/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

PROGRAM DESCRIPTION:THIS PROGRAM CALCULATES INVERSE KINEMATICS FOR A TWO

AXIS ROBOT.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include<stdio.h>

#include<conio.h>

#include<GRAPHICS.h>

#include<math.h>

#include<ctype.h>

#include<process.h>

#define pi 22.0/7.0

void main(void)

{

float t3,t1,t2,l[4],l1[6];

char message[100],choice;

int gd,gm,n,a,b,y,z,x,m;

gd=DETECT;

l[0]=0;

initgraph(&gd,&gm,"\\tc\\bgi");

setcolor(YELLOW);rectangle(60,80,590,410); rectangle(70,70,580,420);

setcolor(YELLOW);settextstyle(3,0,3);

outtextxy(90,90,"INVERSE KINEMATIC PARAMETER PROBLEM OF");

outtextxy(140,130,"TWO AXES ARTICULATED ROBOT");

getch();

// exit(0);

clearviewport();

setcolor(RED);settextstyle(3,0,4);outtextxy(30,30,"INSTRUCTIONS");

setcolor(WHITE);settextstyle(6,0,3);

outtextxy(30,100,"1. To enter the two link lengths");

outtextxy(30,130,"2. To enter the Tool Configuration Vectors");

outtextxy(30,160,"3. Output the TCV in the matrix format");

outtextxy(30,190,"4. Output the base angle and the shoulder angle");

outtextxy(30,220,"5. End");

getch();

clearviewport();

//setcolor(BLUE);rectangle(60,90,580,440);setfillstyle(10,BLUE);

// floodfill(3,3,1);setcolor(WHITE);

outtextxy(145,30,"INVERSE KINEMATICS PROBLEM");

rectangle(290,170,350,210);

line(260,190,290,190);line(284,187,290,190);line(284,193,290,190);

line(320,210,320,240);line(317,217,320,210);line(323,217,320,210);

line(350,190,380,190);line(374,187,380,190);line(374,193,380,190);

settextstyle(SMALL\_FONT,0,4);

outtextxy(400,179,"POSITION &");outtextxy(186,189,"ORIENTATION");

outtextxy(400,179,"GEOMETRIC LINK");outtextxy(395,189,"PARAMETERS");

setcolor(CYAN);settextstyle(TRIPLEX\_FONT,0,1);outtextxy(299,178,"I.K.P");

settextstyle(2,0,6);setcolor(11);

outtextxy(150,300,"THE LINK AND JOINT PARAMETERS ARE:-");

setcolor(WHITE);settextstyle(2,0,5);

outtextxy(120,375,"THE ARM POSITION -P ARM ORIENTATION -R");

settextstyle(SMALL\_FONT,0,5);

outtextxy(420,450,"PRESS ANY KEY TO CONTINUE...");

getch();

cleardevice();

settextstyle(1,0,3);

outtextxy(130,50,"ENTER GEOMETRIC LINK PARAMETERS");

setcolor(YELLOW);

settextstyle(2,0,5);

printf("\n\n\n\n\n\n\n\n\n");

for(n=1;n<3;n++)

{

if(n==1)

sprintf(message,"ENTER BASE LINK LENGTH");else

{

if(n==2)

sprintf(message,"ENTER SHOULDER LINK LENGTH");

// else

// {

// sprintf(message,"ENTER TOOL LENGTH");

// }

}

outtextxy(210,115+(46\*n),message);

printf("\n\n\t\t\t\t ");

scanf("%f",&l[n]);

}

closegraph();

initgraph(&gd,&gm,"\\tc\\bgi");

settextstyle(1,0,3);

outtextxy(120,50,"ENETR TOOL CONFIGURATION VECTOR");setcolor(LIGHTRED);

outtextxy(270,90,"(REDUCED)");

setcolor(YELLOW);

settextstyle(2,0,5);

gotoxy(0,0);

printf("\n\n\n\n\n\n\n\n\n");

m=1;

for(a=1;a<4;++a)

{

if(a==3) m=6;

sprintf(message,"ENTER TCV PARAMETER W%d",m);

outtextxy(240,113+(48\*a),message);

printf("\n\n\t\t\t\t\t");

scanf("%f",&l1[m]);

m=a+1;

}

l1[3]=0; l1[4]=0;l1[5]=0;

cleardevice();

settextstyle(1,0,3);

setcolor(WHITE);outtextxy(170,50,"THE CONFIGURATION VECTOR");

setcolor(14);line(200,150,200,400);line(250,150,250,400);

line(200,150,210,150);line(200,400,210,400);line(240,150,250,150);line(240,400,250,400);

line(350,150,350,400);line(500,150,500,400);

line(360,150,350,150);line(360,400,350,400);line(490,150,500,150);line(490,400,500,400);

settextstyle(2,0,6);setcolor(WHITE);

outtextxy(130,360,"W(q)= =");

for(n=1;n<7;++n)

{

sprintf(message,"W%d %f",n,l1[n]);

outtextxy(210,160+((n-1)\*40),message);

}

settextstyle(SMALL\_FONT,0,5);

outtextxy(420,450,"PRESS ANY KEY TO CONTINUE....");

getch();

cleardevice();

t2=acos((l1[1]\*l1[1]+l1[2]\*l1[2]-l[1]\*l[1]-l[2]\*l[2])/(2\*l[1]\*l[2]));

t1=atan2(((l[1]+l[2]\*cos(t2))\*l1[2]-l[1]\*sin(t2)\*l1[1]),(l[1]+l[2]\*cos(t2))\*l1[1]+l[2]\*sin(t2)\*l1[2]);

t1=t1\*1260/22;

t2=t2\*1260/22;

// t3=log(l1[6]);

settextstyle(2,0,5);

setcolor(YELLOW);

// rectangle(140,150,520,360); rectangle(143,153,517,357);

// setfillstyle(1,BLUE);floodfill(144,160,YELLOW);

for(n=1;n<3;n++)

{

if(n==1)

sprintf(message," BASE ROTATION ANGLE IS %0.00f",t1);else

{

if(n==2)

sprintf(message,"SHOULDER ROTATION ANGLE IS %0.00f",t2);

// else

// {

// sprintf(message," TOOL ROTATION ANGLE IS %0.00f",t3);

// }

}

outtextxy(200,153+(46\*n),message);

}

setcolor(LIGHTGRAY);// setfillstyle(1,DARKGRAY),fillellipse(325,75,200,30);

// setfillstyle(1,RED);

settextstyle(1,0,3);setcolor(RED);

// fillellipse(325,64,200,30);

// setcolor(WHITE);

outtextxy(170,50,"THE SOLUTION TO IKP");

// ellipse(325,64,0,360,202,32);

getch();

}





