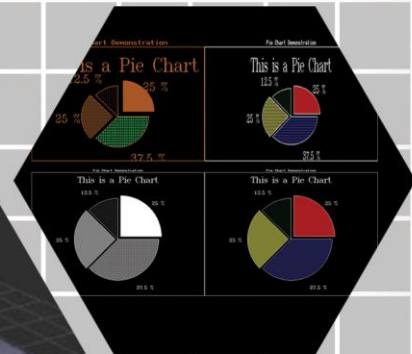
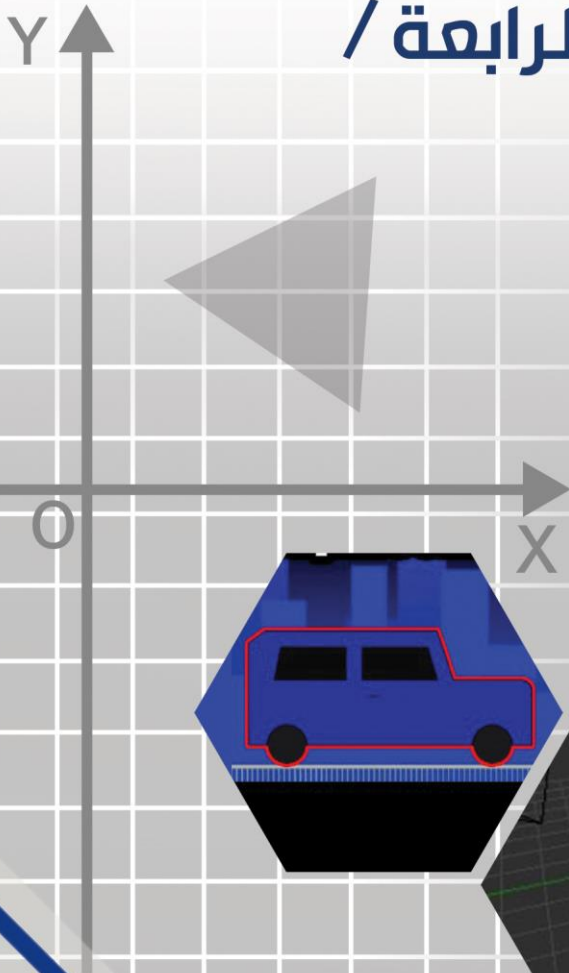




الجمهورية العربية السورية
جامعة البعث
الكلية التطبيقية
قسم تقنيات الحاسوب

بيانات الحاسوب

/ السنة الرابعة /



إعداد : م. فراس عباره

المحاضرة الخامسة:

في هذه المحاضرة سوف نتعلم كيفية استخدام التحويلات الهندسية في مكتبة Graphics.h.

التحويلات الهندسية:

التحويلات الهندسية هي عمليات رياضية يتم من خلالها تغيير موقع أو شكل الأشكال الهندسية في الفضاء.

لها تطبيقات واسعة في الرياضيات، الفيزياء، ورسوم الكمبيوتر. هناك عدة أنواع رئيسية من التحويلات الهندسية، وهي تشمل:

❖ الدوران Rotation:

الدوران هو أحد التحويلات الهندسية التي تدور فيها كل نقطة من شكل ما حول نقطة ثابتة تسمى مركز الدوران بزاوية محددة واتجاه محدد (مع عقارب الساعة أو عكسها).

✚ عناصر الدوران:

➤ مركز الدوران Center of Rotation:

النقطة الثابتة التي يدور حولها الشكل.

➤ زاوية الدوران Angle of Rotation:

مقدار الدوران وتُقاس بالدرجات أو الراديان.

