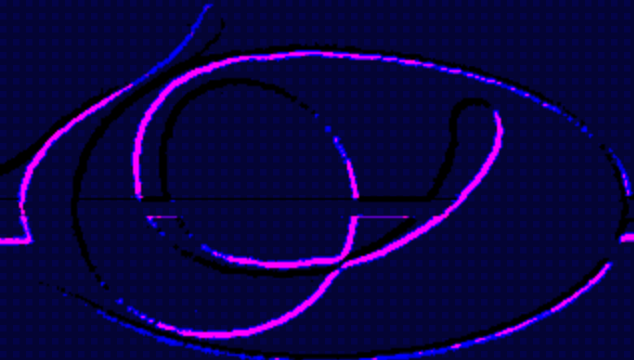




Grafika Komputerowa

wykład 5:

Podstawowe algorytmy rysowania
prymitywów 2D w grafice rastrowej



Prymitywy grafiki 2D

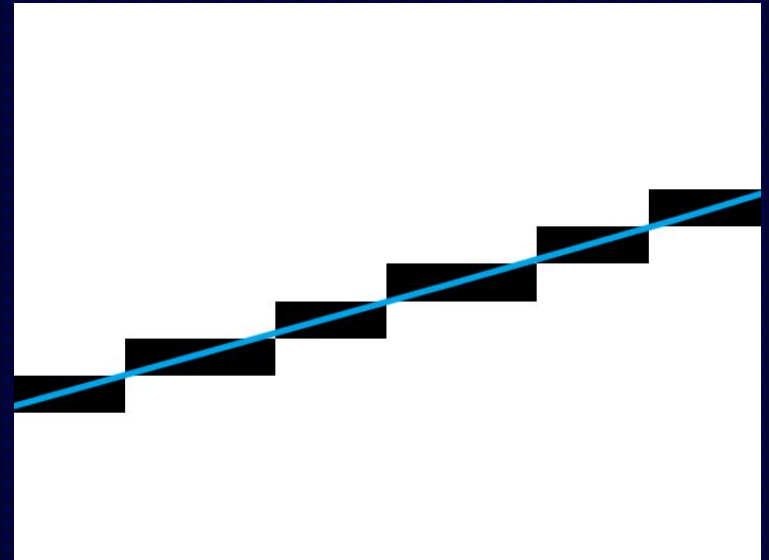
Geometria obrazu – opis za pomocą „prymitywów”, jak:

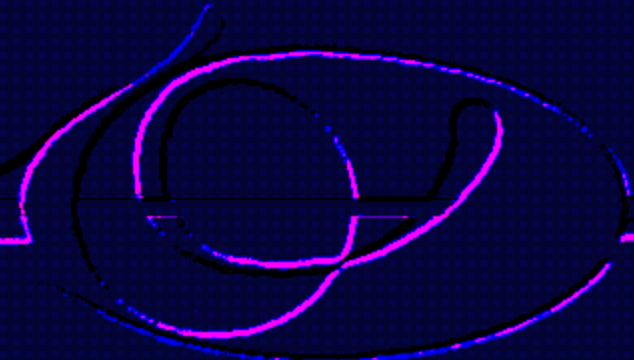
- odcinki,
- trójkąty, prostokąty, wielokąty,
- łuki, okręgi, elipsy,
- koła i ich wycinki.

Rysowanie „prymitywów” w grafice rastrowej – przejście od układu współrzędnych rzeczywistych do układu współrzędnych całkowitych. Konieczność aproksymacji równań do siatki pikseli.

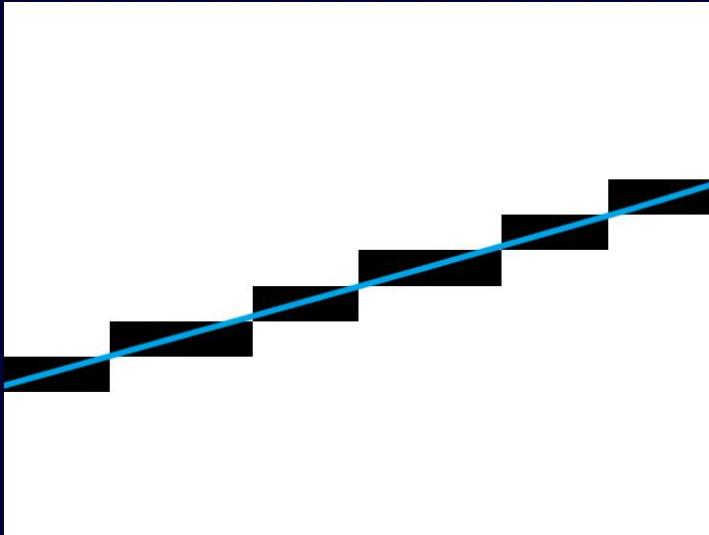
Konwersja odcinków

Algorytm konwersji odcinka oblicza współrzędne pikseli, których odległość od wektora wyznaczającego jego przebieg jest możliwie mała. W wyniku nałożenia idealnego odcinka na siatkę rastra i zastosowania konwersji otrzymujemy zbiór pikseli tworzących przybliżenie wektora.



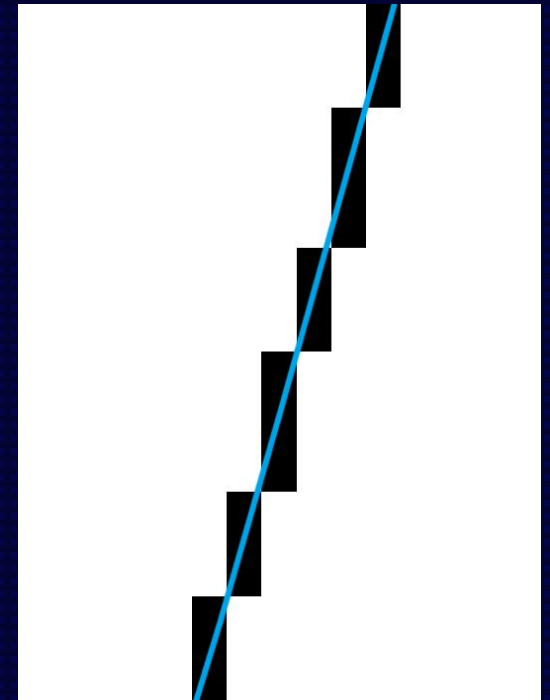


Konwersja odcinków



Dla odcinków o nachyleniu z przedziału $[-1,1]$ włącznie - pojedynczy piksel w każdej kolumnie.

