

Lab 1b

Alex Pop

October 14, 2023

Contents

1	Lexic.txt	1
1.1	Alphabet	1
1.2	Lexic	1
1.2.1	Special symbols	1
1.2.2	Identifiers	1
1.2.3	Constants	2
2	Syntax.in	2
3	Token.in	3

1 Lexic.txt

1.1 Alphabet

- lowercase letters of the English alphabet
- digits 0-9
- underscore: _

1.2 Lexic

1.2.1 Special symbols

- operators: +, -, *, /, %, ==, <, <=, >, >=, =, !=
- separators: {, }, [,], (,), ;, space, newline, "

- reserved words: `int`, `str`, `double`, `if`, `else`, `read_int`, `read_str`, `read_double`, `print_int`, `print_str`, `print_double`, `while`, `get`, `set`

1.2.2 Identifiers

A sequence of lowercase letters, digits and underscore with the first character being a lowercase letter.

```
identifier = (letter|"_"){letter|digit|"_"}
letter = "A"|"B"|"C"|"D"|"E"|"F"|"G"|"H"|"I"|"J"|"K"|"L"|"M"|"N"|"O"|"P"|"Q"|"R"|"S"|"T"|"U"|"V"|"W"|"X"|"Y"|"Z"|"a"|"b"|"c"|"d"|"e"|"f"|"g"|"h"|"i"|"j"|"k"|"l"|"m"|"n"|"o"|"p"|"q"|"r"|"s"|"t"|"u"|"v"|"w"|"x"|"y"|"z"
digit = "0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9"
```

1.2.3 Constants

```
int_constant = maybe_sign non_zero_digit {digit} | "0"
non_zero_digit = "1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9"
maybe_sign = ["+"|"-"]
str_constant = ""{letter|digit|"_"}""
double_constant = (int_constant | (maybe_sign "0")) "." {digit}
```

2 Syntax.in

```
program = {statement}
statement = ((variable_declaration | function_call | assignment) ";") | if | while

type = "int" | "str" | "double"

variable_declaration =
  | type " " identifier
  | type " " identifier "[" int_constant "]"

function_call =
  | get_call
  | set_call
  | read_call
  | print_call

get_call = "get" "(" identifier "," int_expression ")"
set_call = "set" "(" identifier "," int_expression "," expression ")"
```

```

read_call = ("read_int" | "read_str" | "read_double") "(" ")"
print_call = ("print_int" | "print_str" | "print_double") "(" expression ")"

assignment = identifier "=" expression

if = "if" "(" bool_expression ")" "{" program "}" ["else" "{" program "}"]
while = "while" "(" bool_expression ")" "{" program "}"

constant = int_constant | str_constant | double_constant

bool_operator = "==" | "!=" | "<" | ">" | "<=" | ">="
bool_expression = (constant | identifier) bool_operator (constant | identifier)

expression = int_expression | str_expression | double_expression
  int_expression = int_term | int_term ("+" | "-") int_expression
    int_term = int_factor | int_factor ("*" | "/" | "%") int_term
      int_factor = int_constant | identifier | "(" int_expression ")"
double_expression = double_term | double_term ("+" | "-") double_expression
  double_term = double_factor | double_factor ("*" | "/" | "%") double_term
    double_factor = double_constant | identifier | "(" double_expression ")"
str_expression = str_term | str_term "+" str_expression
  str_term = str_constant | identifier

```

3 Token.in

```

+
-
*
/
%
==
<
<=
>
>=
=
!=
{
}

```

```
(  
)  
[  
]  
;  
space  
newline  
"  
int  
str  
double  
if  
else  
while  
get  
set  
read_int  
read_str  
read_double  
print_int  
print_str  
print_double  
.
```