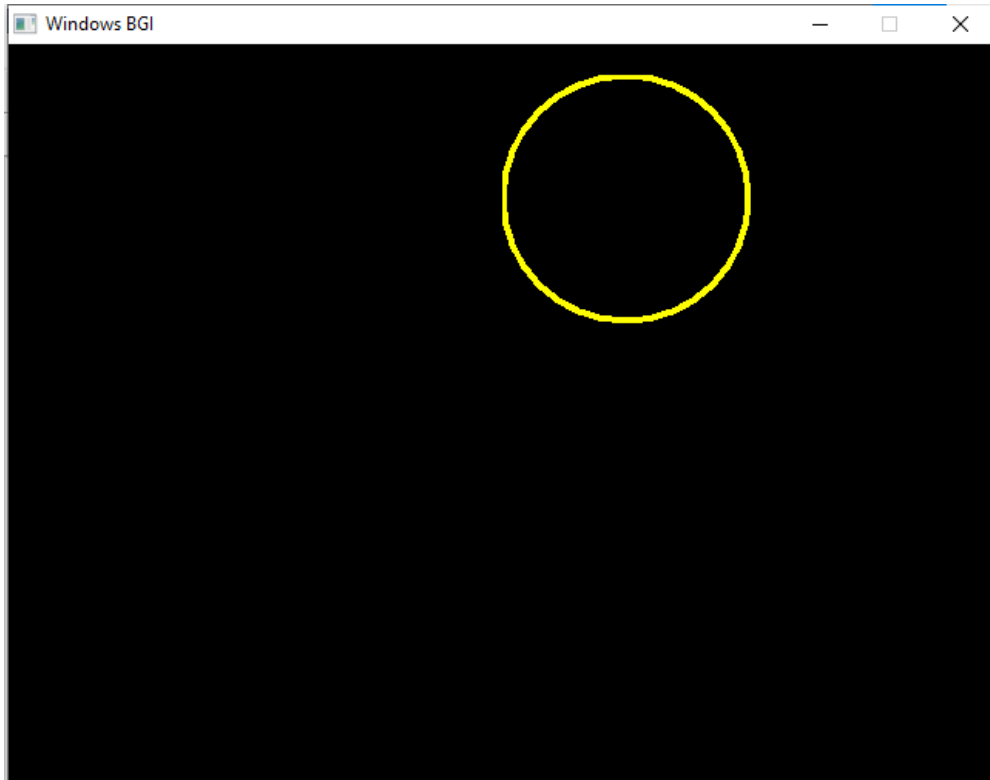
➤ اتجاه الدوران Direction:

عكس عقارب الساعة ← زاوية موجبة.

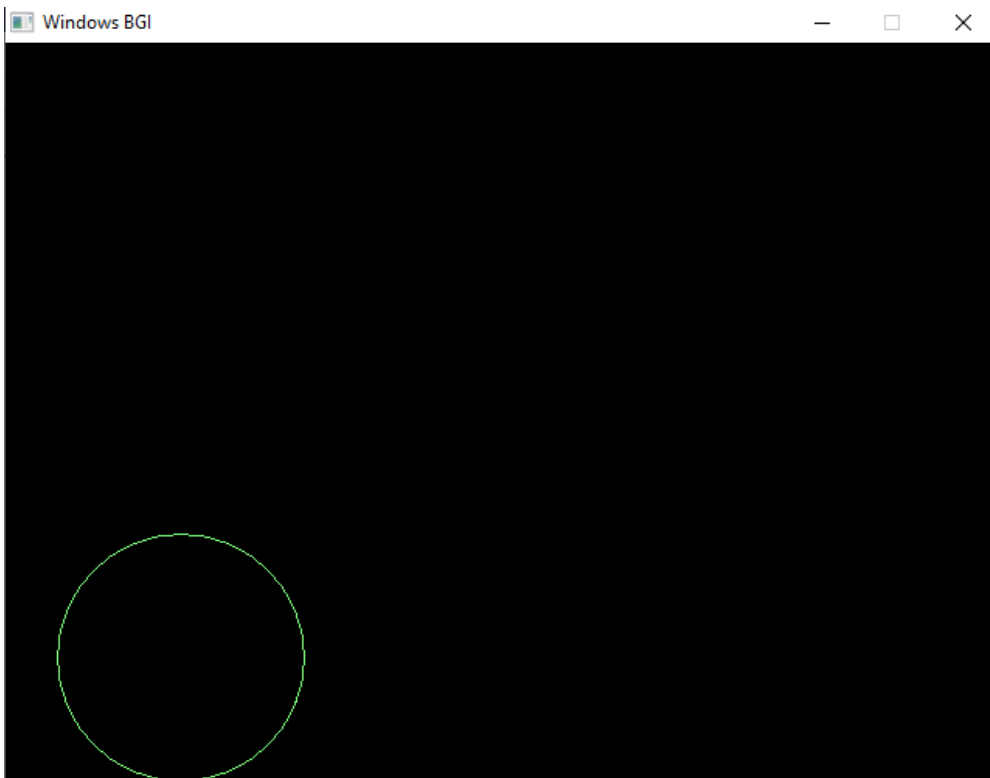
مع عقارب الساعة ← زاوية سالبة.

