Krishnaraj K
195002065

Ex. No:02

Symbol table

Code:

```
#include<stdio.h>
#include<string.h>
#include<conio.h>
#include<ctype.h>

FILE *fp;


char delim[18]={' ','\t','\n',',','(',')','[',']','{','}','#','+','-','*','/','%','=','!'};


char key[21][12]={"int","float","char","double","bool","void","extern","auto","bool","goto","static","class","struct","for","if","else","return","register","long","while","do"};


char ctype[12];


char avoid[5][12]={"include","define","getch","printf","scanf"};
struct symtab
{
	char id[20];
	char type[20];
}p[30];
int in=0;
void construct();
int isdelim(char);
void check(char[]);
int checkkey(char[]);
void showtable();

void main()
{
		char fname[12];
		//clrscr();
		printf("\nEnter the filename : ");
		scanf("%s",fname);
		fp=fopen(fname,"r");
		if(fp==NULL)
```

```
            printf("\nThe file doesn't exist.");
        else
        {
            construct();
            showtable();
        }
        fclose(fp);
        getch();
}
void construct()
{
        char c,ch,token[12];
        int f=0,j=0,kf=0;
        strcpy(ctype,"NULL");
        while(!feof(fp))
        {
            c=getc(fp);
            if(c==';'||c=='(')
            {
                if(f==1)
                {
                    token[j]='\0';
                    j=0;
                    f=0;
                    kf=checkkey(token);
                    if(kf==0)
                        check(token);
                }
                strcpy(ctype,"NULL");
            }
    else if(c=='"')
            {
                while((c=getc(fp))!='"');
            }
            else if(c=='<')
            {
                while((c=getc(fp))!='>');
            }
            else if(isdelim(c))
            {
                if(f==1)
                {
                    token[j]='\0';
                    j=0;
                    f=0;
                    kf=checkkey(token);
```

```
                  if(kf==0)
                      check(token);
              }
          }
          else if(isalpha(c)||c=='_')
          {
              token[j++]=c;
              f=1;
          }
        }
}

int isdelim(char c)
{
        int i;
        for(i=0;i<18;i++)
        {
          if(c==delim[i])
              return 1;
        }
        return 0;
}

int checkkey(char t[])
{
        int i;
        for(i=0;i<5;i++)
          if(strcmp(avoid[i],t)==0)
              return 1;
        for(i=0;i<21;i++)
          if(strcmp(key[i],t)==0)
          {
              strcpy(ctype,key[i]);
              return 1;
          }
        return 0;
}

void check(char t[])
{
        int i;
        for(i=0;i<in;i++)
        {

          if(((strcmp(t,p[i].id))==0)&&((strcmp(ctype,p[i].type))==0))
          {
```

```
                        printf("\nRedeclaration for '%s'",t);
                        return;
                }

                else
if(((strcmp(t,p[i].id))==0)&&((strcmp(ctype,p[i].type))!=0)&&((strcmp(ctype,"NULL")!=0))
)
                {
                        printf("\nMultiple declaration for %s",t);
                        return;
                }
            }

   if(strcmp(ctype,"NULL")==0)
            {
                for(i=0;i<in;i++)
                {
                        if(strcmp(t,p[i].id)==0)
                                return;
                }

                return;
            }
            strcpy(p[in].id,t);
            strcpy(p[in].type,ctype);
            in++;
}


void showtable()
{
            int i,ch=0;
            char type[10];
            if(in==0)
            {
                printf("\nSymbol table is empty.");
                return;
            }

            printf("\nSymbol table");
            printf("\n------------");
            printf("\nVariable\tType\t\Size");
            printf("\n----------\t----\t---");
            //printf("\nSize\t\tSize");
            //printf("\n------------");
            for(i=0;i<in;i++)
{

  ch=0;
  if(strcmp(p[i].id,"main")==0)
  {
    continue;
  }
  else
  {

                printf("\n%s\t\t%s",p[i].id,p[i].type);
                strcpy(type,p[i].type);
                while(ch!=1)
                {
  if(strcmp(type,"int")==0)
  {
    printf("\t%d",sizeof(int));
    break;
  }
  else if(strcmp(type,"char")==0)
  {
    printf("\t%d",sizeof(char));
    break;
  }
  else if(strcmp(type,"float")==0)
  {
    printf("\t%d",sizeof(float));
    break;
  }
  else if(strcmp(type,"long")==0)
  {
    printf("\t%d",sizeof(long));
    break;
  }
  else if(strcmp(type,"double")==0)
  {
    printf("\t%d",sizeof(double));
    break;
  }
  else
  {
    printf("not a data type");
  }
```

```
                    };

      }
}
}
```

Sample file

Hello.c

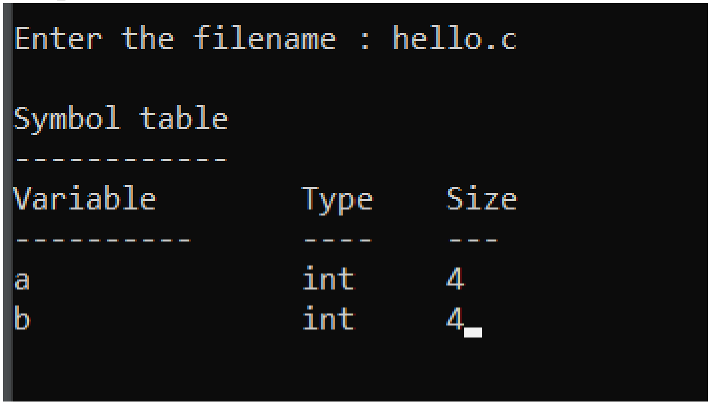
```
#include<stdio.h>

void main()
{

printf("hello");
int a=5,b;
b=a+5;


}
```

Output:



```
Enter the filename : hello.c

Symbol table
------------
Variable        Type    Size
----------      ----    ---
a               int     4
b               int     4
```

Result:

The symbol table has been implemented using C and their outputs were verified.