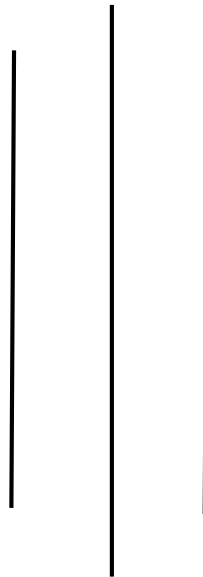


# SAGARMATHA ENGINEERING COLLEGE

(TU Affiliated)

Sanepa, Lalitpur



LAB NO: 3

A LAB REPORT ON

MID POINT CIRCLE ALGORITHM

**Submitted By:**

Name:.....

Faculty/Year:.....

Roll No:.....

Date:.....

**Submitted To:**

Department of electronics and Computer Engineering

Signature:.....

Date:.....



## COMPUTER GRAPHICS LAB-03

### TITLE

### MID POINT CIRCLE ALGORITHM

### OBJECTIVES

- ✓ To be familiar with fundamental knowledge of Mid-point circle drawing algorithm and its implementation.

### HARDWARE/SOFTWARE REQUIRED

- ✓ C Compiler

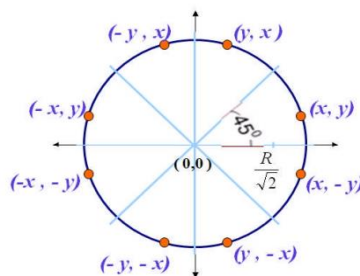
### RELATED THEROY

#### Mid-point Circle Algorithm

A circle is defined as the set of points that are all at a given distance  $r$  from a center position  $(x_c, y_c)$ . The equation of circle is:

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

In the mid-point circle algorithm we use eight-way symmetry so only ever calculate the points for the top right eighth of a circle, and then use symmetry to get the rest of the points.



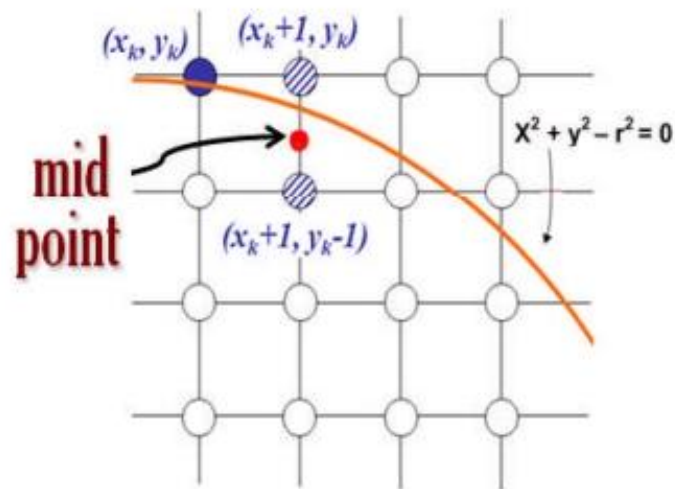
$$f_{\text{circle}}(x, y) \begin{cases} < 0, & \text{if } (x, y) \text{ is inside the circle boundary} \\ = 0, & \text{if } (x, y) \text{ is on the circle boundary} \\ > 0, & \text{if } (x, y) \text{ is outside the circle boundary} \end{cases}$$

**By evaluating this function at the midpoint between the candidate pixels we can make our decision**



ALGORITHM

COMPUTER GRAPHICS LAB-03



1. Input radius  $r$  and circle center  $(x_c, y_c)$  and obtain the first point on the circumference of a circle centered on the origin as

$$(x_0, y_0) = (0, r)$$

2. Calculate the initial value of the decision parameter as

$$P_0 = 5/4 - r$$

3. At each  $x_k$  position, starting at  $k = 0$ , perform the following test:

If  $p_k < 0$  /\* next point  $(x_{k+1}, y_k)$  \*/

$$x_{k+1} = x_k + 1$$

$$y_{k+1} = y_k$$

$$P_{k+1} = p_k + 2x_{k+1} + 1$$

else

/\*next point  $(x_k+1, y_k-1)$  \*/

$$x_{k+1} = x_k + 1$$

$$y_{k+1} = y_k - 1$$

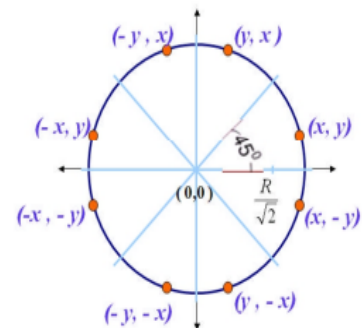
$$P_{k+1} = p_k + 2x_{k+1} + 1 - 2y_{k+1}$$

$$\text{Where } 2x_{k+1} = 2x_k + 2 \text{ and } 2y_{k+1} = 2y_k - 2$$

4. Determine the symmetry points in the other seven octants.
5. Move each calculated pixel position  $(x, y)$  onto the circular path centered on  $(x_c, y_c)$  and plot the co-ordinate values:

$$x = x + x_c, y = y + y_c$$

6. Repeat steps 3 through 5 until  $x \geq y$





## COMPUTER GRAPHICS LAB-03

### IMPLEMENTATION OF MID-POINT CIRCLE ALGORITHM



## COMPUTER GRAPHICS LAB-03

### OUTPUT

### RESULT:

Program is compiled, Mid-point circle algorithm implementation was done.