
Cairo University - Faculty of Engineering

Computer Engineering Department


# Compilers

# Project Document

Submitted to:

## Dr. Mona Farouk

## Eng. Nesma Refaei

## By: Team 17

| Name | Section | B.N. |
|---|---|---|
| Andrew Tadros | 1 | 14 |
| Mark Medhat | 2 | 12 |

Academic Year
2021 / 2022

## ➢ Project Overview:

Simple Programming Language like C language using Lexx and Yacc

## ➢ Tools Used:

- Flex
- Bison
- Python

## ➢ Tokens:

| Token | Description |
|---|---|
| VARIDENTIFIER | Variable name |
| FUNCIDENTIFIER | Funtion name |
| INTEGER | Integer value |
| STRING | String value |
| CONST | “const” |
| INT | “int” |
| STR | “String” |
| BOOL | “bool” |
| TRUEBOOL | 1 |
| FALSEBOOL | 0 |
| func | For function declaration |
| WHILE | “while” |
| DO | “do” |
| FOR | “for” |
| BREAK | “break” |
| IF | “if” |
| SWITCH | “switch” |
| CASE | “case” |
| DEFAULT | “default” |
| IFX | used to clear ambiguity |
| ELSE | “else” |
| EQ | == |
| NE | != |
| AND | && |
| OR | \|\| |
| NOT | ! |

- **Variables &Constants declaration:**

```
int x;
int x=5;
const int x = 5;
bool y = false;
string x = “Project”;
```

- **If-Else statements**

```
if (i==0) { }
else if (i==1) { }
else{ }
```

- **Switch statements**

```
switch (x)
case 0:
x=x+1;
break;
```

- **Loops:**

```
for (int i=0; i<5; i=i+1) { }
do { } while (true);
while(true) { };
```

- **Expressions**

```
int func Main()
{
bool x = true;
bool y = true;
bool z;
z= (x==y);
z= (x!= y);
}
```

- **Comments**

```
// This is comment in our language
```

- **Functions**

```
int func _main()
{ }
```

- **Assumptions:**
  - Logical Expressions work only inside a function
  - Function name must begin with underscore

| Quadruple | Description |
|---|---|
| JMP L | Unconditional jump to label L |
| JZ L | Jump to label L if $1 == 0 |
| JNZ L | Jump to label L if $1 != 0 |
| PUSH X | $1 = X |
| POP X | X = $1 |
| ADD | $1 = ($1 + $2) |
| SUB | $1 = ($1 - $2) |
| MUL | $1 = ($1 * $2) |
| DIV | $1 = ($1 / $2) |
| AND | $1 = ($1 && $2) |
| OR | $1 = ($1 \|\| $2) |
| NOT | $1 = !$1 |
| MOV R1, C | First parameter = C |
| cmpLT | $1 = ($1 < $2) |
| cmpGT | $1 = ($1 > $2) |
| cmpEQ | $1 = ($1 == $2) |