التمرين الأول: برنامج يقوم برسم دائرة، ثم إجراء عملية الدوران بمقدار 60 درجة حول مركز الإحداثيات X,Y.

```
1  #include<iostream>
2  #include<graphics.h>
3  #include<math.h>
4  using namespace std;
5  int main(){
6      int gd=DETECT,gm;
7      initgraph(&gd, &gm, (char *)"");
8      long x1=400,y1=100;
9      setlinestyle(0,0,4);
10     setcolor(14);
11     circle(x1,y1,80);
12     delay(1000);
13     for(int i=1;i<=60;i++){
14         double a=(i*3.14)/180;
15         int x=cos(a)*(x1)-sin(a)*(y1);
16         int y=sin(a)*(x1)+cos(a)*(y1);
17         delay(100);
18         cleardevice();
19         setcolor(10);
20         setlinestyle(0,0,1);
21         circle(x,y,80);
22     }
23     getch();
24     closegraph();
25     return 0;
26 }
```



الدوران بمقدار 60 درجة حول مركز الإحداثيات X, Y .



إزالة تعليمة `cleardevice()` من الكود.



إزالة تعليمة `for` من الكود.



التمرين الثاني: برنامج يقوم برسم خط مستقيم، ثم إجراء عملية الدوران بمقدار 55- درجة حول مركز الإحداثيات X,Y.


```
1  #include<iostream>
2  #include<graphics.h>
3  #include<math.h>
4  using namespace std;
5  int main(){
6      int gd=DETECT,gm;
7      initgraph(&gd, &gm, (char *)"");
8      long x1=150,y1=400,x2=50,y2=450;
9      setlinestyle(0,0,6);
10     setcolor(10);
11     line(x1,y1,x2,y2);
12     delay(1000);
13     double a=-(55*3.14)/180;
14     int x=cos(a)*(x1)-sin(a)*(y1);
15     int y=sin(a)*(x1)+cos(a)*(y1);
16
17     int xx=cos(a)*(x2)-sin(a)*(y2);
18     int yy=sin(a)*(x2)+cos(a)*(y2);
19     delay(100);
20
21     setcolor(7);
22     setlinestyle(0,0,3);
23     line(x,y,xx,yy);
24     getch();
25     closegraph();
26     return 0;
27 }
```

