

0.1 Espressioni

Le espressioni possono essere utilizzate per assegnare valori ad una variabile, in condizioni booleane o anche come argomenti di una funzione.

```

exp          : '(' exp ')'                                     #
  baseExp
  | '-' exp                                         #negExp
  | '!' exp                                         #notExp
  | lhs                                             #
    derExp
  | 'new' type                                     #newExp
  | left=exp op=('*' | '/')                      right=exp #binExp
  | left=exp op=('+' | '-')                      right=exp #binExp
  | left=exp op=('<' | '<=' | '>' | '>=')          right=exp #binExp
  | left=exp op=('==' | '!=')                    right=exp #binExp
  | left=exp op='&&'                             right=exp #binExp
  | left=exp op='||'                             right=exp #binExp
  | call                                           #callExp
  | BOOL                                           #boolExp
  | NUMBER                                         #valExp;

//Booleans
BOOL      : 'true' | 'false';

//IDs
fragment CHAR      : 'a'..'z' | 'A'..'Z' ;
ID         : CHAR (CHAR | DIGIT)* ;

//Numbers
fragment DIGIT      : '0'..'9';
NUMBER      : DIGIT+;

```