MangaCC

Language Descriptions

Tokens

1- Variable and constant Declaration

```
int x;
int x = 5;
const int x = 5;
string x = "Hello";
string x = "c";
```

- Available types
    - int
    - float
    - string
    - bool

2- Mathematical and Logical expressions

- Mathematical operator
    - +, -, *, /, %,
- Logical operator
    - ==, !=, >, <, >=, <=
    - and , or , not , xor

PROF

3 - Assignment statement

```
x = 5;
x = "Hello";
x = 5.5;
x = true;
```

4 - If else statement

```
if (x == 5 and y == 10) {
  x = 10;
} endif
```

```
if (x >= 5 or y <= 10) {
  x = x +1 ;
} else {
  x = x - 1;
}
```

```
if (x < 5 and y > 10) {
  x = x + 1;
} else if (x > 5) {
  x = x - 1;
} else {
  x = 0;
}
```

5 - While loop

```
while (x < 5) {
  x = x + 1;
}
```

6 - Repeat until

```
repeat {
  x = x + 1;
} until (x == 5);
```

PROF

7 - For loop

```
for (int i = 0; i < 5; i = i + 1) {
  x = x + 1;
}
```

8 - Switch case

```
switch (x) {
  case 1:
    x = 1;
    break;
  case 2:
    x = 2;
    break;
  default:
    x = 0;
}
```

9 - Function declaration

```
int func1(int x, int y) {
  return x + y;
}
```

```
void func2(int x) {
  x = x + 1;
}
```

```
int func3() {
  return 5;
}
```

10 - Function call

PROF

```
int x = func1(5, 10);
func2(5);
int y = func3();
```

11 - Block structure

```
{
  int x = 5;
  {
    x = x + 1;
  }
}
```

12- Comments

```
// This is a comment
```

13- Print

```
print("Hello World");
print(x);
print(5);
print(5+6);
```

14- Enum

```
enum Color {
  RED,
  GREEN,
  BLUE
}
enum Color c = RED;
```

How to run

- There is a script named **build.sh** that generates the parser and lexer and takes test file from you as input argurmnt.
- You can run the script by typing the following command in the terminal:

  ```
  ./build.sh if_test.c
  ```

PROF

- Note: the test file should be in the test_cases folder.
- It will print the tokens with its line number.

Quadraples Description

Quadraple	Description
PUSH (value)	Pushing (value) to stack
POP (ID)	Pop value from stack to the (ID)
NOT	Getting complement of value
ADD (arg1) (arg2) (result)	Adding (arg1) (arg2) and save value to (result)

Quadraple	Description
SUB (arg1) (arg2) (result)	Substracting (arg1) (arg2) and save value to (result)
MUL	
DIV	
MOD	
AND	
OR	
EQ	
NE	
LT	
GT	
LE	
GE	
XOR	
JMP	
JUMPZERO	
JMPNONZERO	
Convi	
Convf	

Contributors

PROF

Ahmed Ihab

Yahia Zakaria

Youssef Ahmed

Weaam Bassem