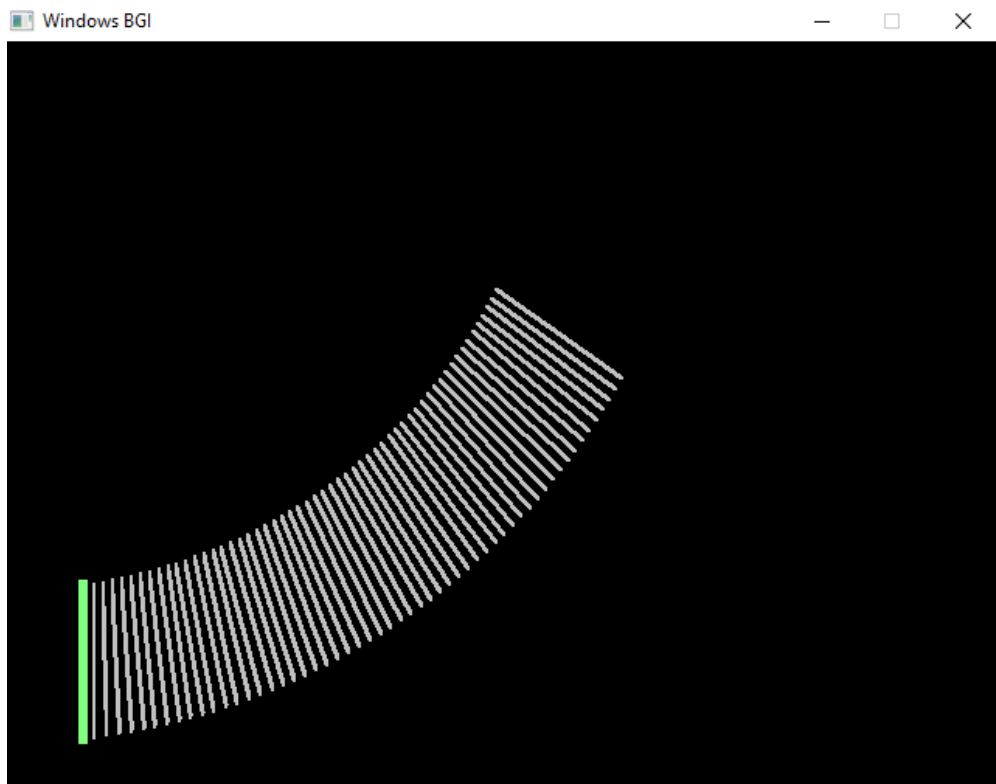
تنشيط Windows



نفس التمرين لكن مع إضافة حلقة for.

```
1  #include<iostream>
2  #include<graphics.h>
3  #include<math.h>
4  using namespace std;
5  int main(){
6      int gd=DETECT, gm;
7      initgraph(&gd, &gm, (char *)"");
8      long x1=50, y1=350, x2=50, y2=450;
9      setlinestyle(0, 0, 6);
10     setcolor(10);
11     line(x1, y1, x2, y2);
12     delay(1000);
13     for(int i=1; i<=55; i++){
14         double a=-(i*3.14)/180;
15         int x=cos(a)*(x1)-sin(a)*(y1);
16         int y=sin(a)*(x1)+cos(a)*(y1);
17
18         int xx=cos(a)*(x2)-sin(a)*(y2);
19         int yy=sin(a)*(x2)+cos(a)*(y2);
20         delay(100);
21         //cleardevice();
22         setcolor(7);
23         setlinestyle(0, 0, 3);
24         line(x, y, xx, yy);
25     }
26     getch();
27     closegraph();
28     return 0;
29 }
```





التمرين الثالث: برنامج يقوم برسم مربع، ثم إجراء عملية الدوران بمقدار 90 درجة حول مركز إحداثيات الجسم المربع $x=200$ & $y=200$.

```

1  #include<iostream>
2  #include<graphics.h>
3  #include<math.h>
4  using namespace std;
5  int main()
6  {
7      int gd=DETECT, gm;
8      initgraph(&gd, &gm, (char *)"");
9      long x1=100, y1=100, x2=300, y2=100,
10     x3=300, y3=300, x4=100, y4=300;
11     setcolor(14);
12     setlinestyle(0,0,4);
13     line(x1,y1,x2,y2);
14     line(x2,y2,x3,y3);
15     line(x3,y3,x4,y4);
16     line(x4,y4,x1,y1);
17     delay(2000);
18     for(int i=1; i<=45; i++){
19         double a=(i*3.14)/180;
20
21         int x11=200+cos(a)*(x1-200)-sin(a)*(y1-200);
22         int y11=200+sin(a)*(x1-200)+cos(a)*(y1-200);
23
24         int x21=200+cos(a)*(x2-200)-sin(a)*(y2-200);
25         int y21=200+sin(a)*(x2-200)+cos(a)*(y2-200);
26
27         int x31=200+cos(a)*(x3-200)-sin(a)*(y3-200);
28         int y31=200+sin(a)*(x3-200)+cos(a)*(y3-200);
29
30         int x41=200+cos(a)*(x4-200)-sin(a)*(y4-200);
31         int y41=200+sin(a)*(x4-200)+cos(a)*(y4-200);
32
33         setcolor(10);
34         setlinestyle(1,0,2);
35         line(x11,y11,x21,y21);
36         line(x21,y21,x31,y31);
37         line(x31,y31,x41,y41);
38         line(x41,y41,x11,y11);
39         delay(20);
40     }
41     getch();
42     closegraph();
43     return 0;
44 }