Dla odcinków o nachyleniu spoza tego - pojedynczy piksel w każdym wierszu.



Konwersja odcinków

Algorytm DDA (ang. *Digital Differential Algorithm*) – **algorytm przyrostowy**

Działania wykonywane są na liczbach zmiennoprzecinkowych.

$$y = mx + b$$

$P_0(x_0, y_0)$ - początkowy punkt odcinka,

$P_k(x_k, y_k)$ - punkt końcowy

Współczynnik kierunkowy: $m = \frac{y_k - y_0}{x_k - x_0}$

Konwersja odcinków

Algorytm DDA – algorytm przyrostowy

gdy $|m| \leq 1$, to przyrost pikseli w kierunku osi OX jest stały i wynosi 1.
Oś OX nazywamy wówczas osią wiodącą.

Położenie kolejnych pikseli odcinka

$$x_{i+1} = x_i + 1$$

$$y_{i+1} = mx_{i+1} + b$$

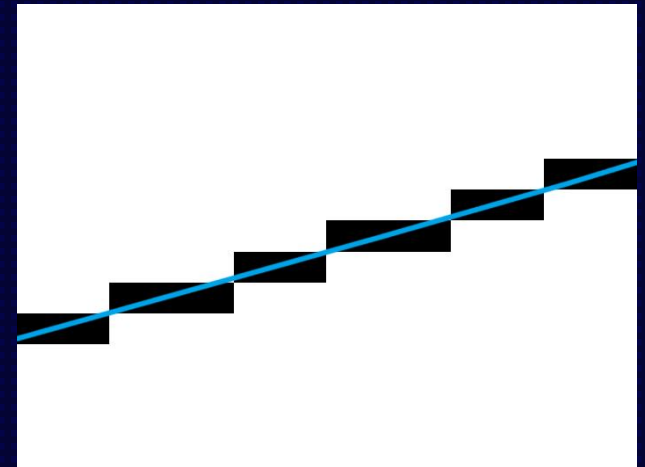
Po podstawieniu: $x_{i+1} = x_i + 1$

$$y_{i+1} = m(x_i + 1) + b$$

$$y_{i+1} = mx_i + b + m$$

Korzystając z równania kierunkowego prostej ($y_i = mx_i + b$):

$$y_{i+1} = y_i + m$$



Konwersja odcinków

Algorytm DDA – algorytm przyrostowy

gdy $|m| > 1$, to przyrost pikseli w kierunku osi OY jest stały i wynosi 1. Oś OY nazywamy wówczas osią wiodącą.

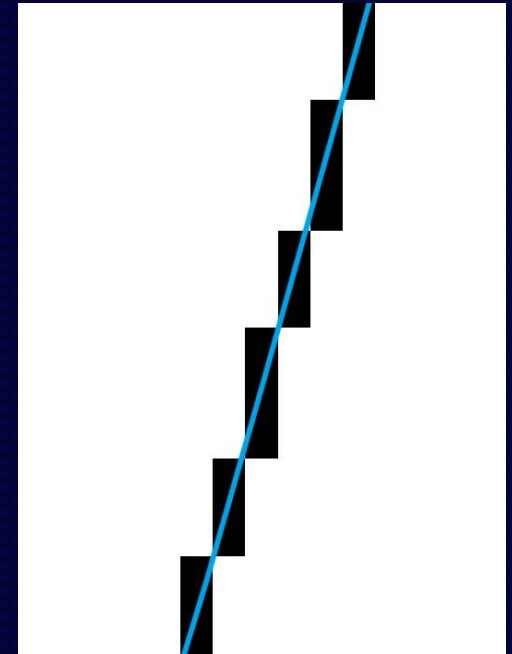
Korzystając z równania kierunkowego prostej:

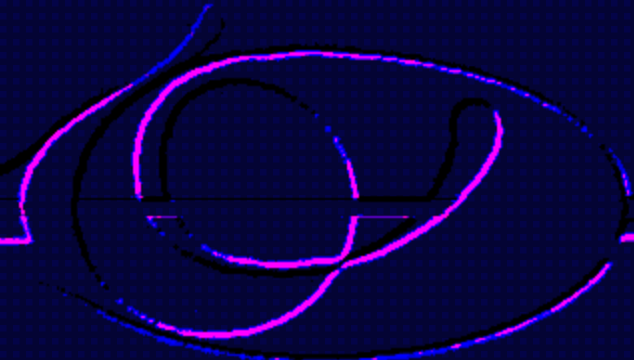
$$x = \frac{y}{m} - \frac{b}{m}$$

Po analogicznych przekształceniach:

$$y_{i+1} = y_i + 1$$

$$x_{i+1} = x_i - \frac{1}{m}$$





Konwersja odcinków

Algorytm DDA – algorytm przyrostowy

Otrzymane punkty położenia pikseli – liczby zmiennoprzecinkowe.

W obrazie rastrowym muszą zostać zaokrąglone do liczb całkowitych.

Wybierany jest piksel o odległości najmniejszej od „idealnego” odcinka.

Algorytm mało efektywny – obliczenia zmiennopozycyjne.

Konwersja odcinków

Algorytm Bresenhama - algorytm punktu środkowego

Najbardziej popularny algorytm rysowania odcinka.

