BTME 3060
Computer Aided Design
Lecture 9

2nd Year

III Semester

Galgotias University

2020-21

Unit I: Introduction to CAD

- Syllabus
  - Product Development Cycle
  - Introduction to CAD, Hardware and software requirement of CAD;
  - Graphics input devices- cursor control devices, Digitizers, Scanners, speech oriented devices and touch panels,
  - Graphics display devices- Refresh cathode ray tubes, Raster-scan displays, Random-scan displays, CRT Monitors;
  - Input devices- keyboard, joy-stick, mouse, scanner;
  - DVST, Flat- panel display, Hard copy devices - Printers and Plotters, dot matrix, inkjet, laser printers,
  - Graphics Standards – Neutral File formats –IGES, STEP,
  - Graphics software, Graphics functions,
  - **Output primitives- Bresenham's Algorithm and DDA**.

Objective of the lecture

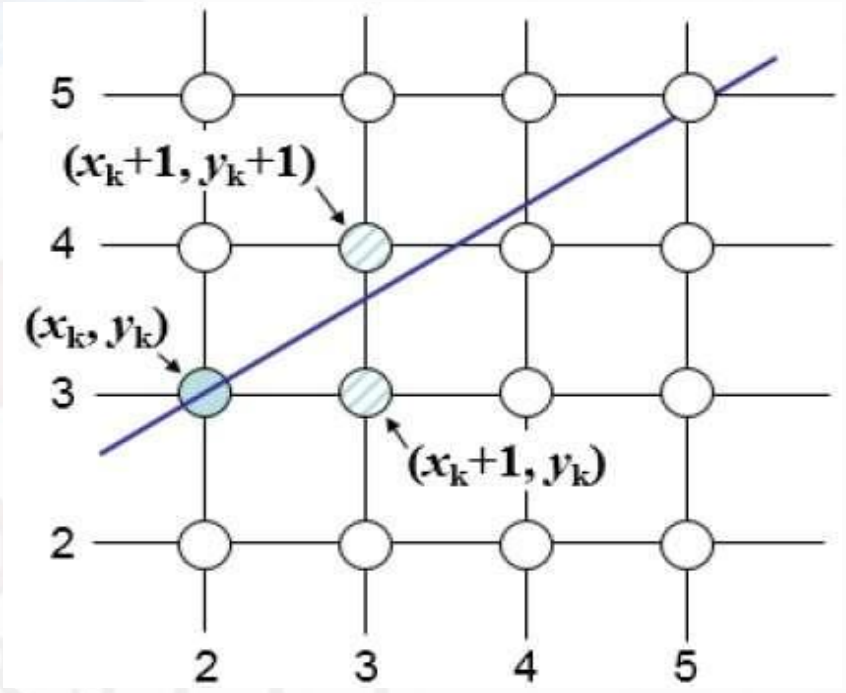
- To learn the working of Output primitives- Bresenham's Algorithm and DDA.
- To solve the numerical problems based on the above algorithms

Bresenham's Algorithm

- Bresenham's Line Generation
- The Bresenham algorithm is another incremental scan conversion algorithm.
- The big advantage of this algorithm is that, it uses only integer calculations.
- Moving across the x axis in unit intervals and at each step choose between two different y coordinates.
- For example, as shown in the following illustration, from position (2, 3) you need to choose between (3, 3) and (3, 4). You would like the point that is closer to the original line.

Contd..

- For example, as shown in the following illustration, from position (2, 3) you need to choose between (3, 3) and (3, 4).
- You would like the point that is closer to the original line.
- position Xk+1,Xk+1, the vertical separations from the mathematical line are labelled as dupperdupper and dlowerdlower.

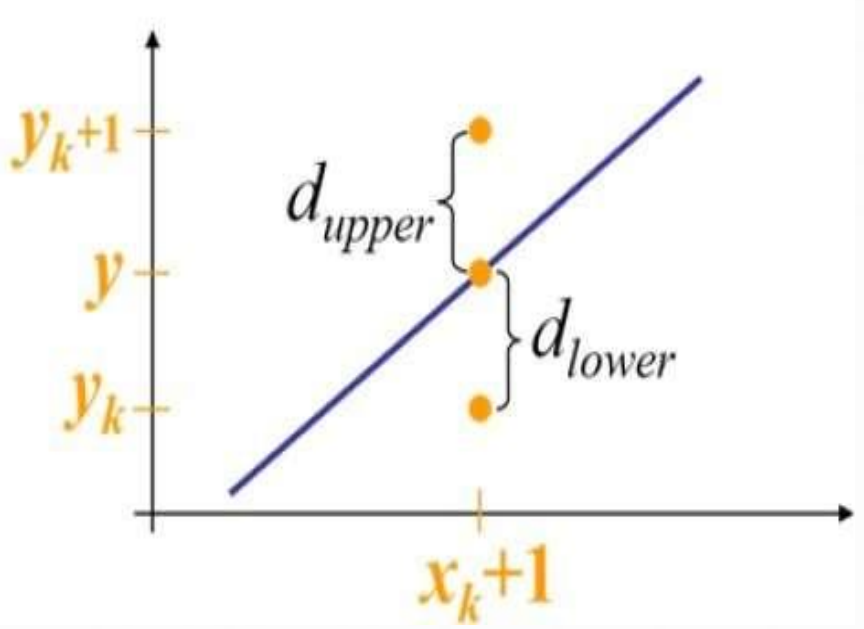


Contd..

