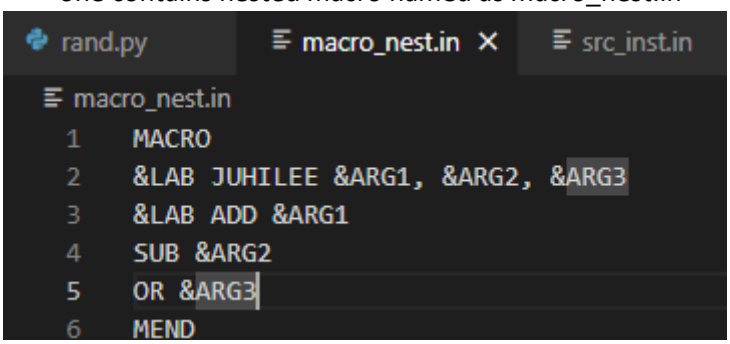
Program:-

```
import sys
macro_file = sys.argv[1]
program_file = sys.argv[2]
macro_cache = {}
with open(macro_file) as f:
   data = [i.strip() for i in f.readlines()]
   macro_state = False

   for i in range(len(data)):
      if data[i] == 'MACRO':
         i = i + 1

         if data[i].split(" ")[0].startswith("&"):
            label = data[i].split(" ")[0]
            mname = data[i].split(" ")[1]
            pholders = ''.join(data[i].split(" ")[2:]).split(',')
         else:
            label = None
            mname = data[i].split(" ")[0]
            pholders = ''.join(data[i].split(" ")[1:]).split(',')

         pholder = {}
         count = 0

         for j in pholders:
            pholder[j] = "{" + f"{count}" + "}"
            count += 1

         macro_cache[mname] = []

         i += 1

         while data[i] != 'MEND':
            for j in pholders:
               data[i] = data[i].replace(j, pholder[j], -1)

            if label != None:
               data[i] = data[i].replace(label, "{"+f"{count}"+"}")

            macro_cache[mname].append(data[i])
            i += 1

         i += 1

macro_calls = 0
src_inst = 0
macro_calls_inst = 0
total = 0
with open(program_file) as f:
   data = [i.strip() for i in f.readlines()]
```

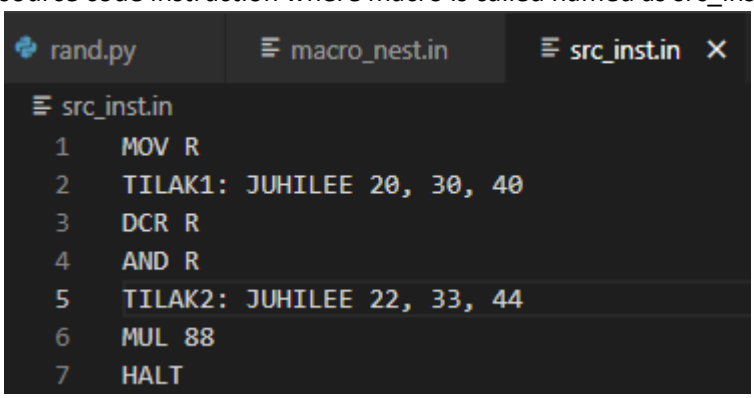
```
    for qwe in range(2):
        output = []

        for i in data:
            if len(i.split(" ")) > 1 and i.split(" ")[1] in macro_cache:
                macro_calls += 1
                macro_calls_inst = len(macro_cache[i.split(" ")[1]])
                output.append("")

                for j in macro_cache[i.split(" ")[1]]:
                    output.append(j.format(*''.join(i.split(" ")[2:]).split(","), i.split(" ")[0]))
                    total += 1

                output.append("")

            elif i.split(" ")[0] in macro_cache:
                macro_calls += 1
                macro_calls_inst = len(macro_cache[i.split(" ")[0]])
                output.append("")

                for j in macro_cache[i.split(" ")[0]]:
                    output.append(j.format(*''.join(i.split(" ")[1:]).split(",")))
                    total += 1

                output.append("")

            else:
                src_inst += 1
                output.append(i)
                total += 1

        data = output

for i in data:
    print(i)

print()
print(f"No. of instructions in input source code (excluding Macro calls): {src_inst}")
print(f"No. of macro calls: {macro_calls}")
print(f"No. of instructions in macro calls: {macro_calls_inst}")
print(f"Total instructions: {total}")
```

there are two input file

-one contains nested macro named as macro_nest.in



- Other is source code instruction where macro is called named as src_inst.in



Output:

```
PS D:\python with ML> python rand.py macro_nest.in src_inst.in
MOV R

TILAK1: ADD 20
SUB 30
OR 40

DCR R
AND R

TILAK2: ADD 22
SUB 33
OR 44

MUL 88
HALT

No. of instructions in input source code (excluding Macro calls): 20
No. of macro calls: 2
No. of instructions in macro calls: 3
Total instructions: 26
PS D:\python with ML>
```