



Experiment 3

Student Name: Gaganjot Singh

UID: 22BCS14843

Branch: B.E. CSE III Yr

Section: 22BCS-IOT-612-B

Semester: 6th

Subject Name: Computer Graphics with Lab

Subject Code: 22CSH-352

1. Aim: Apply translation, scaling, and rotation transformations on a given triangle and observe the changes.

2. Objective: To apply geometric transformations such as translation, scaling, and rotation on a given triangle.

3. Code:

```
#include <iostream.h>
```

```
#include <graphics.h>
```

```
#include <conio.h>
```

```
#include <math.h>
```

```
void Draw(int x1, int y1, int x2, int y2, int x3, int y3, int color)
```

```
{    setcolor(color);
```

```
    line(x1, y1, x2, y2);
```

```
    line(x2, y2, x3, y3);
```

```
    line(x3, y3, x1, y1);} }
```

```
void Translate(int &x1, int &y1, int &x2, int &y2, int &x3, int &y3, int tx, int ty)
```

```
{    x1 += tx;
```

```
    y1 += ty;
```

```
    x2 += tx;
```

```
    y2 += ty;
```

```
    x3 += tx;
```

```
    y3 += ty;} }
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
void Scale(int &x1, int &y1, int &x2, int &y2, int &x3, int &y3, float sx, float sy)
```

```
{    x1 = int(x1 * sx);
    y1 = int(y1 * sy);
    x2 = int(x2 * sx);
    y2 = int(y2 * sy);
    x3 = int(x3 * sx);
    y3 = int(y3 * sy);}
```

```
void Rotate(int &x1, int &y1, int &x2, int &y2, int &x3, int &y3, float angle)
```

```
{    float radian = angle * 3.14159 / 180;
    int x1_new = int(x1 * cos(radian) - y1 * sin(radian));
    int y1_new = int(x1 * sin(radian) + y1 * cos(radian));
    int x2_new = int(x2 * cos(radian) - y2 * sin(radian));
    int y2_new = int(x2 * sin(radian) + y2 * cos(radian));
    int x3_new = int(x3 * cos(radian) - y3 * sin(radian));
    int y3_new = int(x3 * sin(radian) + y3 * cos(radian));
```

```
    x1 = x1_new;
    y1 = y1_new;
    x2 = x2_new;
    y2 = y2_new;
    x3 = x3_new;
    y3 = y3_new;}
```

```
int main()
```

```
{    int gd = DETECT, gm;
    initgraph(&gd, &gm, "C:\\\\Turboc3\\\\BGI");
```

```
    int x1 = 100, y1 = 100, x2 = 200, y2 = 100, x3 = 150, y3 = 50;
    Draw(x1, y1, x2, y2, x3, y3, WHITE);
    getch();
