas de mettre des espaces pour séparer chaque caractère.
- Votre script devra contenir les fonctions effectuant toutes les opérations de base : addition, soustraction, multiplication et division.
- Il n'est pas demandé dans cet exercice de procéder aux vérifications des paramètres envoyés au script.
- Le script devra gérer 2 paramètres.

**Exécution :**

```
./question1 3 2
```

**Affichage du programme :**

```
somme 3 + 2 = 5  
soustraction 3 - 2 = 1  
multiplication 3 * 2 = 6  
division 3 / 2 = 0
```



## Exercice 2

Écrire un script nommé question2.sh permettant de faire l'analyse des paramètres donnés au lancement du script.

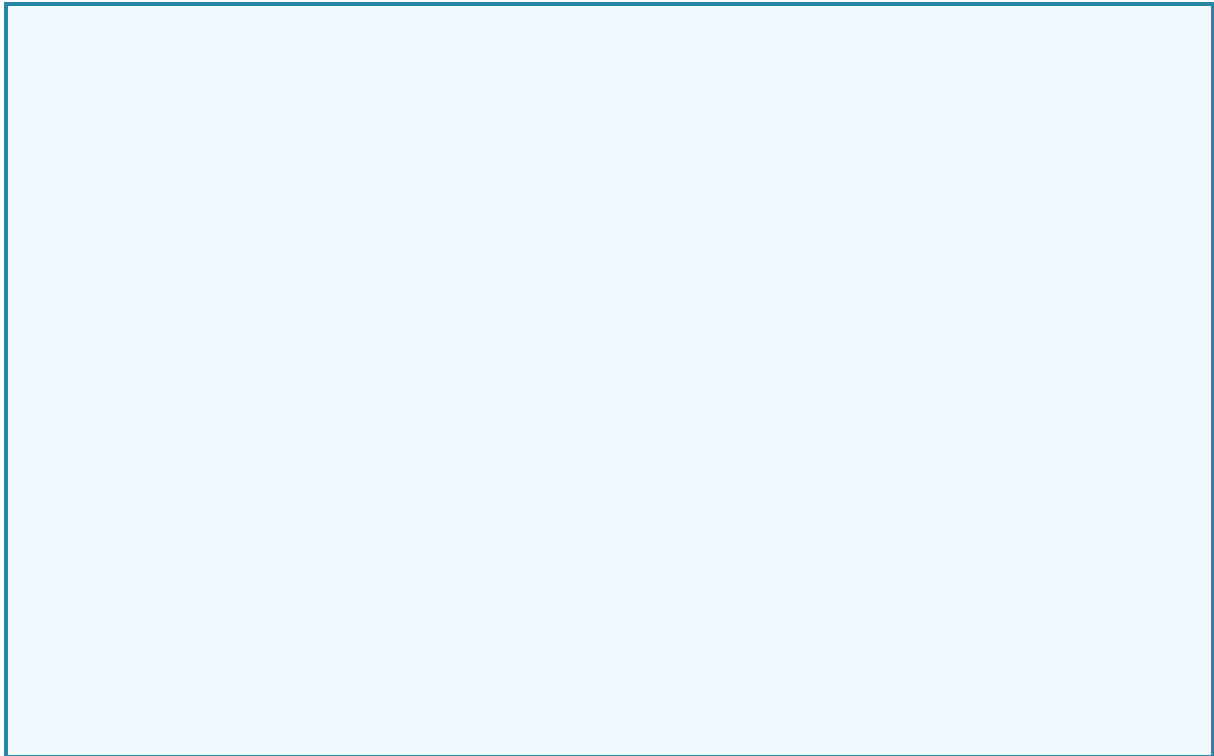
- Ce script devra identifier la nature de chaque élément.
- On admettra que tout autre caractère que les parenthèses ou les opérateurs est un entier.
- Le script devra gérer 2 entiers.

### Exécution :

```
./question2.sh ( 2 + 3 )
```

### Affichage du programme :

```
parenthèse ouvrante entier somme entier parenthèse fermante
```



### Exercice 3

Écrire un script (à partir des deux exercices précédents) nommé question3.sh permettant d'effectuer un calcul simple.

- N'oubliez pas de mettre des espaces pour séparer chaque caractère.
- Votre script devra contenir les fonctions effectuant les opérations de base : addition, soustraction, multiplication et division.
- Il n'est pas demandé dans cet exercice de procéder aux vérifications des paramètres envoyés au script.
- La gestion des parenthèses n'est pas à faire non plus à ce niveau. On se limitera à deux opérandes pour le moment.

#### Exécution :

```
./question2.sh ( 2 + 3 )
```

**Affichage du programme :**

```
5
```

**Exécution :**

```
./question2.sh ( 2 * 3 )
```

**Affichage du programme :**

```
6
```

**Exercice 4**

Écrire un script nommé question4.sh capable de vérifier la syntaxe du calcul demandé par l'utilisateur. Cette syntaxe doit respecter une alternance entre les opérations et les entiers. Le script devra comporter

une fonction capable de détecter une mauvaise alternance.

