

INF404 - Travaux pratiques - TP4

Expressions arithmétiques “générales” (partie 2)

Avant de commencer cette séance :

1. Créez un répertoire **TP4** dans votre répertoire **INF404**
2. Placez-vous dans **INF404/TP4** et recopiez les fichiers utilisés pendant le TP3 : `cp ../TP3/* .`

Remarques :

- Ce TP doit être fait **après** avoir terminé le TP précédent ...
- Vous pouvez vous aider des **transparentes du cours 4**¹

Objectifs

L’objectif de cette séance est de **terminer** votre programme et d’évaluation d’expressions arithmétiques “générales” (avec opérateurs +, − et *, ainsi que les parenthèses). On rappelle la grammaire définissant la syntaxe de ces expressions :

$$\begin{aligned}
 eag &\rightarrow seq_terme \\
 seq_terme &\rightarrow terme \ suite_seq_terme \\
 suite_seq_terme &\rightarrow op1 \ terme \ suite_seq_terme \\
 suite_seq_terme &\rightarrow \varepsilon \\
 terme &\rightarrow seq_facteur \\
 seq_facteur &\rightarrow facteur \ suite_seq_facteur \\
 suite_seq_facteur &\rightarrow op2 \ facteur \ suite_seq_facteur \\
 suite_seq_facteur &\rightarrow \varepsilon \\
 facteur &\rightarrow ENTIER \\
 facteur &\rightarrow PARO \ eag \ PARF \\
 op1 &\rightarrow PLUS \\
 op1 &\rightarrow MOINS \\
 op2 &\rightarrow MUL
 \end{aligned}$$

(suite page suivante)

1. disponible sur Moodle ...

Exercice 1 - construction de l'arbre abstrait d'une EAG

Complétez votre analyse syntaxique du TP3 en étendant la procédure *analyser* afin qu'elle produise un arbre abstrait de l'expression lue (lorsque celle-ci est correcte) :

```
procedure analyser (char *nom_fichier, Ast *arbre) ;
-- e.i : indifférent
-- e.f : une EAG a été lue dans le fichier de nom nom_fichier
-- si elle ne contient pas d'erreur arbre contient son arbre abstrait
-- sinon une erreur est signalée
```

Ecrivez un programme principal *calcullette.c* qui appelle cette procédure, et testez ce programme sur divers exemples. N'oubliez pas de vérifier que votre expression se termine bien sur une "fin de séquence".

Vous pouvez afficher l'arbre produit en utilisant la fonction *afficher* du module *ast_parcours* (voir le TP3).

Exercice 2 - ajouter l'évaluation de l'expression

Complétez votre programme principal pour appeler la fonction *evaluation* (écrite au TP3) sur l'arbre abstrait produit par votre procédure *analyser*.

Testez ensuite votre programme sur différentes expressions ...

Exercice 3 - extensions (non obligatoires !)

Quelques pistes possibles qui pourront être utiles par la suite ...

1. Améliorer le traitement des erreurs en donnant un message explicite (erreur lexicale, erreur syntaxique, opérateur attendu, numéro de colonne, etc.)
2. prendre en compte les réels
3. etc.

Compte-rendu

Comme ce TP termine la partie "calcullette" du projet, votre "compte-rendu" à déposer sur Moodle avant le 4 mars doit donc contenir l'ensemble de vos programmes du TP4 et les jeux de test correspondants ...