

token (lexical items = atomic lexical)

I detect  
def  
for  
if

range

+, \*, =, >  
(, ), :

for in range

0, 1

a, b, c, count, limit

II classify

① def, in, for, ...

(reserved words) ? keywords

② separators (delimiters) (, ), :, ;, [, ], {, }, ...

③ Operators - arithmetic +, -, \*, /  
relational <, >, <=, ==  
logical and, or, not

④ Identifier count a, b, limit

⑤ Constant int 0, 1, 25, -35

01  
"Hello world!"

white sp. space, indent, newline  
1, 2, 3 - list  
4, 5 - rules

id rule

- first symbol = letter or '\_'

rest symbol = letter or digit

≠ reserved word

Count rule.

BNF  $exp = term \mid term \text{ " + " } exp \mid term \text{ " - " } exp$  direct !!!

$term = factor \mid factor \text{ " * " } term \mid factor \text{ " / " } term$

$factor = id \mid const$

EBNF  $exp := term \{ ('+' | '-' ) exp \}$

$term = factor \{ ('*' | '/' ) term \}$

$exp = exp \text{ " + " } exp \mid exp \text{ " - " } exp$

$exp \text{ " * " } exp \mid exp \text{ " / " } exp$

$const \mid id$  ! WRONG

int

const

- contains only digits

- might start with sign '+' | '-' (signed or unsigned)

- first digit non zero

nonzero digit := '1' | '2' | ... | '9'

digit := '0' | nonzero digit

int const := [ '+' | '-' ] nonzero digit { digit } | '0'

int a;

int a, int b;

int a, b, float x;