```

```

#include<iostream>
#include<graphics.h>
#include<math.h>
using namespace std;
int main(){
int gd=DETECT,gm;
initgraph(&gd, &gm, (char *)"");
long x1=100,y1=100,x2=300,y2=100,
x3=300,y3=300,x4=100,y4=300;
setcolor(14);
setlinestyle(0,0,4);
line(x1,y1,x2,y2);
line(x2,y2,x3,y3);
line(x3,y3,x4,y4);
line(x4,y4,x1,y1);
delay(2000);

for(int i=1;i<=45;i++){
double a=(i*3.14)/180;

int x11=200+cos(a)*(x1-200)-sin(a)*(y1-200);
int y11=200+sin(a)*(x1-200)+cos(a)*(y1-200);

```

```
int x21=200+cos(a)*(x2-200)-sin(a)*(y2-200);
```

```
int y21=200+sin(a)*(x2-200)+cos(a)*(y2-200);
```

```
int x31=200+cos(a)*(x3-200)-sin(a)*(y3-200);
```

```
int y31=200+sin(a)*(x3-200)+cos(a)*(y3-200);
```

```
int x41=200+cos(a)*(x4-200)-sin(a)*(y4-200);
```

```
int y41=200+sin(a)*(x4-200)+cos(a)*(y4-200);
```

```
setcolor(10);
```

```
setlinestyle(1,0,2);
```

```
line(x11,y11,x21,y21);
```

```
line(x21,y21,x31,y31);
```

```
line(x31,y31,x41,y41);
```

```
line(x41,y41,x11,y11);
```

```
delay(20);
```