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
int c;

do
{ cout << "\n*** Choose Operation ***: \n1. Translate\n2. Scale\n3.
Rotate\n4. EXIT\nEnter choice: ";

    cin >> c;

switch (c){
case 1:{    int tx, ty;

            cout << "Enter translation (Tx Ty): ";

            cin >> tx >> ty;

            Translate(x1, y1, x2, y2, x3, y3, tx, ty);

            Draw(x1, y1, x2, y2, x3, y3, GREEN);

            break;}

case 2:{    float sx, sy;

            cout << "Enter Scaling Factor (Sx Sy): ";

            cin >> sx >> sy;

            Scale(x1, y1, x2, y2, x3, y3, sx, sy);

            Draw(x1, y1, x2, y2, x3, y3, RED);

            break;}

case 3:{    float angle;

            cout << "Enter Rotation Angle (in degrees): ";

            cin >> angle;

            Rotate(x1, y1, x2, y2, x3, y3, angle);

            Draw(x1, y1, x2, y2, x3, y3, YELLOW);

            break;}

case 4:{    cout << "Exiting program." << endl;

            break;}

default:{    cout << "Invalid choice!" << endl;

            break;}

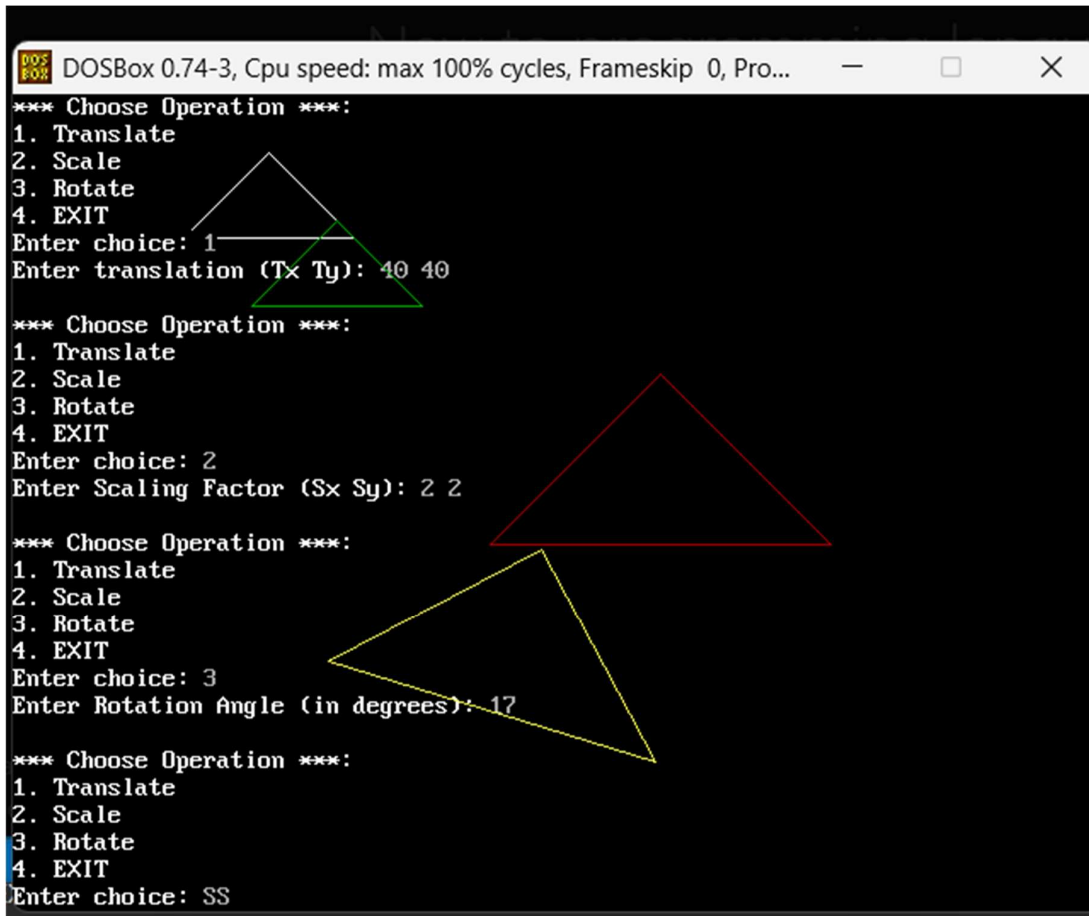
}} while (c != 4);

getch();

closegraph();

return 0;}
```

4. Output:



```
DOSBox 0.74-3, Cpu speed: max 100% cycles, Frameskip 0, Pro...
*** Choose Operation ***:
1. Translate
2. Scale
3. Rotate
4. EXIT
Enter choice: 1
Enter translation (Tx Ty): 40 40
*** Choose Operation ***:
1. Translate
2. Scale
3. Rotate
4. EXIT
Enter choice: 2
Enter Scaling Factor (Sx Sy): 2 2
*** Choose Operation ***:
1. Translate
2. Scale
3. Rotate
4. EXIT
Enter choice: 3
Enter Rotation Angle (in degrees): 17
*** Choose Operation ***:
1. Translate
2. Scale
3. Rotate
4. EXIT
Enter choice: SS
```

5. Learning Outcome:

- Got experience in defining and using functions for specific tasks (e.g., 'Draw', 'Translate', 'Scale', 'Rotate').
- Understood how to apply these transformations to points in a 2D space.
- Applied trigonometric functions (sine and cosine) to perform rotation transformations.
- Applied 'switch' statements for control flow based on user choices.
- Applied loops (e.g., 'do-while') for repeated operations until a certain condition is met.