

# 词法分析作业

使用语言：python

因为python处理字符方便，这里实验使用python来实验。

这次实验执行 `python main.py $input` 输入文件

先读下文件内容，然后分情况处理数据。

## 1. Keyword

```
keyword = {'if':'If', 'else':'Else', 'while':'While', 'break':'Break', 'continue':'Continue',  
          'return':'Return', '=':'Assign', ';':'Semicolon', '(':'LPar', ')':'RPar', '{':'LBrace',  
          '}':'RBrace', '+':'Plus', '*':'Mult', '/':'Div', '<':'Lt', '>':'Gt', '==':'Eq'}
```

如果词法是 keyword 时候 直接变换出来

## 2. Isalpha

如果词法是 只有英文构成 直接变换

```
elif(temp2.isalpha()): # aa, abc  
    print("Ident("+temp2+")")  
    inputFlag = 1
```

Ex) aa , abc

## 3. Isdigit

如果词法是 只有数据构成

```
elif(temp2.isdigit()): # 12  
    print("Number("+temp2+")")  
    inputFlag = 1
```

Ex) 12, 34

#### 4. 英文+数据 变数

如果词法是 英文+数字 或者 数字+英文 ex) a123, 123a

```
elif (temp2.isalnum()): # a123
    if(temp2[0].isdigit()): # 123a123
        m = p.search(temp2)
        # print("테스트 입니다 : " +str(m.end()))
        number = temp2[:m.end()-1]
        string = temp2[m.end()-1:]
        print("Number("+number+")")
        print("Ident(" + string + ")")
    else:
        print("Ident("+temp2+")")
inputFlag = 1
```

如果 数字+英文情况时候, 先分离 两个 前面是 数字 后面是 英文

Ex) 123a -》 123 a 然后 出来

英文+数字 直接 变数名字处理

## 5. 特殊情况

```
if(';' in temp2):
    temp2 = temp2[:temp2.find(';')] # semicolon delete
    semiFlag = 1 # semicolon flag on
    # print("테스트입니다 : "+temp2)
elif('(' in temp2):
    temp2 = temp2[temp2.find('(')+1:]
    # print("테스트용1:"+temp2)
    print("LPar")
elif(')' in temp2):
    temp2 = temp2[:temp2.find(')')]
    # print("테스트용2:" + temp2)
    rParFlag = 1
elif "==" in temp2 and len(temp2)>=3:
    temp2 = temp2[:len(temp2)-3]
    print('Eq')
    inputFlag = 1

if ('\n' in temp2):
    temp2 = temp2[:temp2.find('\n')]
```

这里进行数据处理前，特殊情况先加处理，以免数据处理失败

```
if semiFlag:
    print("Semicolon")
    semiFlag = 0
    inputFlag = 1

if rParFlag:
    print("RPar")
    rParFlag = 0
    inputFlag = 1

if inputFlag==0:
    print("Err")
    # print(temp2)
    exit(0)
inputFlag = 0
```