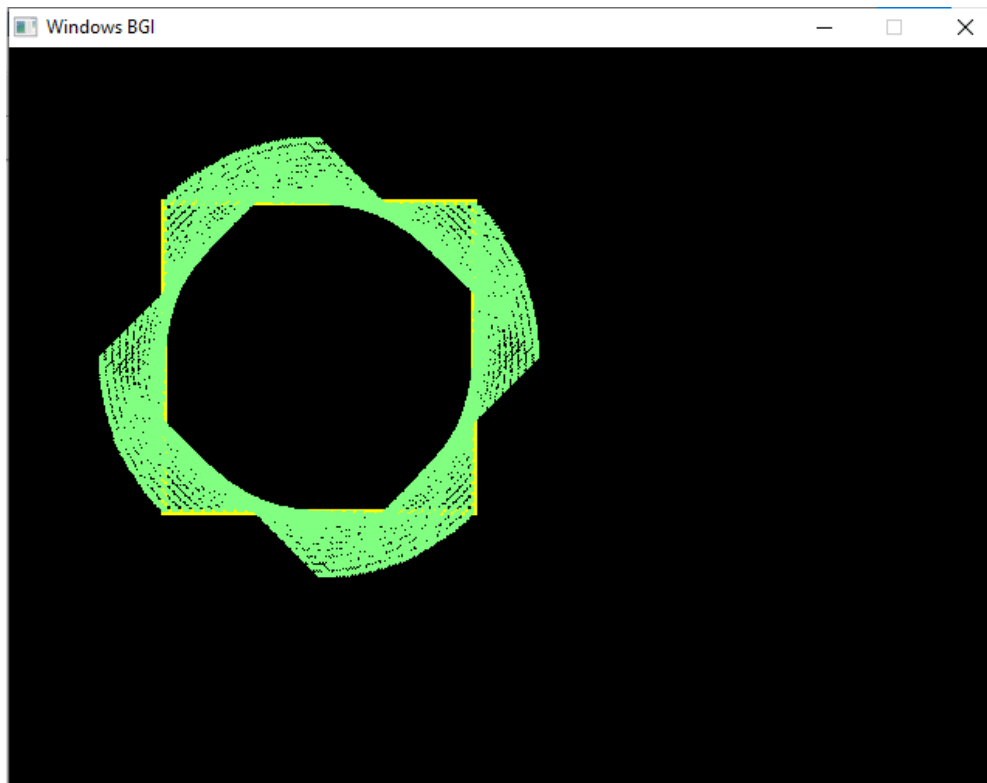
```
{
```

```
getch();
```

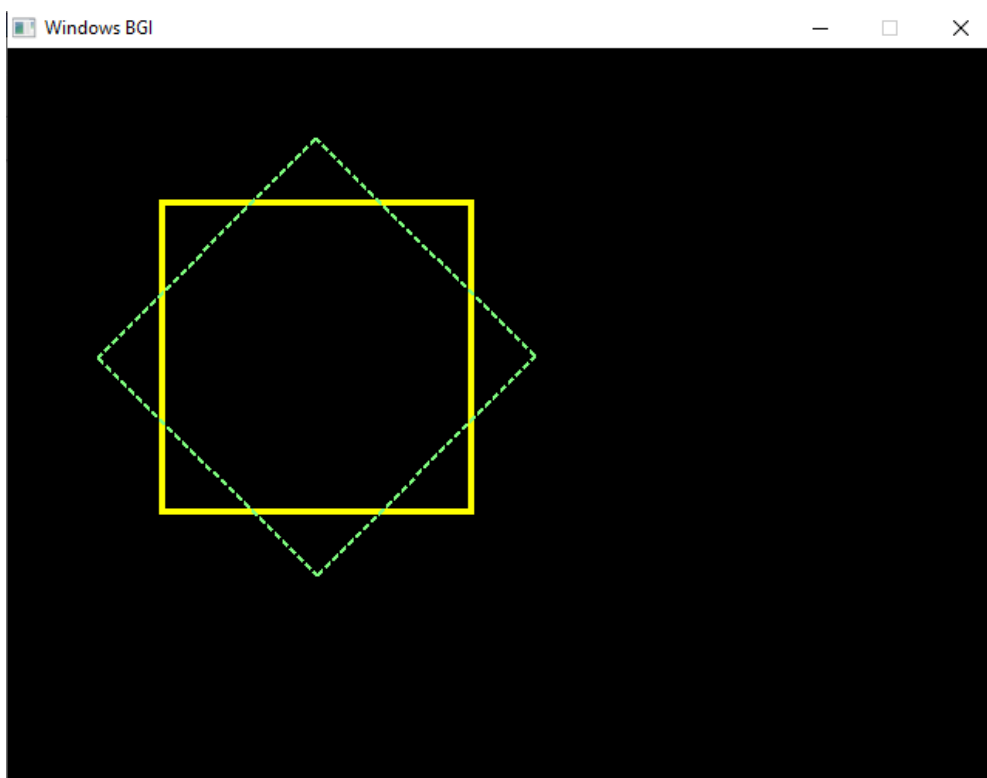
```
closegraph();
```

```
return 0;
```

