

mikroC gramatika

Simboli

letter

→	"_"	"a"	"A"	"b"	"B"	"c"	"C"	"d"	"D"	"e"	"E"	"f"
	"F"	"g"	"G"	"h"	"H"	"i"	"I"	"j"	"J"	"k"	"K"	"l"
	"L"	"m"	"M"	"n"	"N"	"o"	"O"	"p"	"P"	"q"	"Q"	"r"
	"R"	"s"	"S"	"t"	"T"	"u"	"U"	"v"	"V"	"w"	"W"	"x"
	"X"	"y"	"Y"	"z"	"Z"							

digit

→ "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

identifier

→ letter (letter | digit)*

int_constant

→ digit +

unsigned_constant

→ digit + ("u" | "U")

Pojmovi

program

→ variable_list function_list
→ function_list

variable_list

→ variable ";"
→ variable_list variable ";"

variable

→ type identifier

type

→ "int"
→ "unsigned"

function_list

→ function
→ function_list function

function

→ type identifier "(" parameters ")" body

parameters

→ ε
→ parameter_list

parameter_list

→ variable
→ parameter_list "," variable

body

→ "{" variable_list statement_list "
→ "{" statement_list "

statement_list

→ ε
→ statement_list statement

```

statement
  → assignment_statement
  → if_statement
  → return_statement
  → compound_statement

assignment_statement
  → identifier "=" num_exp ";"

num_exp
  → mul_exp
  → num_exp "+" mul_exp
  → num_exp "-" mul_exp

mul_exp
  → exp
  → mul_exp "*" exp
  → mul_exp "/" exp

exp
  → constant
  → identifier
  → function_call
  → "(" num_exp ")"

constant
  → int_constant
  → unsigned_constant

function_call
  → identifier "(" arguments ")"

arguments
  → ε
  → argument_list

argument_list
  → num_exp
  → argument_list "," num_exp

if_statement
  → if_part
  → if_part "else" statement

if_part
  → "if" "(" rel_exp ")" statement

rel_exp
  → num_exp "<" num_exp
  → num_exp ">" num_exp
  → num_exp "<=" num_exp
  → num_exp ">=" num_exp
  → num_exp "==" num_exp
  → num_exp "!=" num_exp

return_statement
  → "return" num_exp ";"

compound_statement
  → "{" statement_list "}"

```

Komentar: /* a comment */

Razmak i kraj linije imaju funkciju separatora simbola

Oznake:

+	1 or more occurrence	<i>italic</i>	pojmovi
*	0 or more occurrences	bold	simboli
	alternatives ("or")		

Prioritet Operatora

()	najveći prioritet
* /	
+ -	
< <= >= > == !=	najmanji prioritet

Tokens

Regularni izraz	Token	Vrednost
" , "	COMMA	-
" ; "	SEMICOLON	-
" ("	LPAREN	-
") "	RPAREN	-
" { "	LBRACKET	-
" } "	RBRACKET	-
" < "	RELOP	LT
" <= "	RELOP	LE
" > "	RELOP	GT
" >= "	RELOP	GE
" == "	RELOP	EQ
" != "	RELOP	NE
" = "	ASSIGN	-
" + "	PLUS	-
" - "	MINUS	-
" * "	TIMES	-
" / "	DIV	-
letter (letter digit) *	ID	adresa stringa
digit +	INT_NUMBER	adresa stringa
digit + ("u" "U")	UNSIGNED_NUMBER	adresa stringa
"int"	TYPE	INT_TYPE
"unsigned"	TYPE	UNSIGNED_TYPE
"if"	IF	-
"else"	ELSE	-
"while"	WHILE	-
"return"	RETURN	-