#include <iostream>

#include <string.h>

using namespace std;

class node

{

public:

char symb[20],type[20];

int size,line;

node \*lc,\*rc;

};

class sym

{

public:

node \*root;

sym()

{

root=NULL;

}

void create();

void insert(node \*r,node \*t);

void disp(node \*r);

void search(node \*r,char d[]);

void update(node \*r,char d[]);

};

void sym::create()

{

node \*temp;

char ch;

do

{

temp=new node;

cout<<"\nEnter Symbol-";

cin>>temp->symb;

cout<<"\nEnter Data Type-";

cin>>temp->type;

cout<<"\nEnter size-";

cin>>temp->size;

cout<<"\nEnter Line no.-";

cin>>temp->line;

temp->rc=NULL;

temp->lc=NULL;

if(root==NULL)

{

root=temp;

}

else

{

insert(root,temp);

}

cout<<"\nWant to enter more symbols ?(y/n)";

cin>>ch;

}while(ch=='y'||ch=='Y');

}

void sym::insert(node \*r,node \*t)

{

if(strcmp(t->symb,r->symb)<0)

{

if(r->lc==NULL)

{

r->lc=t;

}

else

{

insert(r->lc,t);

}

}

else

{

if(r->rc==NULL)

{

r->rc=t;

}

else

{

insert(r->rc,t);

}

}

}

void sym::disp(node \*r)

{

if(r!=NULL)

{

disp(r->lc);

cout<<"\n";

cout<<r->symb<<"\t "<<r->type<<"\t\t "<<r->size<<"\t "<<r->line;

disp(r->rc);

}

}

void sym::search(node \*r,char d[])

{

int flag=0,cnt=0;

while(r!=NULL)

{

cnt++;

if(strcmp(d,r->symb)==0)

{

cout<<"\nDatatype\t\tSize\t\tLine no";

cout<<"\n"<<r->symb<<"\t "<<r->type<<"\t\t "<<r->size<<"\t "<<r->line;

flag=1;

cout<<"\nComparisons Required:"<<cnt;

break;

}

if(strcmp(d,r->symb)<0)

r=r->lc;

if(strcmp(d,r->symb)>0)

r=r->rc;

}

if(flag==0)

cout<<"\nSymbol not Found!";

}

void sym::update(node \*r,char d[])

{

int flag=0;

while(r!=NULL)

{

if(strcmp(d,r->symb)==0)

{

cout<<"\nEnter the data type:";

cin>>r->type;

cout<<"\nEnter Size:";

cin>>r->size;

cout<<"\nEnter Line no:";

cin>>r->line;

cout<<"\nWord update successfull";

flag=1;

break;

}

if(strcmp(d,r->symb)<0)

r=r->lc;

if(strcmp(d,r->symb)>0)

r=r->rc;

}

if(flag==0)

cout<<"\nSymbol Not Found";

}

int main()

{

node \*NEW;

sym s;

char word[20];

int choice;

char ans;

do

{

cout<<"\nProgram for Symbol Table Operations";

cout<<"\n1.Create";

cout<<"\n2.Display";

cout<<"\n3.Search";

cout<<"\n4.Insert ";

cout<<"\n5.Update";

cout<<"\n Enter your choice ";

cin>>choice;

switch(choice)

{

case 1:

s.create();

break;

case 2:

if(s.root!=NULL)

{

cout<<"\nSymbol\tData Type \t Size \tLines";

s.disp(s.root);

}

else

{

cout<<"\nThere is no data to display";

}

break;

case 3:

if(s.root==NULL)

cout<<"\nNo Symbol to Search";

else

{

cout<<"\nEnter Symbol to search:";

cin>>word;

s.search(s.root,word);

}

break;

case 4:

if(s.root!=NULL)

{

NEW=new node;

cout<<"\nEnter Symbol-";

cin>>NEW->symb;

cout<<"\nEnter Data Type-";

cin>>NEW->type;

cout<<"\nEnter size-";

cin>>NEW->size;

cout<<"\nEnter Line no.-";

cin>>NEW->line;

s.insert(s.root,NEW);

cout<<"\nSymbol inserted Successfully";

}

else

cout<<"\nCreate Symbol table First";

break;

case 5:

if(s.root==NULL)

{

cout<<"\nNo Table to update";

}

else

{

cout<<"\nEnter symbol to update:";

cin>>word;

s.update(s.root,word);

}

break;

default:

cout<<"\nInvalid choice!";

break;

}

cout<<"\nReturn to Menu ?(y/n)";

cin>>ans;

}while(ans=='y'||ans=='Y');

}

/\* OUTPUT

Program for Symbol Table Operations

1.Create

2.Display

3.Search

4.Insert

5.Update

Enter your choice 1

Enter Symbol-!

Enter Data Type-exclamatory

Enter size-2

Enter Line no.-1

Want to enter more symbols ?(y/n)y

Enter Symbol-%

Enter Data Type-percent

Enter size-1

Enter Line no.-2

Want to enter more symbols ?(y/n)y

Enter Symbol-$

Enter Data Type-dollar

Enter size-3

Enter Line no.-3

Want to enter more symbols ?(y/n)n

Return to Menu ?(y/n)y

Program for Symbol Table Operations

1.Create

2.Display

3.Search

4.Insert

5.Update

Enter your choice 2

Symbol Data Type Size Lines

! exclamatory 2 1

$ dollar 3 3

% percent 1 2

Return to Menu ?(y/n)y

Program for Symbol Table Operations

1.Create

2.Display

3.Search

4.Insert

5.Update

Enter your choice 3

Enter Symbol to search:$

Datatype Size Line no

$ dollar 3 3

Comparisons Required:3

Return to Menu ?(y/n)y

Program for Symbol Table Operations

1.Create

2.Display

3.Search

4.Insert

5.Update

Enter your choice 4

Enter Symbol-\*

Enter Data Type-star

Enter size-4

Enter Line no.-4

Symbol inserted Successfully

Return to Menu ?(y/n)y

Program for Symbol Table Operations

1.Create

2.Display

3.Search

4.Insert

5.Update

Enter your choice 5

Enter symbol to update:\*

Enter the data type:astrik

Enter Size:5

Enter Line no:5

Word update successfull

Return to Menu ?(y/n)y

Program for Symbol Table Operations

1.Create

2.Display

3.Search

4.Insert

5.Update

Enter your choice 2

Symbol Data Type Size Lines

! exclamatory 2 1

$ dollar 3 3

% percent 1 2

\* astrik 5 5

Return to Menu ?(y/n)n

Process returned 0 (0x0) execution time : 272.074 s

Press any key to continue.\*/