

Компютърна графика

Генериране на графично изображение (визуализация)

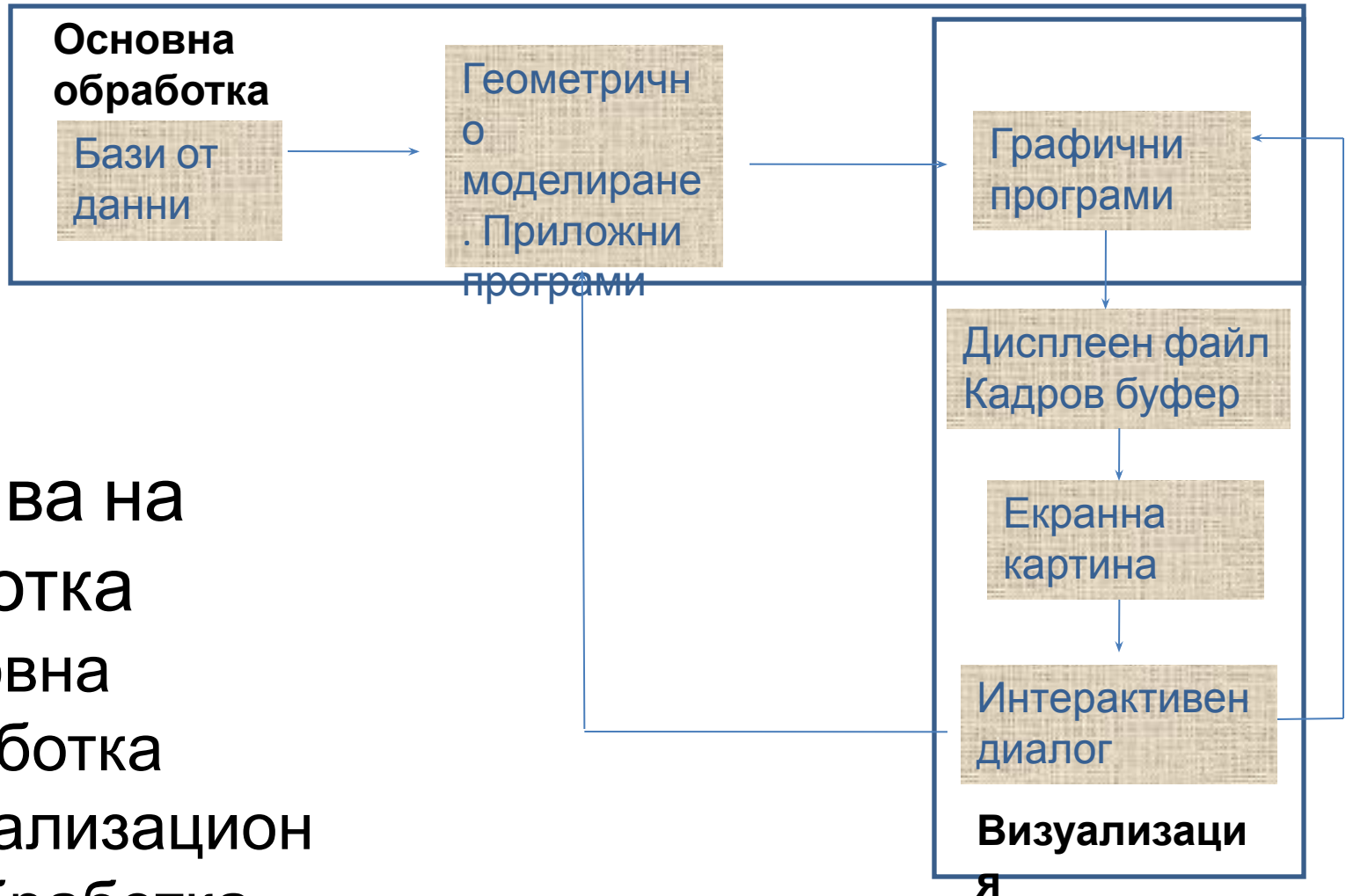


гл.ас. д-р инж. М. Иванова



Технически университет - София

Обработка на информацията в графичната система



- Две нива на обработка
 - Основна обработка
 - Визуализационна обработка

Състав на дисплейната картина

Графичните примитиви са основни графични елементи, които се използват за изграждане на дисплейния образ.

Обикновено в инженерната практика примитиви се наричат тези елементи, които се генерират апаратно в дисплейната станция или плотера.

В този смисъл типични примитиви са:

- точка,
- вектор (отсечка),
- буквено-цифров символ (текст).

Графични примитиви - стандартен набор

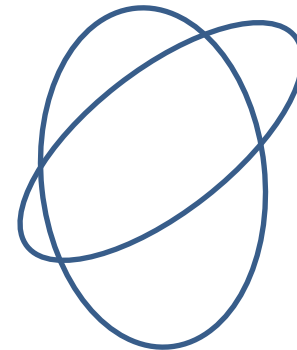
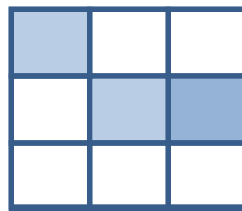
1. *Полилиния (Poliline)* - представлява последователно свързани отсечки и се определя от координатите на крайните точки на всяка отсечка.
2. *Полимаркер (Polimarker)* - представлява множество от графични символи (маркери) от един и същи тип, зададени чрез координатите на своите центрове.
3. *Текст (Text)* - последователност (низ) от буквено-цифрови символи.



