

Minilisp

Settings

- environment: linux
- lang: python

run

```
python3 run.py demo2.py
```

Features

1. Syntax Validation
2. Print
3. Numerical Operations
4. Logical Operations
5. if Expression
6. Variable Definition
7. Function
8. Named Function

program流程

main() -> run() -> parseTree() -> recurrence calling evalu()

- 在main把所有輸入丟到x裡面。把所有輸入都丟進去是為了避免那些tab和無用的換行造成問題
- parseTree主要做把剛讀到的x去切成具有層層包好的tuple，用成tuple而不適用list是因為tuple是用小括號，而本來就是用小瓜好去分層的

```
def parseTree(code):  
    # split tokens to recurrence  
    tokens = code.replace('(', ' ( ').replace(')', ' ) ').split()  
    tree_token = []  
    for token in tokens:  
        if token not in ['(', ')']:  
            tree_token.append(f'({token})')  
        else:  
            tree_token.append(token)  
        if token != '(':  
            tree_token.append(',')  
    return eval(''.join(['(', *tree_token, ')']))
```

- 準備好切好grammar的tuple後，把他用recurrence的方法去呼叫evalu的function，也就是計算最重要的一個function
- evalu func有下面幾種主要判斷：

- 傳入的這個第一個參數是if
- 參數是fun
- 參數是define
- 參數還是tuple
 - 繼續遞迴下去找

