# Code:

```python
from tabulate import tabulate
nemonic_table = {"MOVER":{"opcode":4, "length":1}, "MULTI":{"opcode":4, "length":1}, "MOVEM":{"opcode":4, "length":1}, "ADD":{"opcode":4, "length":1}}
inp = """START 100
MOVER BREG, FIVE
ADD BREF, ='4'
MOVEM BREG, TEMP
FIVE DC '5'
TEMP DS '1'
END"""
defs = ['DB','DW', 'DD','DQ', 'DT', 'DS', 'DC']
op_file = open("op.txt","w+")
#splitting the input code at '\n'
each_line = inp.split('\n')
#print(each_line)
lc = 0
literal_t = []
symbol_t = []
# print(each_line)
for line in each_line:
    text = line.split()
    # print(text)
    # print("This is")
    for i in range(len(text)):
        # print(text[i])
        if(text[i] == "END"):
            print(line)
            break
        if(text[i] == "START"):
            lc = int(text[i+1])
        if(text[i] in nemonic_table):
            print(line + "     LC = ",lc)
            lc += nemonic_table[text[i]]["length"]
        elif(text[i] in defs):
            if(i==0):
                print(line + "     LC = ",lc)
                lc += 1
            else:
                print(text[i] + "    LC = ",lc)
                lc += 1
        elif(i==0):
            if(text[i] == "START"):
                continue
            symbol_t.append(tuple((text[i],lc)))
            print(line + "     LC = ",lc)
            lc += 1
            break


        if(text[i].startswith("='")):
            literal_t.append(tuple((text[i][2],lc-1)))

table = []
print("\n\n\n")
stable = []
ltable = []
print("Mnemonic Table")
table.append(["Mnemonic","Opcode","Length"])
for key in nemonic_table:
    table.append([key,nemonic_table[key]["opcode"],nemonic_table[key]["length"]])
print(tabulate(table))

print("\n\n")

print("Symbol Table")
stable.append(["Symbol","Address"])
for pair in symbol_t:
    stable.append([pair[0],pair[1]])
print(tabulate(stable))

print("\n\n")

print("Literal Table")
ltable.append(["Literal", "Address"])
for pair in literal_t:
    ltable.append([pair[0],pair[1]])
print(tabulate(ltable))
```

## Output:

```
C:\PD\TE\SEM 6\LABS\SPCC>python "Pass 1 Assembler.py"
MOVER BREG, FIVE       LC =  100
ADD BREF, ='4'      LC =  101
MOVEM BREG, TEMP      LC =  102
FIVE DC '5'      LC =  103
TEMP DS '1'      LC =  104
END



Mnemonic Table
```

## Symbol Table

| Symbol | Address |
| ------ | ------- |
| FIVE   | 103     |
| TEMP   | 104     |

```
Mnemonic Table
--------  ------  ------
Mnemonic  Opcode  Length
MOVER     4       1
MULTI     4       1
MOVEM     4       1
ADD       4       1
--------  ------  ------


Symbol Table
```

## Literal Table

| Literal | Address |
| ------- | ------- |
| 4       | 101     |

```
C:\PD\TE\SEM 6\LABS\SPCC>
```