ТЕКСТ

Графични примитиви - стандартен набор

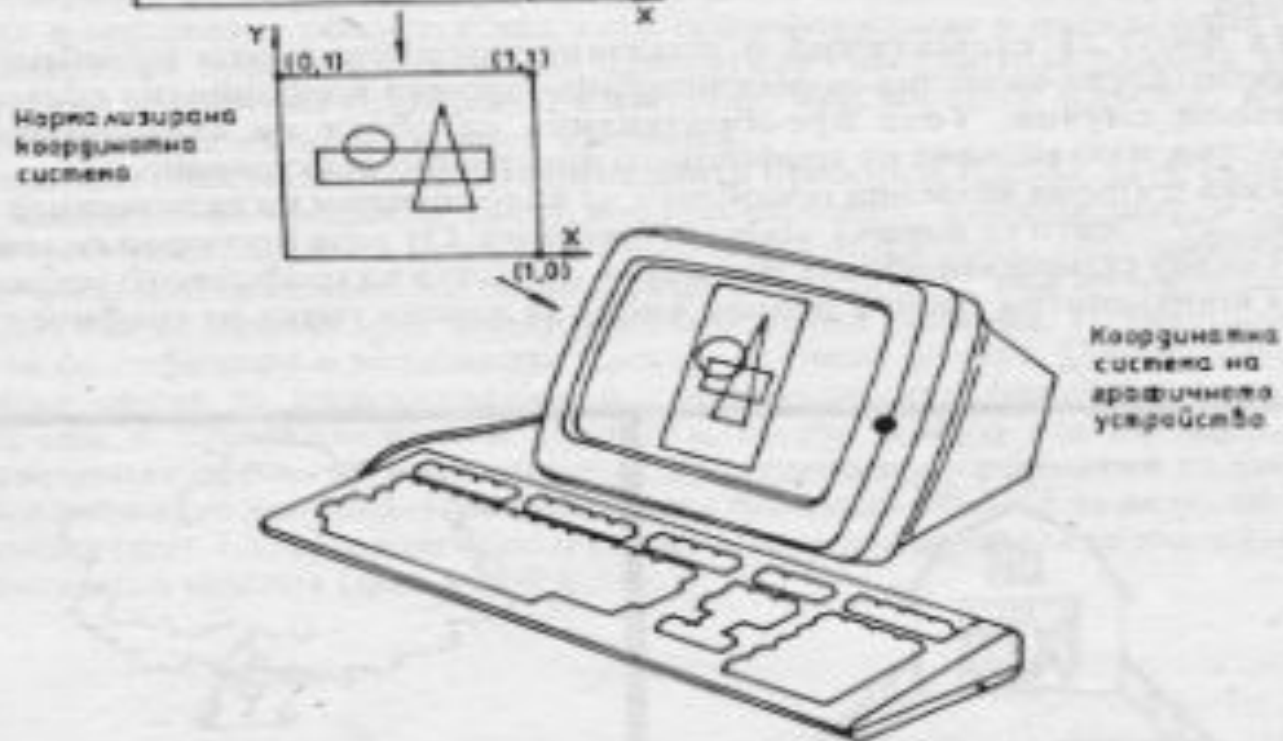
4. *Запълнена област (Fill Area)* - равнинен многоъгълник, който може да бъде празен, запълнен с даден цвят или с щриховка .
5. *Матрица от клетки (Cell Array)* - матрица от клетки, запълнени с различни цветове .
6. *Обобщен чертожен примитив (Generalized Drawing Primitive)* - по-сложни фигури, като окръжност, интерполирана крива, елипса и др.



Графични примитиви - параметри

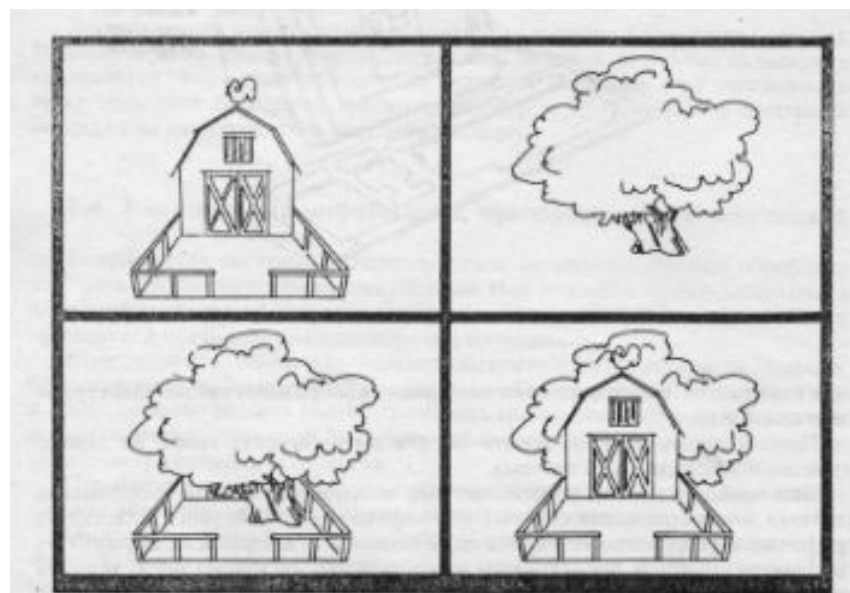
- Всеки примитив има три типа параметри:
 - Геометрични – управляват формата и размера на примитива
 - Негеометрични – указват начина на изобразяване: цвят, тип на линията
 - Идентификатори – за улавяне на примитива с интерактивно средство

Координатни системи



Моделен прозорец

- Всяка избрана моделна подобласт за визуализация на включените в тази подобласт обекти се нарича *моделен прозорец*. От този прозорец се генерира образ върху екранната област или върху областта на графичното устройство.