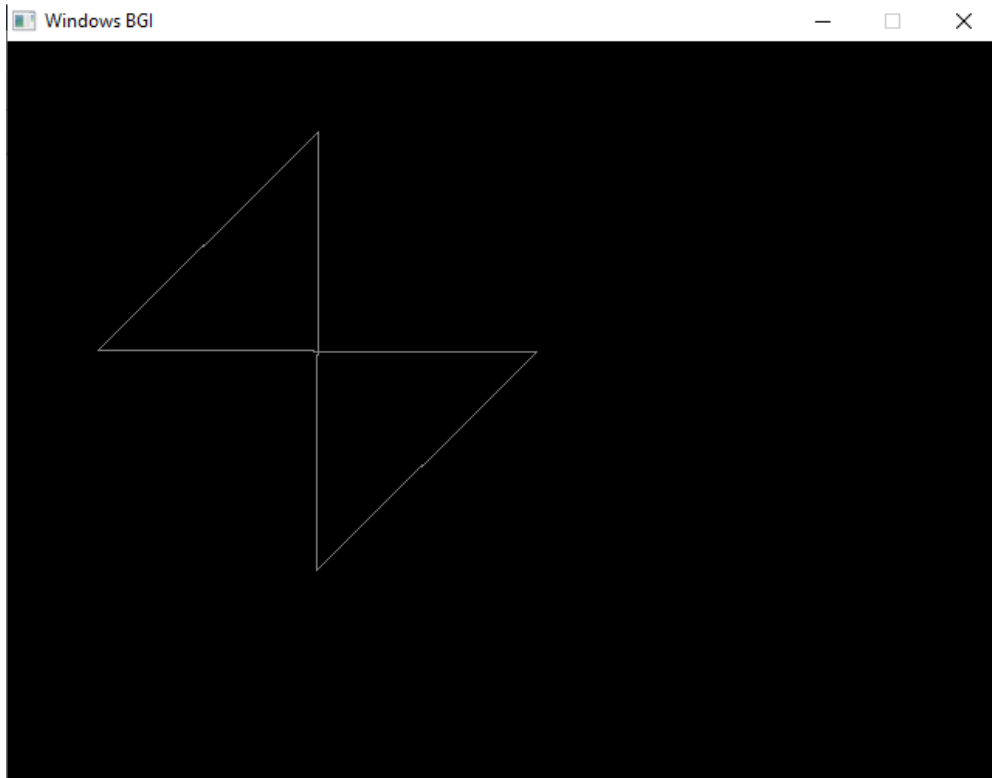
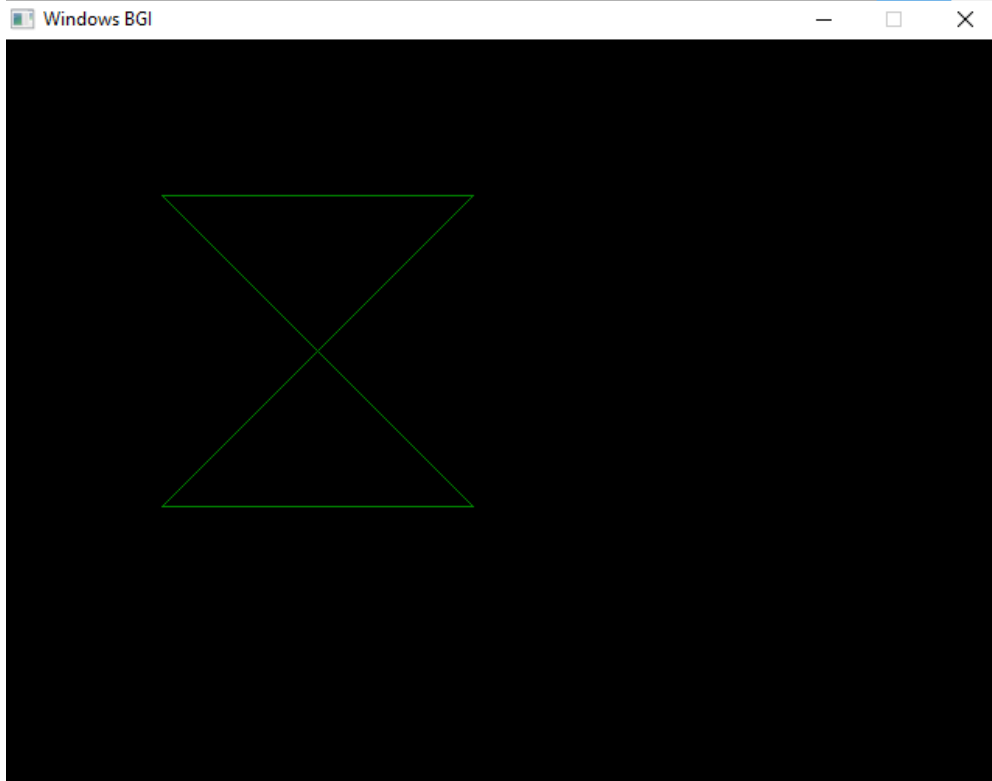
```
{
```



إزالة تعليمة for من الكود.



التمرين الرابع: ارسم الشكل التالي باستخدام الخطوط، ثم قم
بإجراء عملية الدوران بمقدار -45 درجة حول مركز إحداثيات
الجسم المربع $x=200$ & $y=200$.