- Y = m(XkXk+1) + b
- So, dupperdupper and dlowerdlower are given as follows −
- dlower=y−ykdlower=y−yk
- =m(Xk+1)+b−Yk=m(Xk+1)+b−Yk

and

- dupper=(yk+1)−ydupper=(yk+1)−y
- =Yk+1−m(Xk+1)−b=Yk+1−m(Xk+1)−b



Contd..

- You can use these to make a simple decision about which pixel is closer to the mathematical line. This simple decision is based on the difference between the two pixel positions.
- dlower−dupper=2m(xk+1)−2yk+2b−1dlower−dupper=2m(xk+1)−2yk+2b−1
- Let us substitute m with dy/dx where dx and dy are the differences between the end-points.
- dx(dlower−dupper)=dx(2dydx(xk+1)−2yk+2b−1)dx(dlower−dupper)=dx(2dy dx(xk+1)−2yk+2b−1)
- =2dy.xk−2dx.yk+2dy+2dx(2b−1)=2dy.xk−2dx.yk+2dy+2dx(2b−1)
- =2dy.xk−2dx.yk+C=2dy.xk−2dx.yk+C

Contd..

- So, a decision parameter PkPk for the kth step along a line is given by −
- pk=dx(dlower−dupper)pk=dx(dlower−dupper)
- =2dy.xk−2dx.yk+C=2dy.xk−2dx.yk+C
- The sign of the decision parameter PkPk is the same as that of dlower−dupperdlower−dupper.
- If pkpk is negative, then choose the lower pixel, otherwise choose the upper pixel.
- Remember, the coordinate changes occur along the x axis in unit steps, so you can do everything with integer calculations. At step k+1, the decision parameter is given as −
- pk+1=2dy.xk+1−2dx.yk+1+Cpk+1=2dy.xk+1−2dx.yk+1+C

Contd..

- Subtracting pkpk from this we get –
- pk+1−pk=2dy(xk+1−xk)−2dx(yk+1−yk)pk+1−pk=2dy(xk+1−xk)−2dx(yk+1−yk)
- But, xk+1xk+1 is the same as (xk)+1(xk)+1. So –
- pk+1=pk+2dy−2dx(yk+1−yk)pk+1=pk+2dy−2dx(yk+1−yk)
- Where, Yk+1–YkYk+1–Yk is either 0 or 1 depending on the sign of PkPk.
- The first decision parameter p0p0 is evaluated at (x0,y0)(x0,y0) is given as –
- p0=2dy−dxp0=2dy−dx

Contd..

- Now, keeping in mind all the above points and calculations, here is the Bresenham algorithm for slope m < 1 −
- **Step 1** − Input the two end-points of line, storing the left end-point in (x0,y0)(x0,y0).
- **Step 2** − Plot the point (x0,y0)(x0,y0).
- **Step 3** − Calculate the constants dx, dy, 2dy, and (2dy – 2dx) and get the first value for the decision parameter as −
- p0=2dy−dxp0=2dy−dx
- **Step 4** − At each XkXk along the line, starting at k = 0, perform the following test −
- If pkpk < 0, the next point to plot is (xk+1,yk)(xk+1,yk) and
- pk+1=pk+2dypk+1=pk+2dy

Contd..

- Otherwise,
- (xk,yk+1)(xk,yk+1)
- pk+1=pk+2dy−2dxpk+1=pk+2dy−2dx
- Step 5 − Repeat step 4 (dx – 1) times.
- For m > 1, find out whether you need to increment x while incrementing y each time.
- After solving, the equation for decision parameter PkPk will be very similar, just the x and y in the equation gets interchanged.

DDA-Digital Differential Analyzer - Algorithm

- Digital Differential Analyzer (DDA) algorithm is the simple line generation algorithm which is explained step by step here.
- Step 1 – Get the input of two end points (X0,Y0)(X0,Y0) and (X1,Y1)(X1,Y1).
- Step 2 – Calculate the difference between two end points.
  - dx = X1 - X0
  - dy = Y1 - Y0

Contd..

- Step 3 – Based on the calculated difference in step-2, you need to identify the number of steps to put pixel. If dx > dy, then you need more steps in x coordinate; otherwise in y coordinate.

```
if (absolute(dx) > absolute(dy))
  Steps = absolute(dx);
else
  Steps = absolute(dy);
```

- Step 4 – Calculate the increment in x coordinate and y coordinate.

```
Xincrement = dx / (float) steps;
Yincrement = dy / (float) steps;
```

Contd..

- Step 5 – Put the pixel by successfully incrementing x and y coordinates accordingly and complete the drawing of the line.

```
for(int v=0; v < Steps; v++)
{
  x = x + Xincrement;
  y = y + Yincrement;
  putpixel(Round(x), Round(y));
}
```

Summery

- These line drawing techniques are useful to understand the line drawing mechanism for CAD
- These are the basis of many types of drawing in CAD

Questions

- Digitize the line with the end points (0,0) and (8,4) using Bresenham's line algorithm
- Digitize the line with the end points (2,3) and (10, 6) using DDA line algorithm.

Text Books

- Computer Graphics - D.A.Godse A.P.Godse
- Computer Graphics : Algorithms and Implementations - D. P. Agarwal

- Thank you