

PASS 1 CODE AND OUTPUT:

```
#include <iostream>
#include <conio.h>
#include <string.h>
using namespace std;
void lite(string lit[2][3])
{
    for(int l=0;
    l< 2;l++)
    {
        cout << "(DL, 02) (C, " << lit[l][1]<<")" << endl;
    }
}
void
chcklit(string
lit[2][3], string
l) { for (int i =
0; i < 2;
i++)
{
    if(l==lit
[i][1])
{
    cout <<"(L, " << i << ")";
    } } }
void chcksym(string sym[2][3], string s)
{
    int cons=0;
    i = 0; i < 2; i++)

if (s == sym[i][1])
{
    cout << "(S, " << i << ")";
    cons++;
}
if(cons==0)
{
    cout << "(C, " << s << ")";
}
}
void check(string MOT[8][3], string instr[8][5], string sym[2][3], string
lit[2][3])
{
    int i, j;    i =0;
while (i < 8)
{
    j = 0;    if (instr[i][j]=="origin")
    {
        cout << "(AD, 03)" << endl;
        i++;
    }
    else if (instr[i][j] == "ltorg")
    {lite(lit);
    i++;
    cout <<endl;
    }
    else
    {
        while (j < 5 && instr[i][j]!="origin" && instr[i][j]!="ltorg")
        {int m=0;
        while(m<8)
        {
```

```

if (instr[i][j] == MOT[m][0])
{
cout << "(" << MOT[m][1] << ", " << MOT[m][2] << ") ";
j++;

m=0
;
els
e
{
m++;
} }
if (instr[i][j] == "2" || instr[i][j] == "1")
{string litval
instr[i][j];chcklit(lit,litval);
j++;
}
else if (instr[i][j] == " " || instr[i][j] == "+" || instr[i][j] == "," ||
instr[i][j] == "=",)
{
j++;
continue;
}
else
{
string      symb =      instr[i][j];
chcksym(sym, symb);
j++;
i+  } }
+;  cout <<endl;
} }
void MOT(string instr[8][5], string sym[2][3], string lit[2][3])
{
string machinetable[8][3] = {"start", "AD", "01",
"mover", "IS", "04",
"breg", "RG", "02",
"areg", "RG", "01",
"add", "IS", "01",
"origin", "AD", "04",
"ltorg", "AD", "03",
"dc","DL","01"};
check(machinetable, instr, sym, lit);
}
void symbol(string s[2][3])
{
for (int i = 0; i < 2; i++)
{
for (int j = 0; j < 3; j++)
{
cout << "Enter symbol table";
cin >> s[i][j];
} } }
void litetable(string lit[2][3])
{ for (int i = 0; i < 2; i++)
{
for (int j = 0; j < 3; j++)
{
cout << "Enter literal table"; cin >> lit[i][j];
} } }
void display(string s[2][3], string lit[2][3])

```

```

{cout<<"-----"<<endl; cout << "Index "<< "
Symbol"<< " Location
Counter" << endl;
for (int i = 0; i <
2; i++)
{
for (int j = 0; j < 3; j++)
{
cout << " " << s[i][j] << " ";
}
cout << endl;
}
cout<<"-----"<<endl;
cout << "Index "<<" literal"<<" Location Counter" << endl;
for (int i = 0; i < 2; i++)
{
for (int j = 0; j < 3; j++)
cout << " " << lit[i][j] << " ";
}
cout << endl;
}
cout<<"-----"<<endl;
}
int main()
{
string
sym[2][3]; string
symbol(s);
display(s);string
let[8][5] = {"start",
"100"," "," "," "," ",
"mover", "breg",
",=", "2", " ",
"loop", "mover", "areg",
",", "n",
"add","breg", ",=", "1",
" ",
"ltorg", " ", " ", " ",
" ",
"n", "dc", "5", " ", " ",
" ",
"end", " ", " ", " ", " ",
"};
"origin", "loop", "+",
"5", " ",
for(int i=0; i<8;i++)
{
for(int j=0; j<5; j++)
{
cout<<let[i][j]<<" ";
}
cout<<endl;
}
symbol(sym);
litetable(lit);
display(sym, lit);
MOT(let, sym, lit);
return 0;
}

```

onlinegdb.com/online\_c++\_compiler

```

// litetable(li);
// display(s);
string let[s][5] = {"start", "100", " ", " ", " ",
                  "mover", "breg", " ", "2", " ",
                  "loop", "mover", "areg", " ", " ", "n"};

```

194

input

```

start 100
mover breg ,= 2
loop mover areg , n
add breg ,= 1
origin loop + 5
ltorg
n do 5
end
Enter symbol tableR
Enter symbol tableT
Enter symbol tableF
Enter symbol tableH
Enter symbol tableE
Enter symbol tableD
Enter literal table1
Enter literal table2
Enter literal table3
Enter literal table4
Enter literal table6
Enter literal table8

```

Index	Symbol	Location	Counter
R	T	F	
H	E	D	

Index	literal	Location	Counter
1	2	3	

Waiting for www.google.com...

onlinegdb.com/online\_c++\_compiler

main.cpp

```

190 // litetable(li);
191 // display(s);
192 string let[s][5] = {"start", "100", " ", " ", " ",
193                  "mover", "breg", " ", "2", " ",
194                  "loop", "mover", "areg", " ", " ", "n"};

```

input

```

Enter literal table2
Enter literal table3
Enter literal table4
Enter literal table6
Enter literal table8

```

Index	Symbol	Location	Counter
R	T	F	
H	E	D	

Index	literal	Location	Counter
1	2	3	
4	6	8	

```

(AD, 01) (C, 100)
(IS, 04) (RG, 02) (L, 0)
(C, loop) (IS, 04) (RG, 01) (C, n)
(IS, 01) (RG, 02)
(AD, 03)
(DL, 02) (C, 2)
(DL, 02) (C, 6)

(C, n) (DL, 01) (C, 5)
(C, end)

```

...Program finished with exit code 0

Waiting for www.google.com...