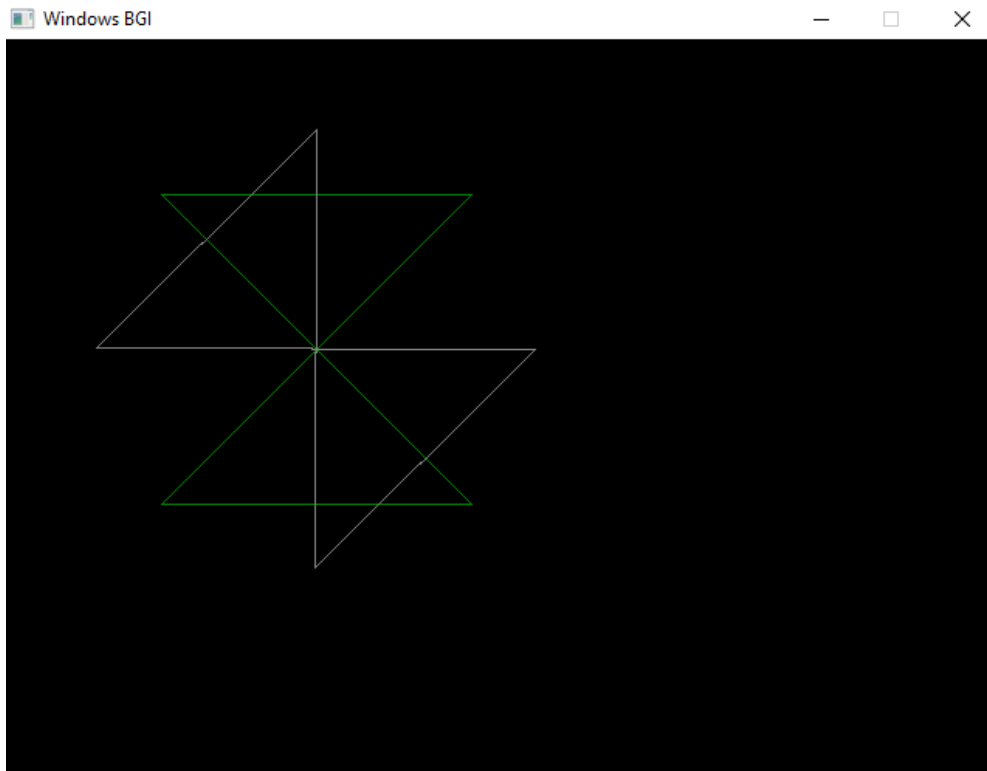


```

1  #include<iostream>
2  #include<graphics.h>
3  #include<math.h>
4  using namespace std;
5  int main(){
6  int gd=DETECT, gm;
7  initgraph(&gd, &gm, (char *)"");
8  long x1=100,y1=100,x2=300,y2=100,
9  x3=100,y3=300,x4=300,y4=300;
10 setcolor(2);
11 line(x1,y1,x2,y2);
12 line(x2,y2,x3,y3);
13 line(x3,y3,x4,y4);
14 line(x4,y4,x1,y1);
15
16 for(int i=1;i<=45;i++){
17 double a=-(i*3.14)/180;
18 int x11=200+cos(a)*(x1-200)-sin(a)*(y1-200);
19 int y11=200+sin(a)*(x1-200)+cos(a)*(y1-200);
20
21 int x21=200+cos(a)*(x2-200)-sin(a)*(y2-200);
22 int y21=200+sin(a)*(x2-200)+cos(a)*(y2-200);
23
24 int x31=200+cos(a)*(x3-200)-sin(a)*(y3-200);
25 int y31=200+sin(a)*(x3-200)+cos(a)*(y3-200);
26
27 int x41=200+cos(a)*(x4-200)-sin(a)*(y4-200);
28 int y41=200+sin(a)*(x4-200)+cos(a)*(y4-200);
29
30 setcolor(8);
31 line(x11,y11,x21,y21);
32 line(x21,y21,x31,y31);
33 line(x31,y31,x41,y41);
34 line(x41,y41,x11,y11);
35 delay(10);
36 cleardevice();
37 }getch();closegraph();
38 return 0;
39

```

تنشيط windows



إزالة تعليمة `cleardevice()` من الكود.