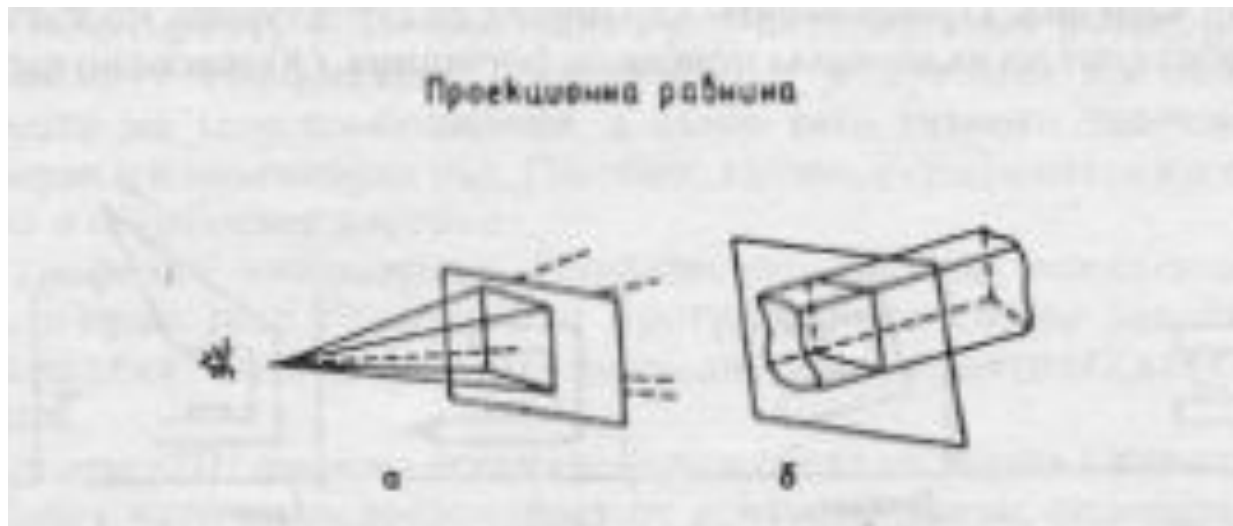


Функция прозорец, функция изрязване

- Преобразуването на моделния прозорец в екранен образ се нарича *функция прозорец*.
- Друга функция, директно свързана с функцията прозорец, е т.нар. *изрязване*, с което се определя кои части от зададения графичен обект попадат в екранната област.

Проекции

- При *централна проекция* обемът за визуализация е пирамида, а при *паралелна проекция* пирамидата се модифицира в проекционна призма.



Основни трансформации на картината

- Основни трансформации са:
 - Транслация – преместване на картинен сегмент
 - Ротация – завъртане около точка или ос
 - Мащабиране – свиване или разширяване на елемент

Производни трансформации

- а) *разтягане* или *свиване* по дадена ос - извършва се чрез трансляция на дадени контурни линии или точки от тях, другите линии остават непроменени;
- б) *огледало* (осева симетрия) - построява се огледален образ на картинния елемент;
- в) *разтягане* (като ластик) - в резултат на интерактивна намеса върх или отсечка от дадена фигура се премества в произволна посока, при което свързаните ръбове се разтягат като ластик; указаното разтягане по ос (т.а) може да се счита частен случай на метода на ластика;
- г) *зуминг* (*лупа*) - последователно мащабиране в прогресия с еднакви мащабни коефициенти по две оси, така че да се получи впечатление за отдалечаване или приближаване на картината или сегмент от нея (подобно е на вариооптичния ефект във фотографията);
- д) *превъртане* - динамично изобразяване на картината по елементи от картината около дадена ос, непрекъснато се сменя в пространството



Генериране на изображения на графични примитиви

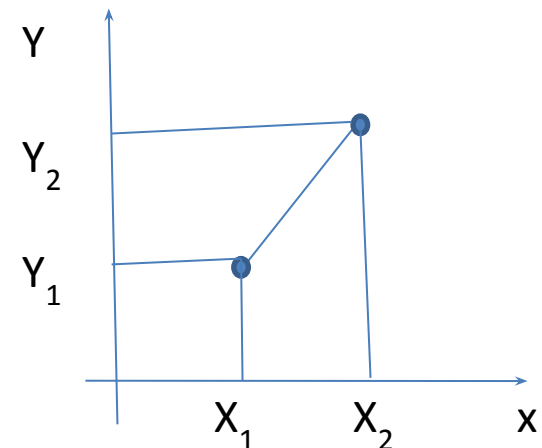
- **Полилиния**

- $Y = mx + b$ – растерни устройства

- $m = \frac{Y_2 - Y_1}{X_2 - X_1} = \frac{\Delta y}{\Delta x}$

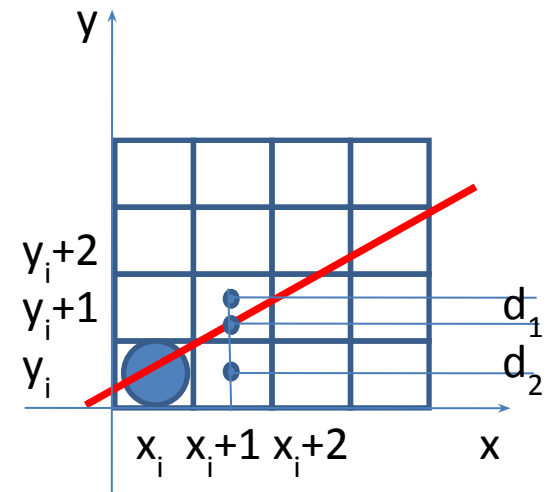
- $b = Y_1 - mX_1$

- $\Delta y = m \Delta x$ – векторни устройства



Генериране на изображения на графични примитиви

- Полилиния – алгоритъм на Брезенхайм
- $Y = m(x_i + 1) + b$
- $d_1 = Y - y_i = m(x_i + 1) + b - y_i$
- $d_2 = (y_i + 1) - Y = y_i + 1 - m(x_i + 1) - b$
- $d_1 - d_2 = 2m(x_i + 1) - 2y_i + 2b - 1$
- $p_1 = 2\Delta y - \Delta x$



Генериране на изображения на графични примитиви

- Стъпки в алгоритъма:

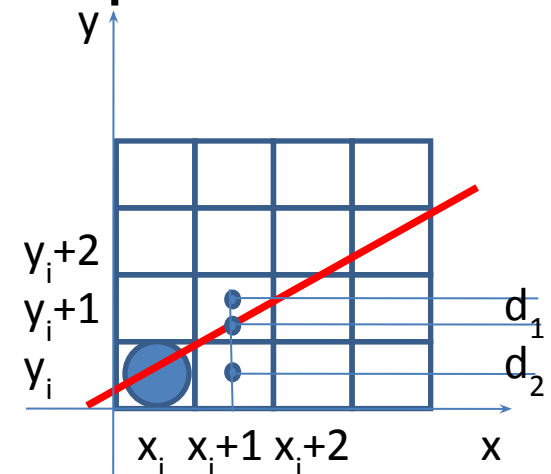
1. Запомнят се координатите на крайните точки на отсечката: (X_1, Y_1) и (X_2, Y_2)

2. Изчисляват се Δx , Δy , p_1 . Ако $p_1 < 0$ – следващия пиксел е: (X_1+1, Y_1) , в противен случай е: (X_1+1, Y_1+1)

3. Координата X нараства с 1.

