Exp. No. 11

Implement a C program to perform symbol table operations.

Program:

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int cnt=0;
struct symtab
{
char label[20];
int addr;
}
sy[50];
void insert();
int search(char *);
void display();
void modify();
int main()
{
int ch,val;
char lab[10];
do
{
printf("\n1.insert\n2.display\n3.search\n4.modify\n5.exit\n");
scanf("%d",&ch);
switch(ch)
{
case 1:
insert();
break;
case 2:
display();
break;
case 3:
printf("enter the label");
scanf("%s",lab);
val=search(lab);if(val==1)
printf("label is found");
else
printf("label is not found");
break;
case 4:
modify();
```

```
break;
case 5:
exit(0);
break;
}
}while(ch<5);
}
void insert()
{
int val;
char lab[10];
int symbol;
printf("enter the label");
scanf("%s",lab);
val=search(lab);
if(val==1)
printf("duplicate symbol");
else
{
strcpy(sy[cnt].label,lab);
printf("enter the address");
scanf("%d",&sy[cnt].addr);
cnt++;
}
}
int search(char *s)
{
int flag=0,i; for(i=0;i<cnt;i++)
{
if(strcmp(sy[i].label,s)==0)flag=1;
}
return flag;
}
void modify()
{
int val,ad,i;
char lab[10];
printf("enter the labe:");
scanf("%s",lab);
val=search(lab);
if(val==0)
printf("no such symbol");
else
```

```
{
printf("label is found \n");
printf("enter the address");
scanf("%d",&ad);
for(i=0;i<cnt;i++)
{
if(strcmp(sy[i].label,lab)==0)
sy[i].addr=ad;
}
}
}
void display()
{
int i;
for(i=0;i<cnt;i++)
printf("%s\t%d\n",sy[i].label,sy[i].addr);
}
```

Output

```
1.insert
2.display
3.search
4.modify
5.exit
1
enter the labela
enter the address100

1.insert
2.display
3.search
4.modify
5.exit
2
a   100
```

Output

```
1.insert
2.display
3.search
4.modify
5.exit
3
enter the labela
label is found
1.insert
2.display
3.search
4.modify
5.exit
4
enter the labe:a
label is found
enter the address200
```


Output
1.insert
2.display
3.search
4.modify
5.exit
2
a 200
1.insert
2.display
3.search
4.modify
5.exit
5
=== Code Execution Successful ===