**Exécution :**

```
./question4.sh 2 + + 3
```

**Affichage du programme :**

```
entier somme somme ** erreur alternance **
```

**Exécution :**

```
./question4.sh 2 2 + 3
```

**Affichage du programme :**

```
entier entier ** erreur alternance **
```



### Exercice 5

Écrire un script nommé `question5.sh` capable de vérifier la syntaxe du calcul demandé par l'utilisateur en tenant compte du parenthésage. Cette syntaxe doit respecter une alternance entre les opérations, les entiers et les parenthèses. Le script devra comporter une fonction capable de détecter une mauvaise alternance. Pour simplifier le travail, un mauvais parenthésage ne sera notifié qu'à la fin de la vérification de l'alternance.

**Exécution :**

```
./question5.sh (( 2 + 3 ))
```

**Affichage du programme :**

```
para ouvrante para ouvrante entier somme entier para fermante ** erreur parenthèse ! **
```

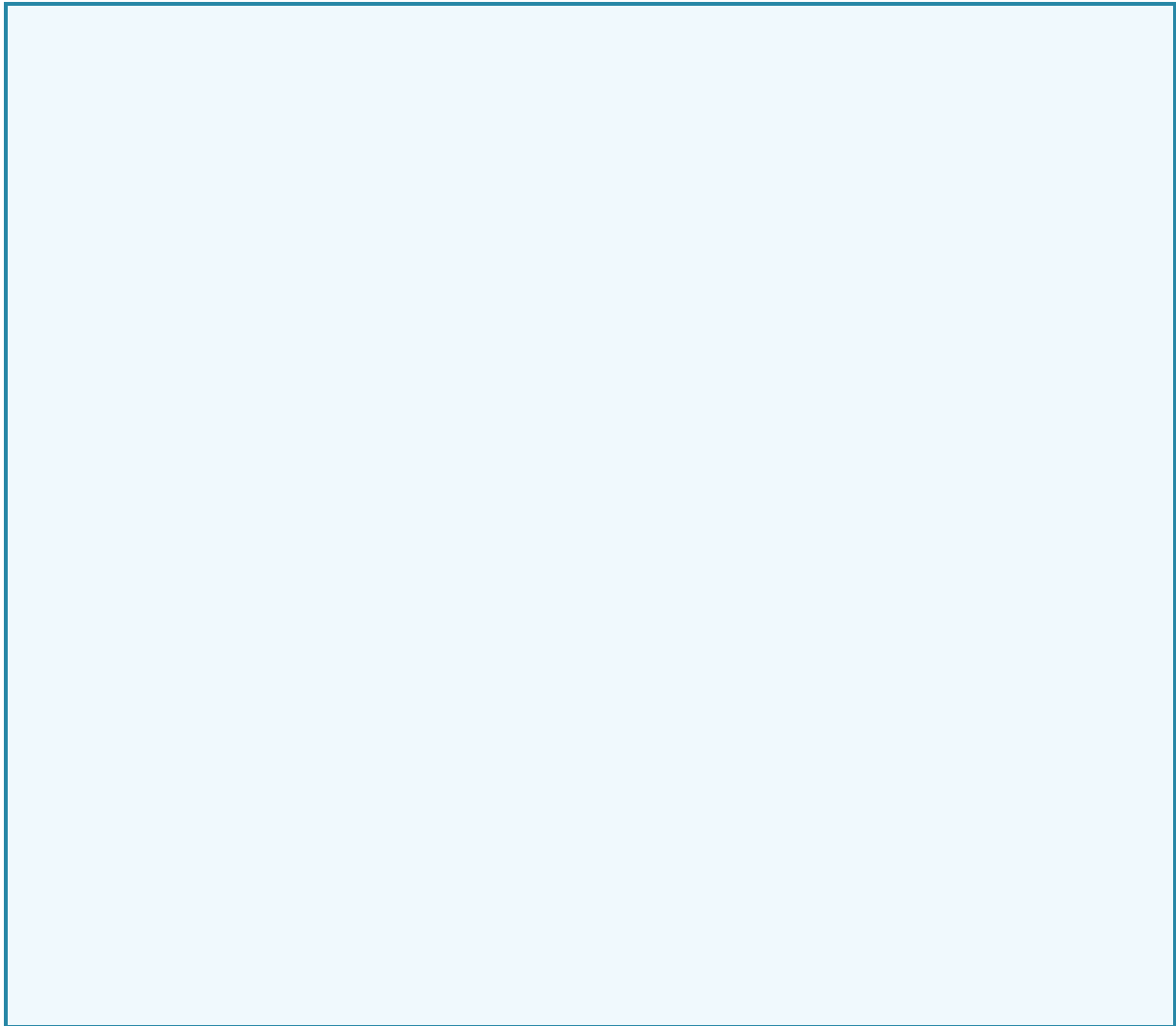
**Exécution :**

```
/question5.sh ( 2 + ( 3 ) )
```

**Affichage du programme :**

```
para ouvrante entier somme para ouvrante entier para fermante ** erreur parenthèse ! **
```



**Exercice 6**

À partir du script question3.sh, proposez un script question6.sh permettant de gérer plusieurs opérandes. Il n'est pas demandé de gérer la priorité des opérateurs.

