Name: - Dev Chhawachharia
Roll Number:- 322014
PRN:- 22010381
TY B

Assignment 3

AIM: - Design suitable data structures and implement pass-I of a two-pass Macro processor.

Source code:

```
import pandas as pd
from itertools import chain

mdt = pd.DataFrame(columns=['Index', 'Description'])
mnt = pd.DataFrame(columns=['Index', 'Name', 'No of Arguments', 'Start
Index'])

temp_ala = []
mdtc = 1
mntc = 1
alac = 1
id_mnt = 0
id_mdt = 0
id = 0
my_file = open("input_macro.txt", "r")
data = my_file.read()
data_into_list = data.split("\n")
# print(data_into_list)
my_file.close()
macro_start_index = [i for i in range(len(data_into_list)) if
data_into_list[i] == "MACRO"]
# print(macro_start_index)
macro_stop_index = [i for i in range(len(data_into_list)) if
data_into_list[i] == "MEND"]
# print(macro_stop_index)
# print(type(data_into_list[2]))
for i in range(0, len(macro_start_index)):
    start = int(macro_start_index[i] + 1)
    stop = int(macro_stop_index[i])
    for x in range(start, stop + 1):
        # print(data_into_list[x])
        token = data_into_list[x].split()
        if x == start:
            # print(token)
            entry1 = (mntc, token[0], str(len(token) - 1), mdtc)
            mnt.loc[id_mnt] = entry1
            mntc += 1
            id_mnt += 1
            for j in range(1, len(token)):
                # print(token[j])
                res = token[j] in chain(*temp_ala)
                if not res:
```

```
                    temp_ala.append((token[j], "#" + str(alac)))
                    alac += 1
            continue
        if token[-1] in chain(*temp_ala):
            old = token[-1]
            ind = [x[0] for x in temp_ala].index(token[-1])
            new = temp_ala[ind][1]
            # print(new)
            data_into_list[x] = data_into_list[x].replace(old, new)
        if len(token) > 1:
            if token[-2] in chain(*temp_ala):
                old = token[-2]
                ind = [x[0] for x in temp_ala].index(token[-2])
                new = temp_ala[ind][1]
                # print(new)
                data_into_list[x] = data_into_list[x].replace(old, new)
        entry2 = (mdtc, data_into_list[x])
        mdt.loc[id_mdt] = entry2
        mdtc += 1
        id_mdt += 1

ala = pd.DataFrame(temp_ala, columns=['Name', 'Position'])
# print(mnt)
# print(ala)
# print(mdt)

f2 = open("mdt.txt", mode="wt")
dfasString = mdt.to_string(index=False)
f2.write(dfasString)

f3 = open("mnt.txt", mode="wt")
dfasString = mnt.to_string(index=False)
f3.write(dfasString)

f4 = open("ala.txt", mode="wt")
dfasString = ala.to_string(index=False)
f4.write(dfasString)
```

Input program:

```
START
MACRO
INCR &ARG1 &ARG2
ADD AREG &ARG1
MOVER BREG &ARG1
MEND
MACRO
PVG &ARG2
SUB AREG &ARG2
MOVER CREG &ARG1
MEND
INCR
END
```

Output:

Since this is the pass 1 of the macro processor we only create the macro description table(mdt) , macro name table(mnt) and argument list array(ala) and are passed on to pass 2.

MNT:

Index	Name	No of Arguments	Start Index
1	INCR	2	1
2	PVG	1	4

ALA:

Name	Position
&ARG1	#1
&ARG2	#2

MDT:

Index	Description
1	ADD AREG #1
2	MOVER BREG #1
3	MEND
4	SUB AREG #2
5	MOVER CREG #1
6	MEND

Note: No advance macro facility.