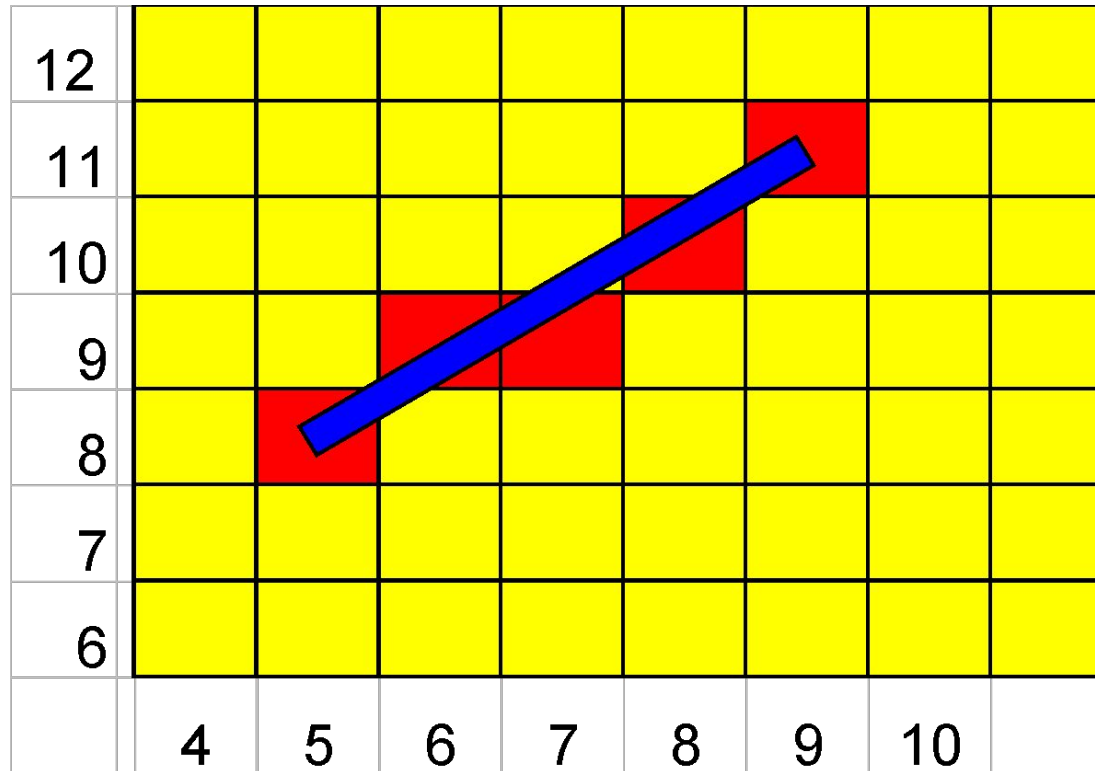
(x_i+1, y_i) ако $p_i < 0$

(x_i+1, y_i+1) ако $p_i > 0$

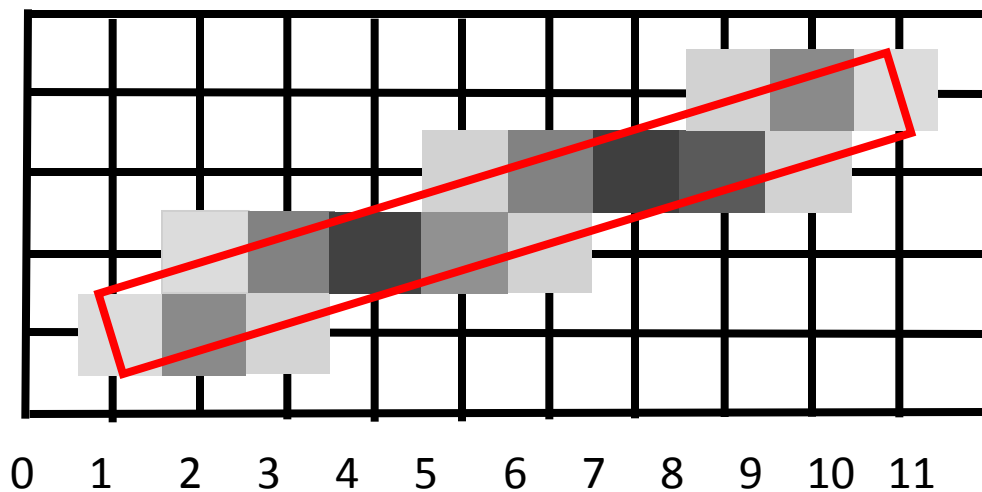


1. Повтаря се стъпка 3 до достигане $x_i = X_2$.