Obliczenia na liczbach stałoprzecinkowych – zwiększenie szybkości działania.

Dla odcinka o kącie nachylenia względem osi 0X zawartym w przedziale $(0, 45]$, współczynnik kierunkowy

$$m = \frac{y_k - y_0}{x_k - x_0} = \frac{dy}{dx} \quad 0 < m \leq 1$$

Konwersja odcinków

Algorytm Bresenhama - algorytm punktu środkowego

Przyrost w kierunku osi OX (osi wiodącej) jest stały - 1 piksel.

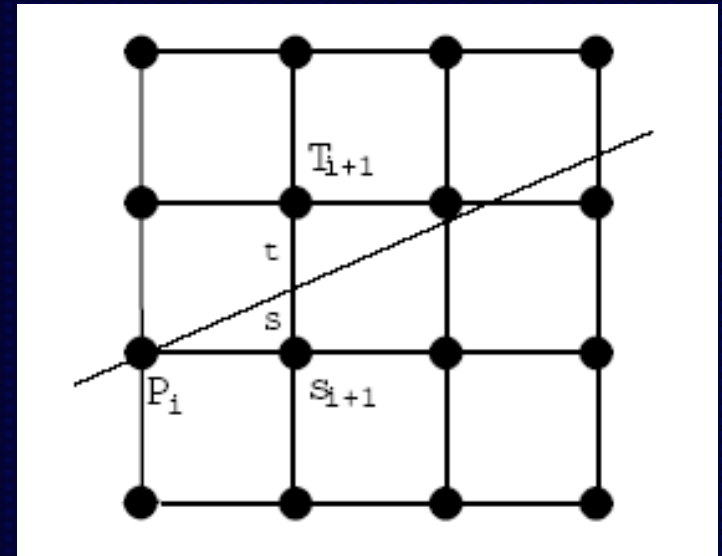
Dany piksel - $P_i(x_i, y_i)$

Następny:

$S_{i+1} = (x_{i+1}, y_i)$

lub

$T_{i+1}(x_{i+1}, y_{i+1})$.



Konwersja odcinków

Algorytm Bresenhama - algorytm punktu środkowego

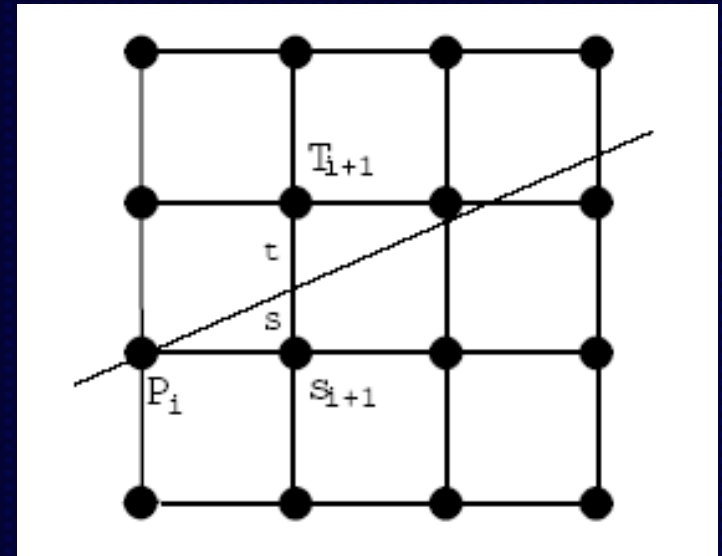
Korzystając z równania kierunkowego prostej:

$$y = \frac{dy}{dx}(x - x_0) + y_0$$

można obliczyć odległości pikseli od punktu przecięcia „idealnego” odcinka z prostą łączącą te 2 piksele

$$s = \frac{dy}{dx}(x_i + 1 - x_0) - (y_i - y_0)$$

$$t = (y_i + 1 - y_0) - \frac{dy}{dx}(x_i + 1 - x_0)$$



Konwersja odcinków

Algorytm Bresenhama - algorytm punktu środkowego

Równania odejmujemy stronami i mnożymy przez dx :

$$d_i = dx(s - t) = 2dy(x_i - x_0) - 2dx(y_i - y_0) + 2dy - dx$$

$dx > 0$,

zatem znak d_i wskazuje,
która z odległości s i t jest większa.

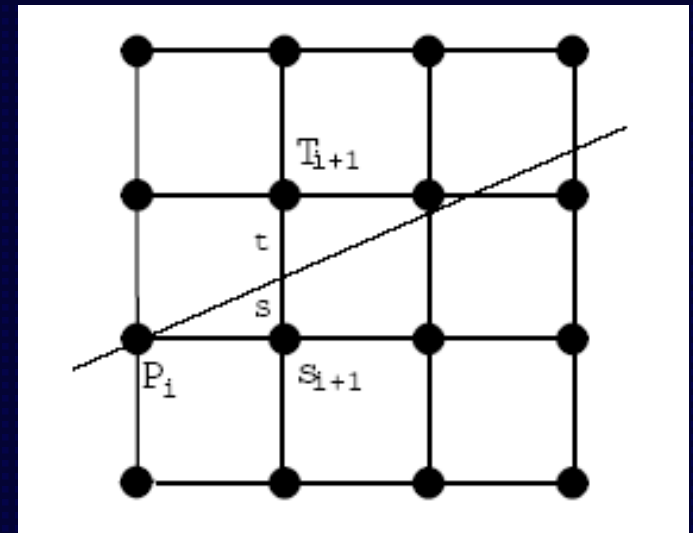
Dla $d_i > 0$, $\Rightarrow s > t$

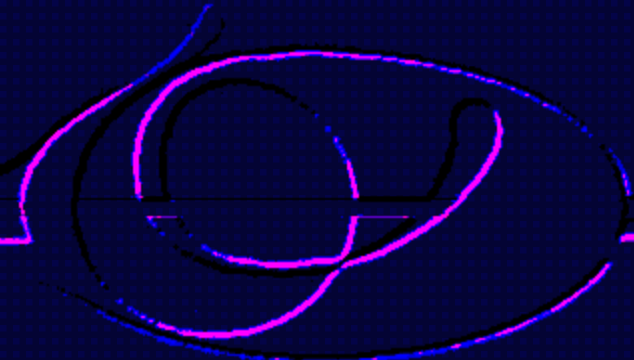
punkt P_{i+1} przyjmie położenie T_{i+1} ,

jeżeli $d_i < 0$,

punkt P_{i+1} przyjmie położenie S_{i+1}

Dla $d_i = 0$ zaokrąglamy wartość w górę.





