* **OPEN HANDED PROBLEM**

**RE to NFA conversion**

#include<iostream.h>

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<ctype.h>

#include<stdlib.h>

#include<graphics.h>

char reg[20];

int T[20][4],i,j,l,k;

char a,b,c;

void display()

{

cout<<"\nNFA for given String :\n ";

cout<<"\n------------------------------------------------";

cout<<"\nFrom\tTo\tINPUT\t\tDISCRIPTION";

cout<<"\n------------------------------------------------"<<endl;

for(i=0;i<=j;i++)

{

if(T[i][0]!=0)

{

cout<<i<<"\t"<<T[i][0]<<"\t"<<a;

cout<<endl;

// cout<<"\t\tT["<<i<<","<<a<<"]->"<<T[i][0]<<endl;

}

if(T[i][1]!=0)

{

cout<<i<<"\t"<<T[i][1]<<"\t"<<b<<endl;

// cout<<"\t\tT["<<i<<","<<b<<"]->"<<T[i][1]<<endl;

}

if(T[i][2]!=0)

{

if(T[i][2]<10)

{

cout<<i<<"\t"<<T[i][2]<<"\tNULL(e)"<<endl;

// cout<<" \tT["<<i<<",e]->"<<T[i][2]<<endl;

}

else

{

cout<<i<<"\t"<<T[i][2]/10<<","<<T[i][2]%10<<"\tNULL(e)"<<endl;

// cout<<"\t\tT["<<i<<",e]->"<<T[i][2]/10<<","<<T[i][2]%10<<endl;

}

}

if(T[i][3]!=0)

{

cout<<i<<"\t"<<T[i][3]<<"\t#"<<endl;

// cout<<"\t\tT["<<i<<",#]->"<<T[i][3]<<endl;

}

}

cout<<"------------------------------------------------";

}

void display1()

{

int gdriver =DETECT,gmode,errorcode;

initgraph(&gdriver, &gmode,"C:\\TURBOC3\\BGI");

int ycor=7,ycor1;

char output[50],output1[50];

setbkcolor(14);

for(i=0;i<=j;i++)

{

ycor1=ycor-4;

if(T[i][0]!=0)

{

sprintf(output,"%d",i);

setcolor(4);

outtextxy(10,ycor1,output);

setcolor(1);

line(20,ycor,50,ycor);

setcolor(4);

sprintf(output1,"%d",T[i][0]);

outtextxy(60,ycor1,output1);

}

if(T[i][1]!=0)

{

sprintf(output,"%d",i);

setcolor(4);

outtextxy(10,ycor1,output);

setcolor(1);

line(20,ycor,50,ycor);

sprintf(output1,"%d",T[i][1]);

setcolor(4);

outtextxy(60,ycor1,output1);

}

if(T[i][2]!=0)

{

if(T[i][2]<10)

{ //ycor+=15;

sprintf(output,"%d",i);

setcolor(4);

outtextxy(10,ycor1,output);

setcolor(1);

line(20,ycor,50,ycor);

sprintf(output1,"%d",T[i][2]);

setcolor(4);

outtextxy(60,ycor1,output1);

}

else

{

sprintf(output,"%d",i);

setcolor(4);

outtextxy(10,ycor1,output);

setcolor(1);

line(20,ycor,50,ycor);

sprintf(output1,"%d",T[i][2]/10);

setcolor(4);

outtextxy(60,ycor1,output1);

sprintf(output1,"%s",",");

outtextxy(66,ycor1,output1);

sprintf(output1,"%d",T[i][2]%10);

outtextxy(72,ycor1,output1);

}

}

if(T[i][3]!=0)

{

sprintf(output,"%d",i);

setcolor(4);

outtextxy(10,ycor1,output);

setcolor(1);

line(20,ycor,50,ycor);

sprintf(output1,"%d",T[i][3]);

setcolor(4);

outtextxy(60,ycor1,output1);

}

ycor+=16;

}

getch();

//closegraph();

}

void main()

{

clrscr();

for(i=0;i<20;i++)

{

for(j=0;j<4;j++)

{

T[i][j]=0;

}

}

cout<<"\nEnter String : ";

cin>>reg;

l=strlen(reg);

for(i=0;i<l;i++)

{

if(isalnum(reg[i]))

{

a=reg[i];

break;

}

}

for(j=i+1;j<l;j++)

{

if(isalnum(reg[j]) && reg[j]!=a)

{

b=reg[j];

break;

}

}

i=0;

j=1;

while(i<l)

{

if(reg[i]==a & reg[i+1]!='|' & reg[i+1]!='\*'& reg[i+1]!='+')

{

T[j][0]=j+1;

j++;

}

if(reg[i]==b & reg[i+1]!='|'& reg[i+1]!='\*')

{

T[j][1]=j+1;

j++;

}

if(reg[i]=='e'& reg[i+1]!='|' & reg[i+1]!='\*')

{

T[j][2]=j+1;

j++;

}

if(reg[i]=='#' & reg[i+1]!='|' & reg[i+1]!='\*')

{

T[j][3]=j+1;

j++;

}

if(reg[i]==a & reg[i+1]=='|'& reg[i+2]==b)

{

T[j][2]=((j+1)\*10)+(j+3);

j++;

T[j][0]=j+1;

j++;

T[j][2]=j+3;

j++;

T[j][1]=j+1;

j++;

T[j][2]=j+1;

j++;

i=i+2;

}

if(reg[i]==b & reg[i+1]=='|'& reg[i+2]==a)

{

T[j][2]=((j+1)\*10)+(j+3);

j++;

T[j][1]=j+1;

j++;

T[j][2]=j+3;

j++;

T[j][0]=j+1;

j++;

T[j][2]=j+1;

j++;

i=i+2;

}

if(reg[i]==a & reg[i+1]=='\*')

{

T[j][2]=((j+1)\*10)+(j+3);

j++;

T[j][0]=j+1;

j++;

T[j][2]=((j+1)\*10)+(j-1);

j++;

}

if(reg[i]==a & reg[i+1]=='+')

{

T[j][0]=j+1;

j++;

T[j][2]=((j+1)\*10)+(j+3);

j++;

T[j][0]=j+1;

j++;

T[j][2]=((j+1)\*10)+(j-1);

j++;

}

if(reg[i]==b & reg[i+1]=='\*')

{

T[j][2]=((j+1)\*10)+(j+3);

j++;

T[j][1]=j+1;

j++;

T[j][2]=((j+1)\*10)+(j-1);

j++;

}

if(reg[i]==')'&reg[i+1]=='\*')

{

T[0][2]=((j+1)\*10)+1;

T[j][2]=((j+1)\*10)+1;

j++;

}

i++;

}

display();

cout<<"\n\nDo u wann see graph?";

char q;

cin>>q;

if(q=='y'||q=='Y')

display1();

closegraph();

getch();

}

**output**