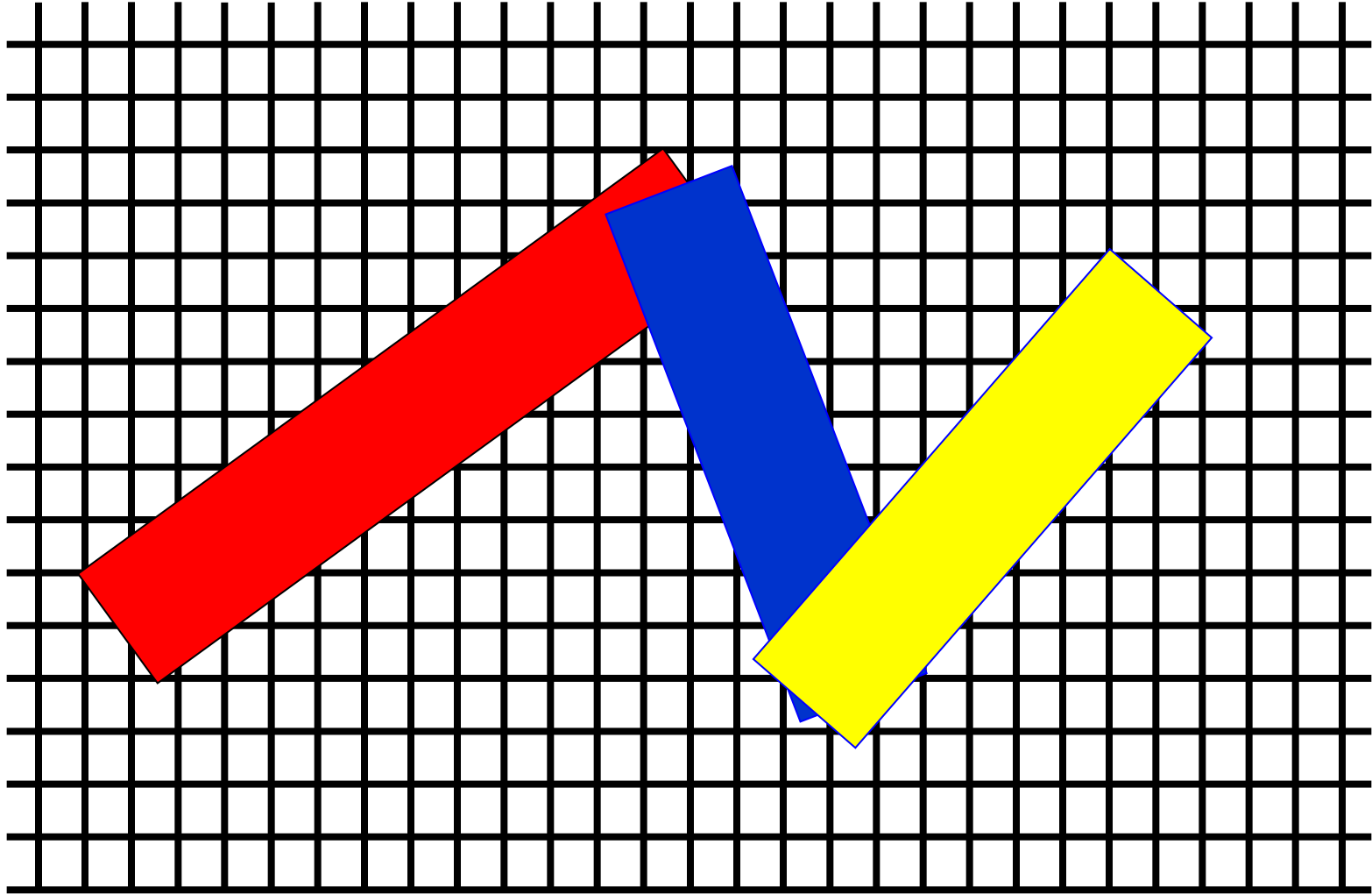
Генериране на изображения на графични примитиви



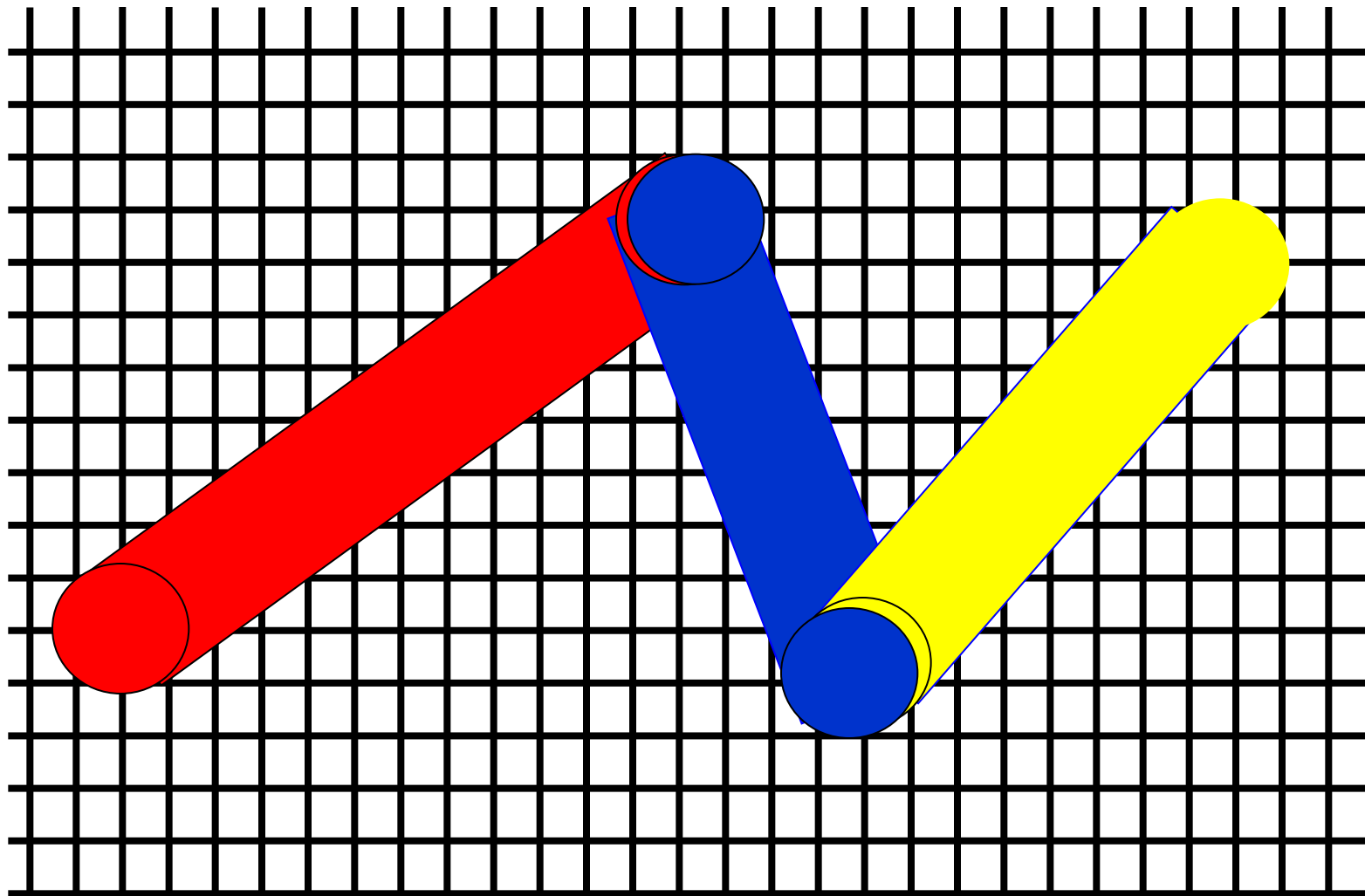
Генериране на изображения на графични примитиви - изглаждане



Генериране на изображения на графични примитиви

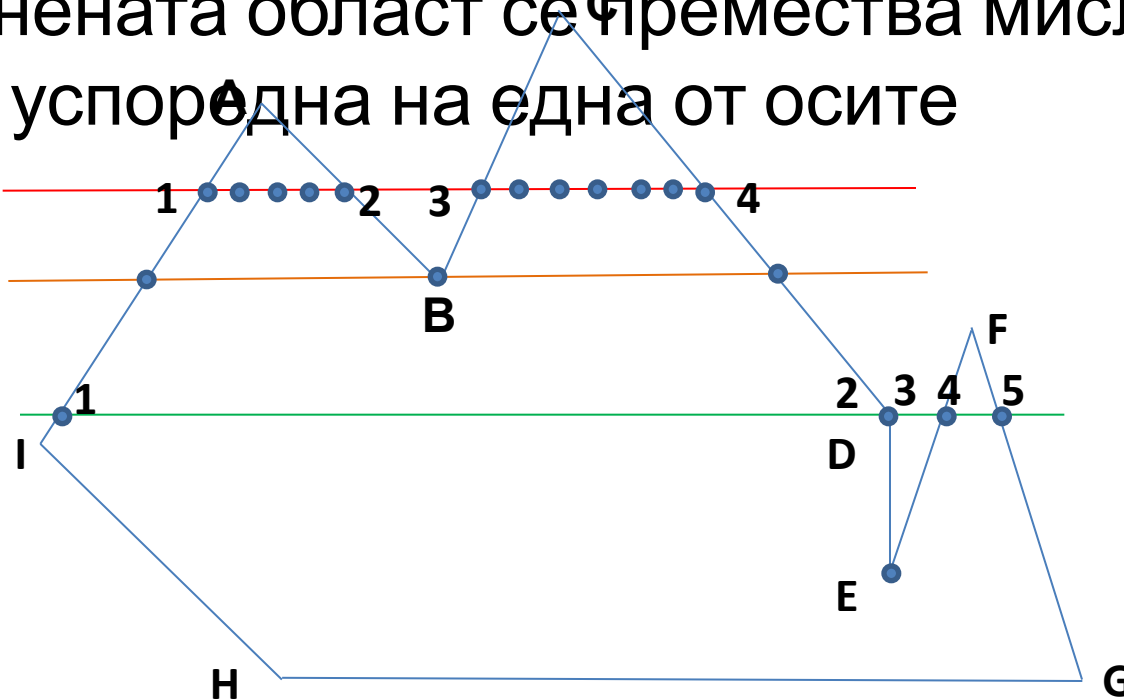


Генериране на изображения на графични примитиви



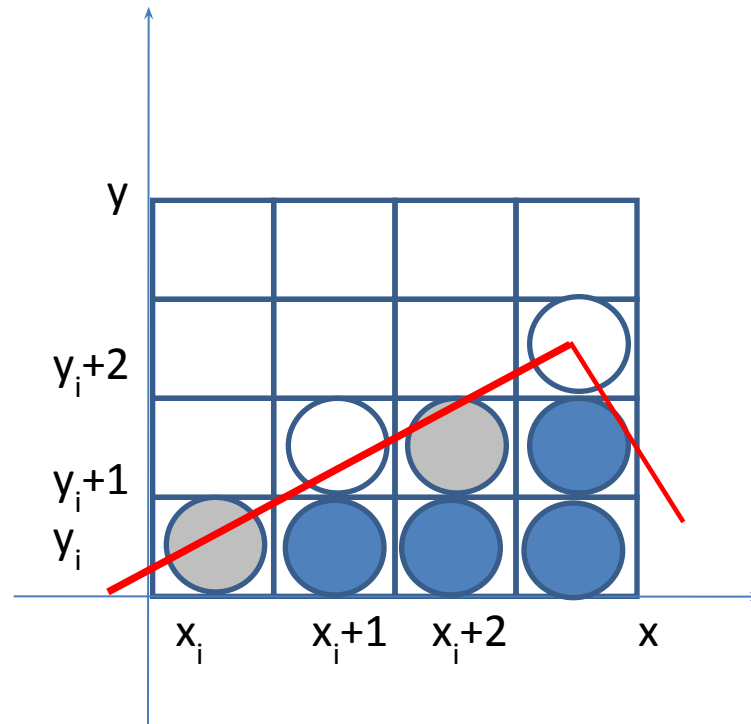
Генериране на изображения на графични примитиви

- **Запълнена област** – растерни графични устройства
- **Метод на сканиращата линия** – през запълнената област се премества мислена линия успоредна на една от осите



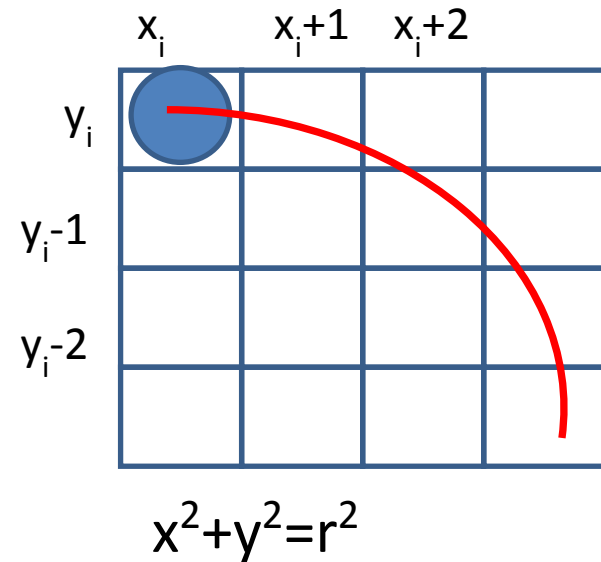
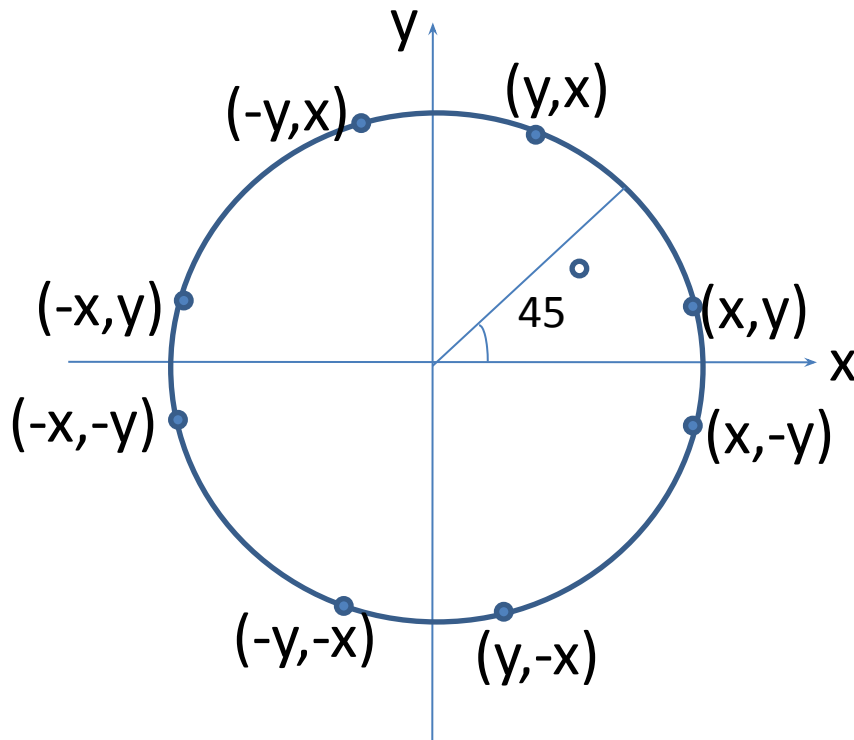
Генериране на изображения на графични примитиви

- Метод на сканиращата линия – изглаждащи процедури



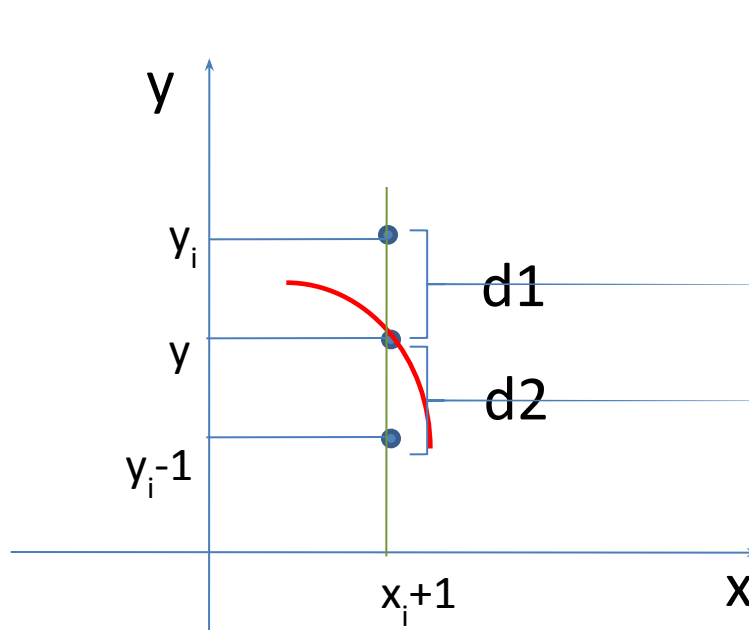
Генериране на изображения на графични примитиви