Konwersja odcinków

Problemy dotyczące konwersji odcinków:

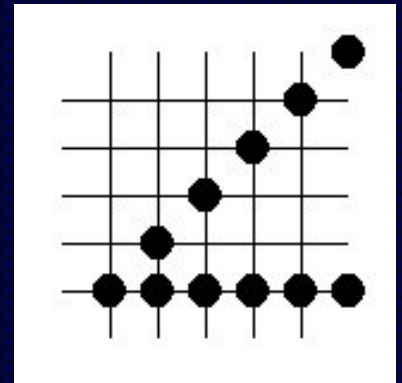
- W przypadku, gdy zmienna decyzyjna $d_i = 0$, punkt przecięcia „idealnego” odcinka z prostą łączącą piksele leży w połowie odległości między pikselami. Wybór piksela poprzez „zaokrąglenie” jest różny w zależności od kierunku kreślenia odcinka.

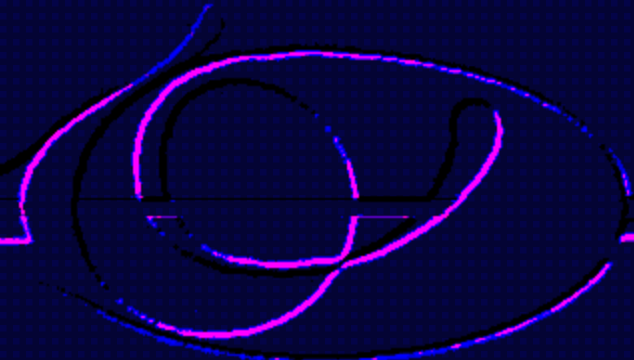
1000000

- Odcinki kreślone z zastosowaniem „stylu linii”
- zależność od kierunku rysowania.

- Zmiana jasności odcinka w funkcji nachylenia –
Różnica odległości pomiędzy pikselami. Nie jest
Możliwa do usunięcia w grafice 1-bitowej.

- Prostokąt obcinający a początek odcinka – różnice
szczególnie widoczne przy małym kącie nachylenia.





Konwersja okręgu i elipsy

Równanie okręgu o środku w początku układu współrzędnych (0, 0) i promieniu R:

$$x^2 + y^2 = R^2$$

Równanie elipsy o środku w początku układu współrzędnych (0, 0) i półosiach o długości a oraz b:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

Konwersja okręgu

Algorytm symetrii oktantowej

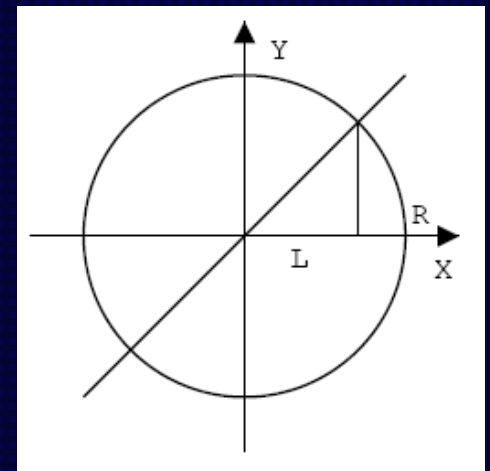
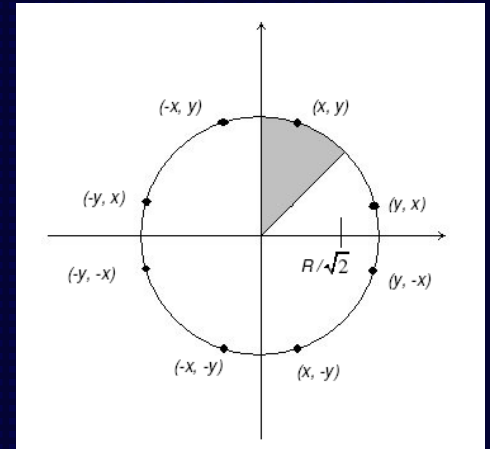
Podział okręgu na 8 symetrycznych części.

Ze wzoru na równanie okręgu można obliczyć y:

$$y = \pm\sqrt{R^2 - x^2}$$

A stosując wzory trygonometryczne L:

$$L = \frac{R}{\sqrt{2}} = R \frac{\sqrt{2}}{2}$$



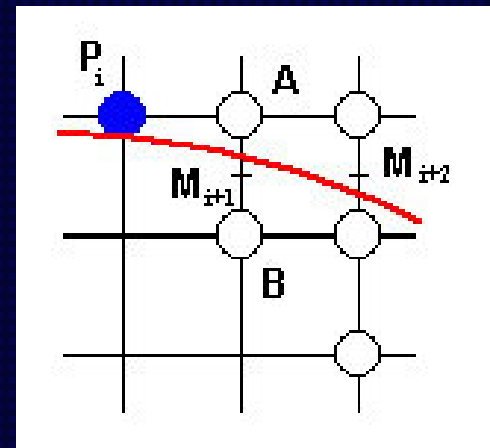
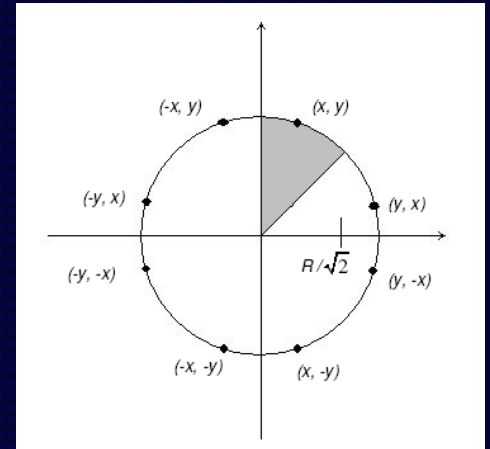
Konwersja okręgu

Algorytm z punktem środkowym

Obliczenia wykonywane w drugim oktancie,

$$\text{dla } x \in \left[0, \frac{R}{\sqrt{2}} \right]$$

Jako punkt następny po $P_i (x_i, y_i)$ może zostać wybrany piksel A albo B.

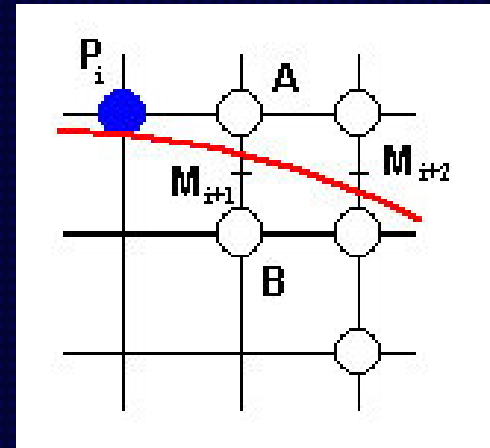
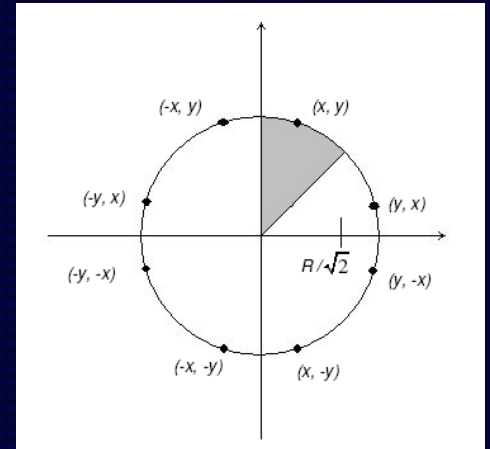


Konwersja okręgu

Algorytm z punktem środkowym

$$F(x,y) = x^2 + y^2 - R^2 = 0$$

$F(x,y) = 0$, dla punktów leżących na okręgu,
 $F(x,y) > 0$, dla punktów leżących poza okręgiem,
 $F(x,y) < 0$, dla punktów leżących wewnątrz okręgu.



Konwersja okręgu

Algorytm z punktem środkowym

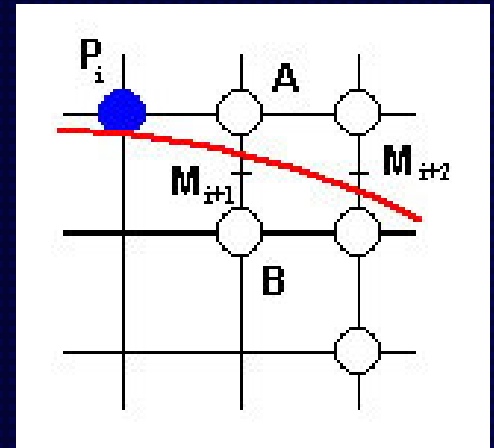
Wprowadzamy zmienną decyzyjną d_i , określoną w punkcie M wzorem

$$d_i = F(x_i + 1, y_i - \frac{1}{2}) = (x_i + 1)^2 + (y_i - \frac{1}{2})^2 - R^2$$

Dla $d_i < 0$ wybieramy piksel A. Następny punkt środkowy znajdzie się w odległości 1 przyrostu wzdłuż osi 0X

$$d_n = F(x_i + 2, y_i - \frac{1}{2}) = (x_i + 2)^2 + (y_i - \frac{1}{2})^2 - R^2$$

czyli $d_n = d_i + (2x_i + 3)$ wówczas przyrost wynosi $\Delta_A = 2x_i + 3$



Konwersja okręgu

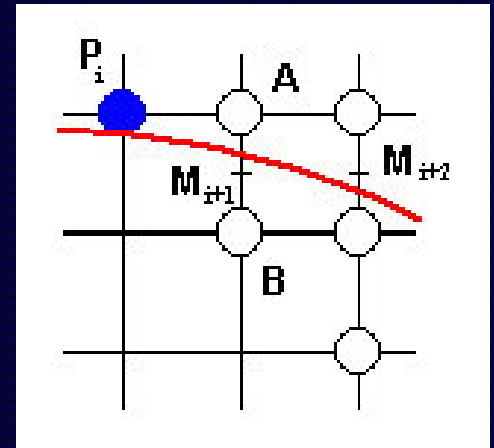
Algorytm z punktem środkowym

Dla $d_i \geq 0$ wybieramy piksel B. Następny punkt środkowy znajdzie się w odległości 1 przyrostu wzdłuż osi OX i OY

$$d_n = F(x_i + 2, y_i - \frac{3}{2}) = (x_i + 2)^2 + (y_i - \frac{3}{2})^2 - R^2$$

czyli $d_n = d_i + (2x_i - 2y_i + 5)$

wówczas przyrost wynosi $\Delta_B = 2x_i - 2y_i + 5$



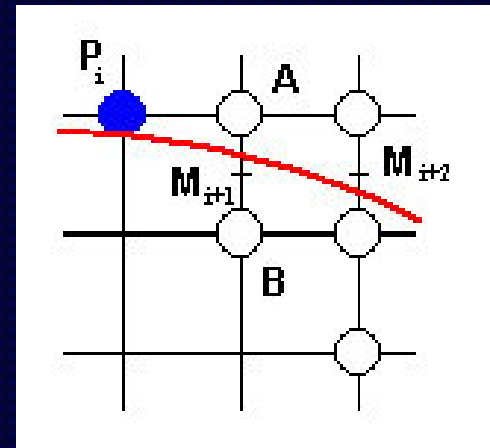
Konwersja okręgu

Algorytm z punktem środkowym

W przypadku odcinka przyrost był wartością stałą, w przypadku okręgu przyrosty są zmienne, gdyż zależą od wartości współrzędnych punktu P. Punkt ten jest zatem *punktem odniesienia*.

Obliczenia iteracyjne:

- wybór piksela na podstawie wartości (znaku) zmiennej decyzyjnej,
- uaktualnienie zmiennej decyzyjnej o wartość przyrostu.



Wypełnianie obszaru

Wypełnianie ograniczonego obszaru:

- barwą,
- gradientem,
- wzorem.

Problemy związane z wypełnianiem:

- wybór pikseli, które powinny zostać wypełnione,
- wybór metody wypełniania pikseli.

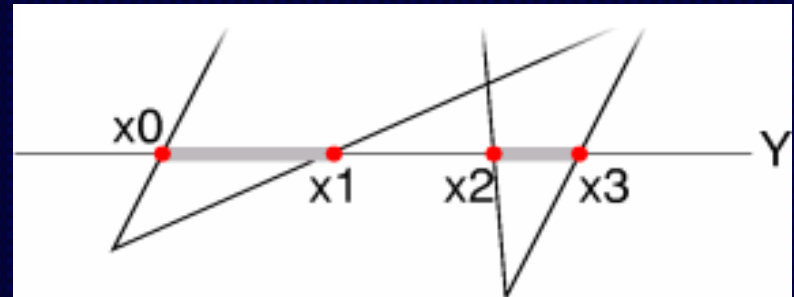
Wypełnianie wielokątów

Algorytm wypełniania wielokąta:

Obraz jest analizowany wiersz po wierszu. Rozpatrywane są kolejne linie przecinające wielokąt i wypełniane są segmenty (poziome ciągi) pikseli leżących wewnątrz niego.

Każdemu pikselowi segmentu przypisywana jest stała wartość (*spójność segmentowa*).

O wyborze pikseli decyduje reguła parzystości z założeniem, że przy przejściu przez pierwszą krawędź Wartość bitu zmienia się z parzystej na nieparzystą, a przy kolejnych przecięciach przyjmuje wartość przeciwną. Rysowane są tylko piksele o nieparzystej wartości.

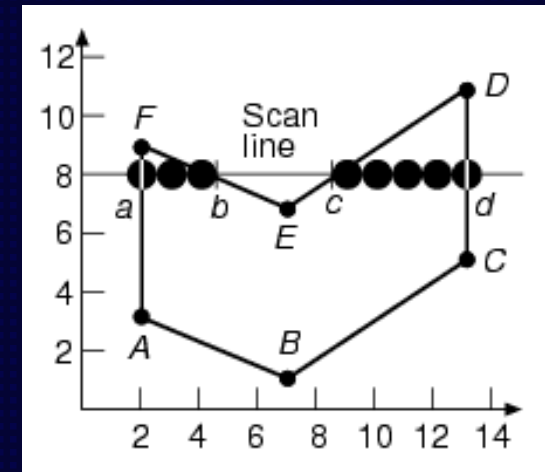


Wypełnianie wielokątów

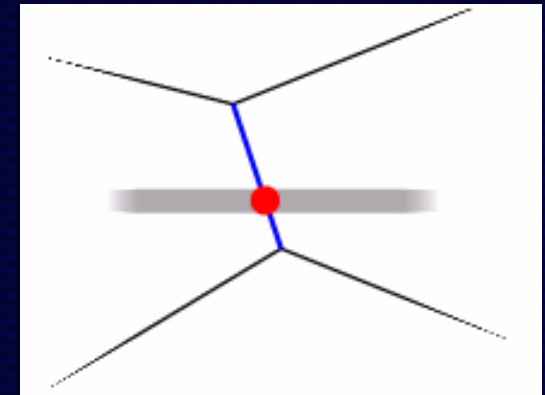
Piksele końcowe segmentów wyznaczone są przy zastosowaniu algorytmu przyrostowego.

Punkty przecięć a i d – współrzędne całkowite.

Punkty b i c – współrzędne zmienno-przecinkowe, konieczna aproksymacja.



Problemy z wielokątami, które posiadają wspólne krawędzie – powielanie punktów brzegowych.



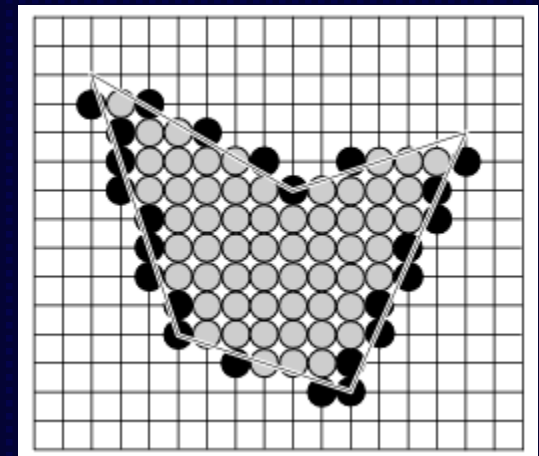
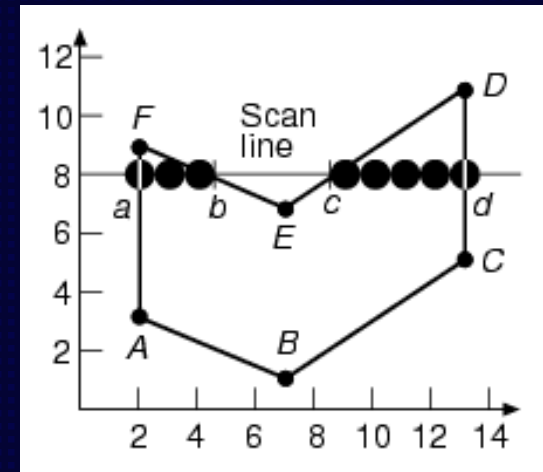
Wypełnianie wielokątów

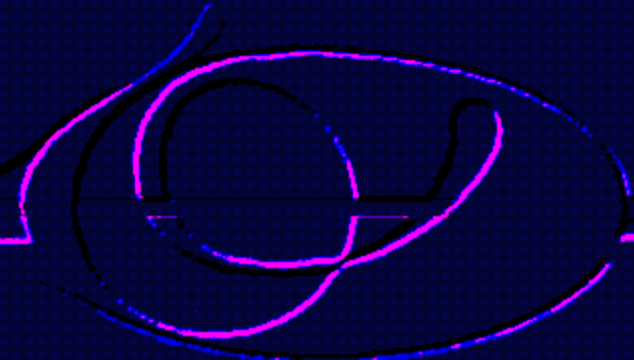
Położenie punktów końcowych jest ustalane zgodnie z zasadą punktu środkowego.

Punkty te dla poszczególnych segmentów w danej linii zapamiętywane są w tablicy.

Niektóre z punktów mogą znaleźć się poza obrysem wielokąta, co może zakłócać obszary wielokątów sąsiednich.

Algorytm musi zostać ulepszony w taki sposób, aby uwzględniał jedynie punkty wewnętrzne wielokąta.

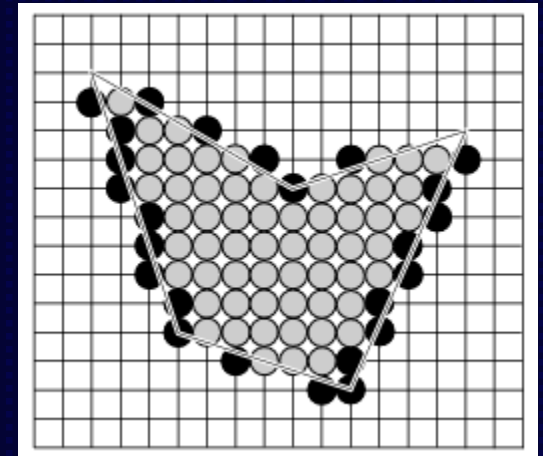




Wypełnianie wielokątów

Rozwiązanie:

- jeżeli jesteśmy na zewnątrz wielokąta (i przesuwamy się w prawo), to wartość x jest zaokrąglana w górę, jeżeli jesteśmy wewnątrz – wartość x jest zaokrąglana w dół,
- przy przecięciach o współrzędnych całkowitych – pierwszy piksel segmentu (od lewej) traktowany jest jako wewnętrzny, a ostatni (po prawej) jako zewnętrzny,
- w przypadku wspólnych wierzchołków – uwzględniany dla danej krawędzi jest piksel o wartości y_{\min} , nie uwzględniany jest y_{\max} ,
- krawędzie poziome „dolne” są rysowane, „górne” – nie są.

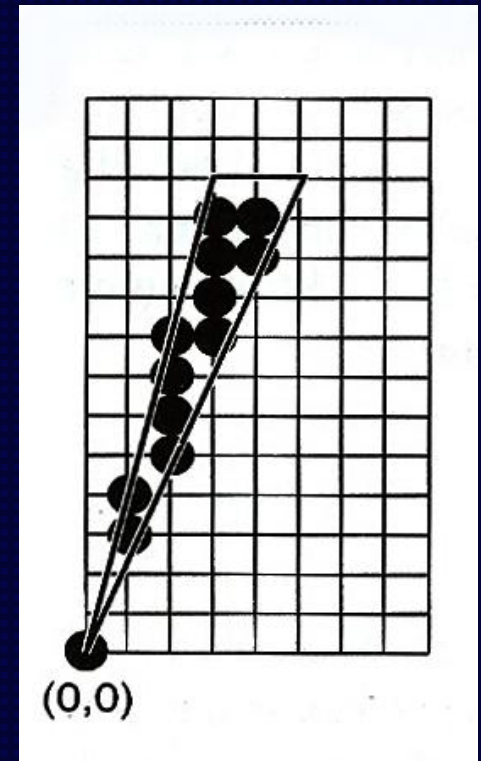


Wypełnianie wielokątów

„Drzazgi”