**Exécution :**

```
./question6.sh 2 + 3 x 4
```

**Affichage du programme :**

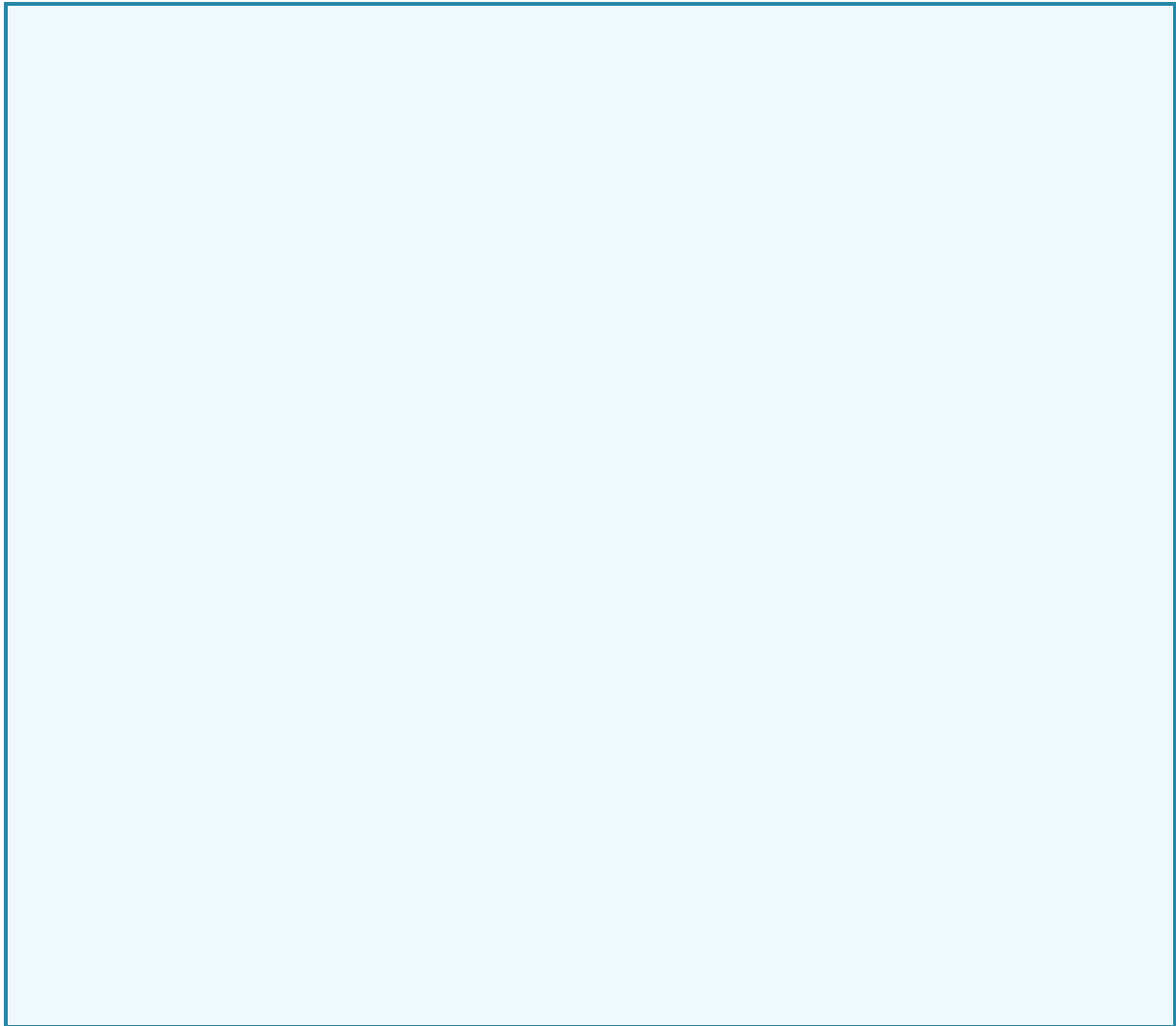
```
résultat = 20
```

**Exécution :**

```
./question6.sh 2 x 3 + 2
```

**Affichage du programme :**

```
résultat = 8
```



### Exercice 7

En vous servant de tous les exercices précédents, écrire le script nommé **calc.sh** permettant de simuler une calculatrice simple. Pour rappel, cette calculatrice ne doit pas gérer la priorité des opérateurs. D'autre part, pour simplifier, on ne tiendra compte que d'un seul niveau de parenthésage. Le script pourra générer une erreur dans le cas d'un parenthésage à plusieurs niveaux.

**Exemple à ne pas traiter**

```
./calc.sh (2 + 3) x ((2 - 4) + 3)
```

**Exemple traiter**

```
./calc.sh (2 + 3) x (2 - 4) résultat = -10
```

Il est à noter que la priorité des opérateurs peut être gérée par l'intermédiaire des parenthèses.

```
./calc.sh 2 + 3 x 4 résultat = 20
```

```
./calc.sh 2 + (3 x 4) résultat = 14
```

```
1 \boxA{14cm}{}
```