# Group B

# Assignment No: 7(a)

**Title:** 2D Transformations

**Problem Statement:** Write C++ program to draw 2-D object and perform following basic transformations,

1. Scaling
2. Translation
3. Rotation

Apply concept of operator overloading

# PREREQUISITES:

1. Knowledge of matrix fundamentals and basic transformations on polygon - translation, rotation & scaling.
2. Basic mathematics, vectors & matrices

# COURSE OBJECTIVE:

1. To acquaint the learner with the basic concepts of Computer Graphics
2. To get familiar with mathematics behind the graphical transformations

**COURSE OUTCOME:** Develop scientific and strategic approach to solve complex problems in the domain of Computer Graphics

# THEORY:

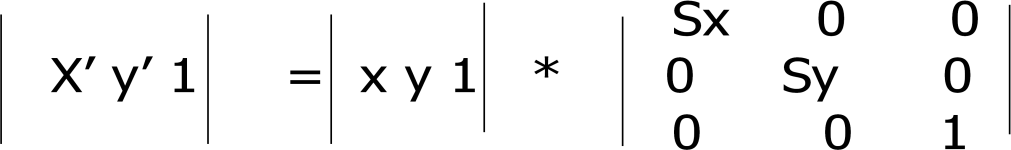
**2D tansformations:**

Transformations allow us to uniformly alter the entire picture. The geometric transformations considered here - translation, scaling and rotation are expressed in terms of matrix multiplication.

# Scaling

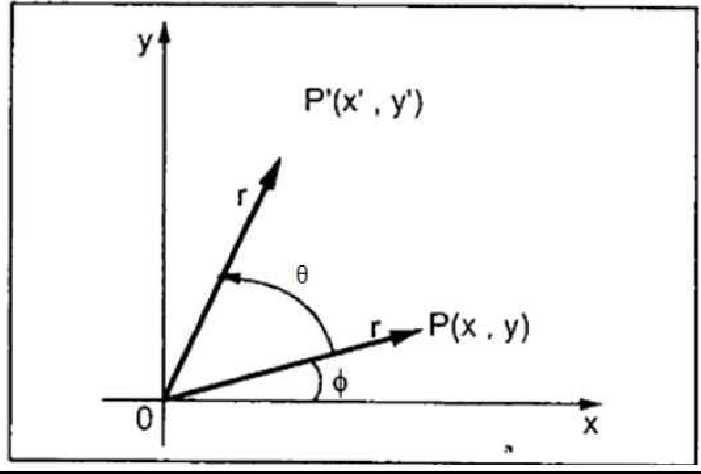
scaling refers to changing the size of the object either by increasing or decreasing.

Scaling can be achieved by multiplying the original coordinates of the object with the scaling factor to get the desired result.The scaling factor SX, SY scales the object in X and Y direction respectively. The above equations can also be represented in matrix form as below −



# Rotation

In rotation, we rotate the object at particular angle θ (theta) from its origin. From the following figure, we can see that the point P(X, Y) is located at angle φ from the horizontal X coordinate with distance r from the origin.



Using standard trigonometric the original coordinate of point P(X, Y) can be represented as − X = r cosϕ (1)

Y = r sinϕ (2)

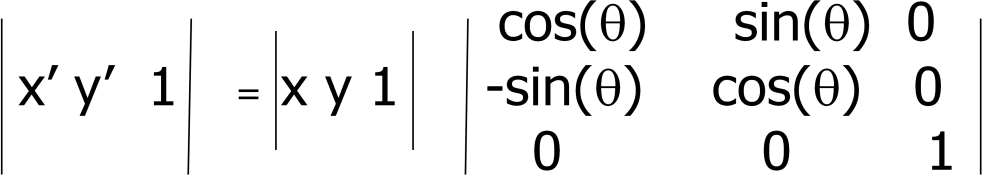
Same way we can represent the point P’ (X’, Y’) as − x′ = r cos(ϕ+θ) = r cosϕ cosθ − r sinϕ sinθ (3)

y′ = r sin(ϕ+θ) = r cosϕ sinθ + r sinϕ cosθ. (4)

Substituting equation (1) & (2) in (3) & (4) respectively, we will get x′ = x cosθ –y sinθ

y′ = x sinθ + y cosθ

Representing the above equation in matrix form,



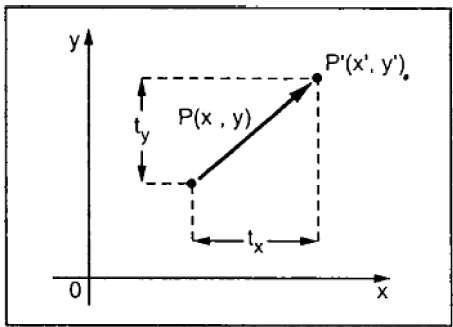
The rotation angle can be positive and negative.

For positive rotation angle, we can use the above rotation matrix. However, for negative angle rotation, the matrix will change

# Translation

A translation moves an object to a different position on the screen. You can translate a point in 2D by adding translation coordinate (tx, ty) to the original coordinate (X, Y) to get the new coordinate (X’, Y’).

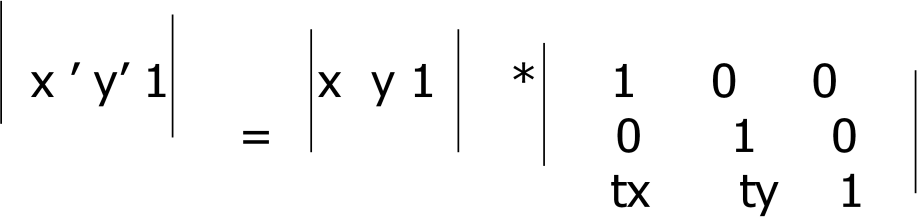
Translation Distance: It is nothing but by how much units we should shift the object from one location to another along x, y-axis.



From the above figure, you can write that − X’ = X + tx

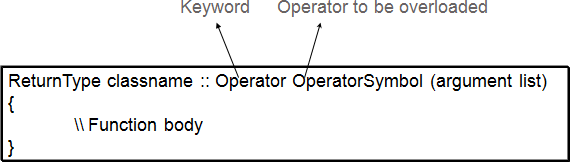
Y’ = Y + ty

The pair (tx, ty) is called the translation vector or shift vector. The translation represented in matrix form is



# Operator Overloading

Operator overloading is an important concept in C++. It is a type of polymorphism in which an operator is overloaded to give user defined meaning to it. Overloaded operator is used to perform operation on user-defined data type. For example '+' operator can be overloaded to perform addition on various data types, like for Integer, String(concatenation) etc.



# Algorithm:

1. Read input object in the form of matrix
2. Multiply input matrix with transformation matrix
3. Repeat same for all transformations
4. Display transformed object
5. stop

# OOP’s concepts used

1. **Class**

# Operator Overloading CONCLUSION: