

IN406 - Théorie des langages
Projet - Evaluation d'expressions
booléennes

Lience 2 informatique
Universite de Versailles
Saint-Quentin-en-Yvelines

BRAHIMI LOUNES
ADDOUCHE MOULOUE

2018/2019

0.1 GRAMMAIRE

$$G = (\Sigma, N, S, P)$$

$$\Sigma = \{0, 1, +, ., =>, <=>, NON, (,)\}$$

$$N = \{expr, cte, oper\}$$

$$S = \{expr\}$$

P (les regles de productions) :

$$expr \longrightarrow cte \mid (expr) \mid expr\ oper\ expr \mid NON\ expr$$

$$cte \longrightarrow 0 \mid 1$$

$$oper \longrightarrow + \mid . \mid => \mid <=>$$

0.2 String to token :

La chaîne de caractère introduite en 2eme position lors de l'exécution de eval : `(./eval "(1+0)")` est récupéré et elle est passé en argument à la fonction string to token qui retourne une liste chaînée de struct token, cette dernière extrait des

sous-chaine de notre string et avance de caractère en caractère (pour gérer les opérateur, parenthèses et constantes quand elles sont collées) et il est implementer de façon à gérer les espaces(un ou plusieurs).

0.3 Expression booléenne bien formée :

La liste de token qu'on a obtenue précédemment est passé en argument a la fonction Automate qui vérifie si notre expression est bien formée et s'assure que les parenthèses si elle en a sont correcte.

0.4 Creation de l'arbre à partir de la liste de tokens :

0.5 Calculer la valeur de l'expression :

l'évaluation ce fait par la fonction du meme nom "evaluation" qui prend en argument notre arbre binaire tout bien construit, elle commence par retourner la valeur des feuilles a leur parents en commenceant par la gauche, c'est a dire si le

noeud present est une constante elle retourne ça valeur si non elle visite c fils, a son retour elle execute l'operation du noeud parent, ainsi si notre noeud est un operateur est que son operation a ete exécuté la valeur est retourner au noeud parent (si il existe il est necessairement operateur) donc ce dernier l'utilise pour son equation, et enfin notre fonction nous retourne le resultat (un int) Vrai ou Faux.