Окръжност – алгоритъм на Брезенхайм



Генериране на изображения на графични примитиви

- Окръжност – алгоритъм на Брезенхайм



$$y^2 = r^2 - (x_i + 1)^2$$

$$d_1 = y_i^2 - y^2 = y_i^2 - r^2 + (x_i + 1)^2$$

$$d_2 = y^2 - (y_i - 1)^2 = r^2 - (x_i + 1)^2 - (y_i - 1)^2$$

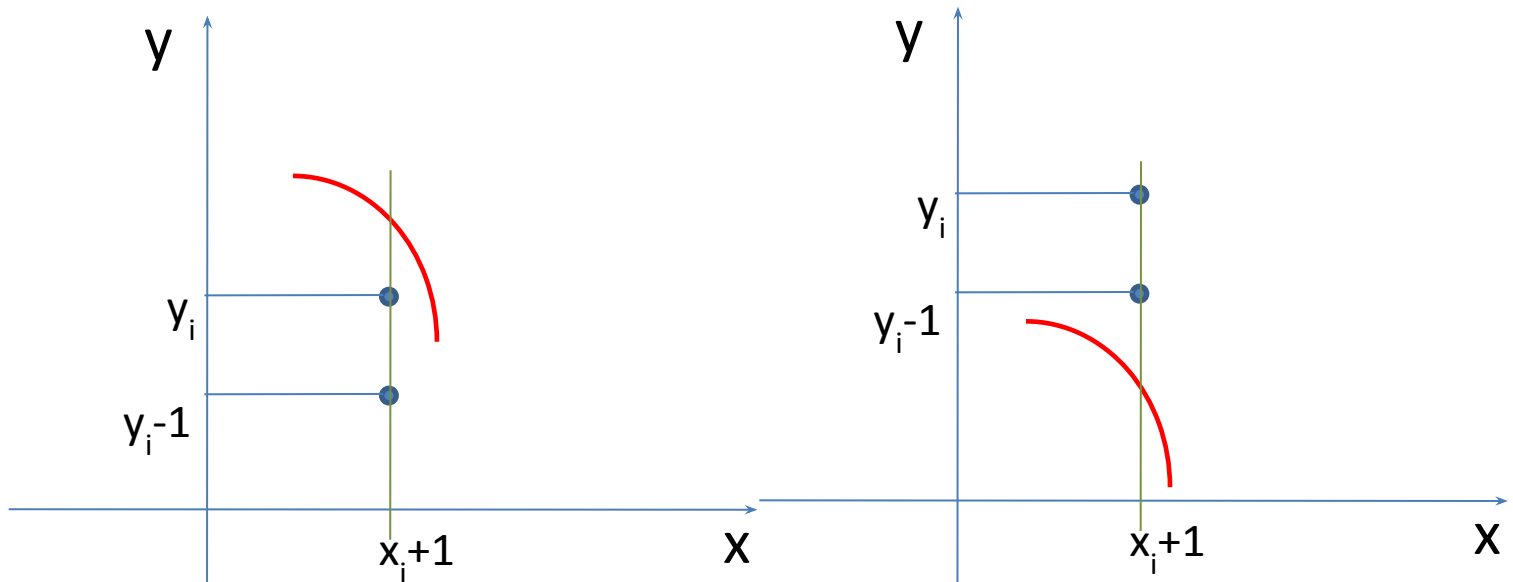
$$p_i = d_1 - d_2 = 2(x_i + 1)^2 + y_i^2 + (y_i - 1)^2 - 2r^2$$

Ако $p_i < 0$ - $(x_i + 1, y_i)$

Ако $p_i > 0$ - $(x_i + 1, y_i - 1)$

Генериране на изображения на графични примитиви

- Окръжност – алгоритъм на Брезенхайм

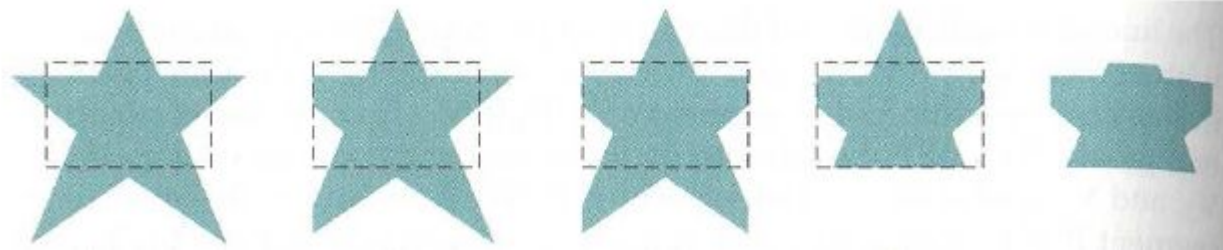
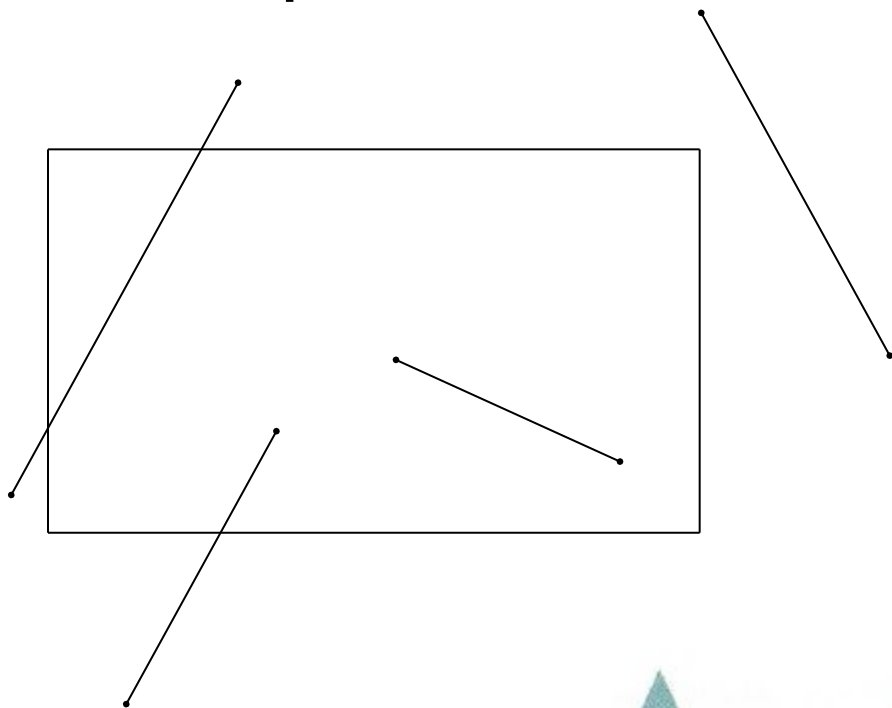


Ако $d_1 < 0$, $d_2 < 0$, $p_i < 0$ - (x_{i+1}, y_i)

Ако $d_1 > 0$, $d_2 < 0$, $p_i > 0$ - (x_{i+1}, y_i-1)

Определяне и скриване на невидими линии

- Алгоритъм на Коен-Съдърленд



Определяне и скриване на невидими линии

- Алгоритъм на Коен-Съдърленд

$K_x = -1$ $K_y = 1$	$K_x = 0$ $K_y = 1$	$K_x = 1$ $K_y = 1$	$K_x = -1$ – точка отляво на лявата граница на прозореца $K_x = 1$ – точка отдясно на дясната граница на прозореца
$K_x = -1$ $K_y = 0$	$K_x = 0$ $K_y = 0$ Екран	$K_x = 1$ $K_y = 0$	$K_x = 0$ – точка между двете вертикални граници на прозореца $K_y = -1$ – точка под долната граница на прозореца $K_y = 1$ – точка над горната граница на прозореца
$K_x = -1$ $K = -1$	$K_x = 0$ $K = -1$	$K_x = 1$ $K = -1$	$K_y = 0$ – точка между двете хоризонтални граници на прозореца