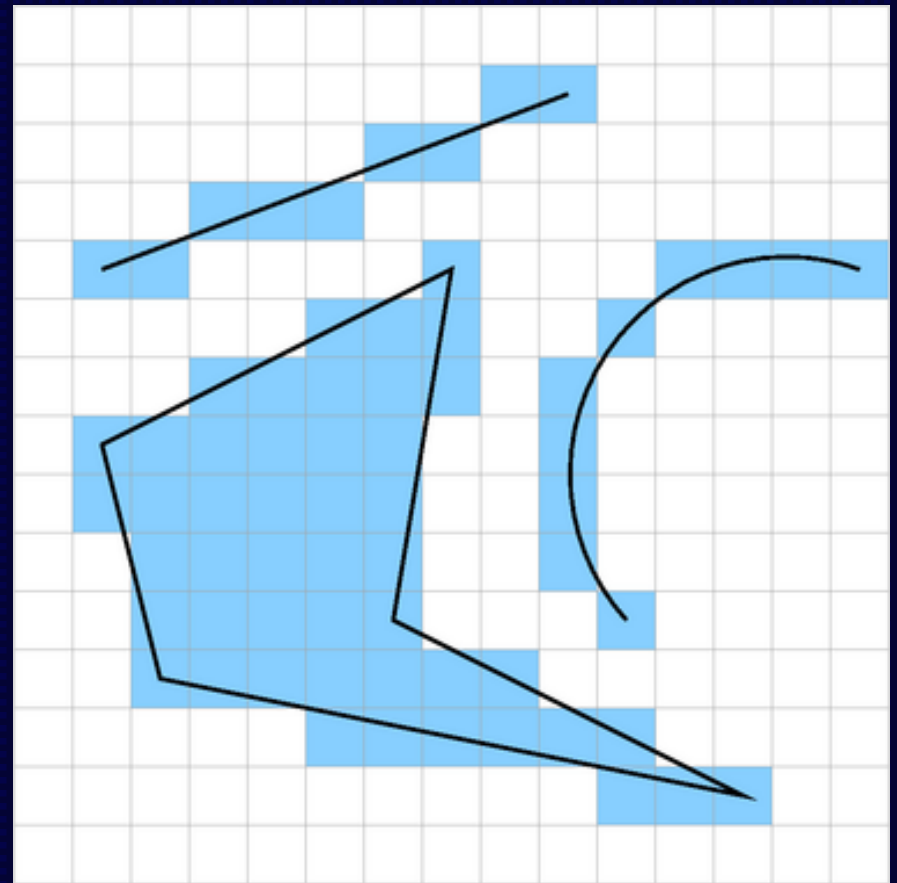
Negatywny efekt dyskretyzacji obrazu.

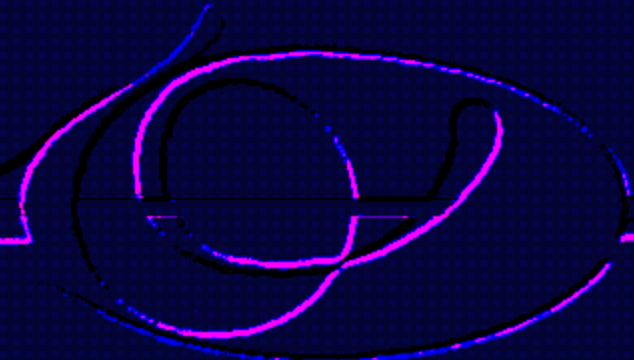
W wielokątach o krawędziach, między którymi jest niewielki kąt tworzą się „drzazgi”. Segmenty pikseli zawartych we wnętrzu wielokąta mają niewielką lub zerową długość.



Rasteryzacja prymitywów

Konwersja odcinka,
fragmentu okręgu
oraz wielokąta



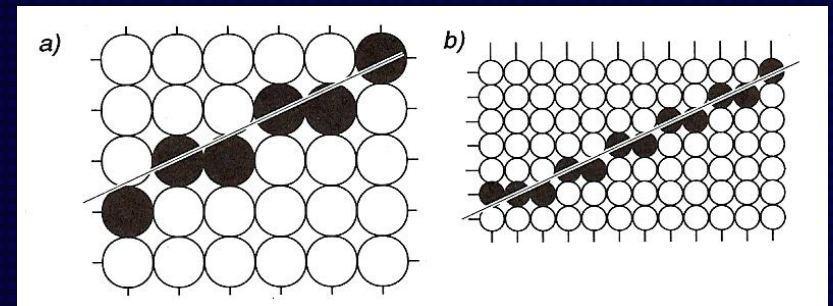


Metody usuwania zakłóceń

Zwiększenie rozdzielczości obrazu

Po wykonaniu tej operacji zmniejszają się odległości pikseli od „idealnego” odcinka. Pamiętać należy o tym, że przy dwukrotnym zwiększeniu rozdzielczości na każdej z osi, liczba pikseli zwiększa się czterokrotnie – wzrost objętości pliku.

- a) odcinek po konwersji algorytmem z punktem środkowym,
- b) ten sam odcinek przy podwójnej rozdzielczości

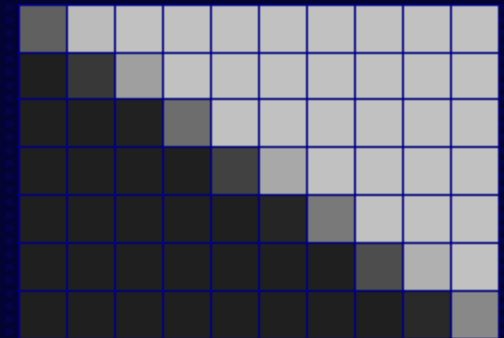
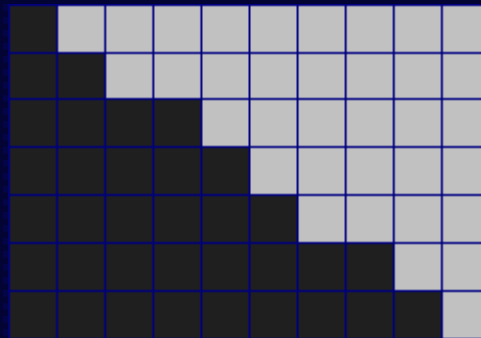
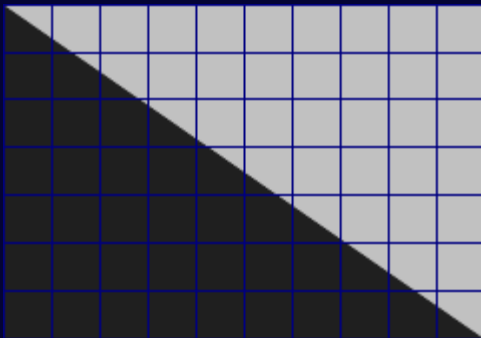


Metody usuwania zakłóceń

Antyaliasing

Aliasing polega na pojawianiu się "schodków" na liniach ukośnych występujących w obrazach rastrowych.

Do redukcji tego negatywnego efektu stosuje się tzw. *antyaliasing*, którego skutkiem ubocznym jest zmniejszenie kontrastowości obrazu.



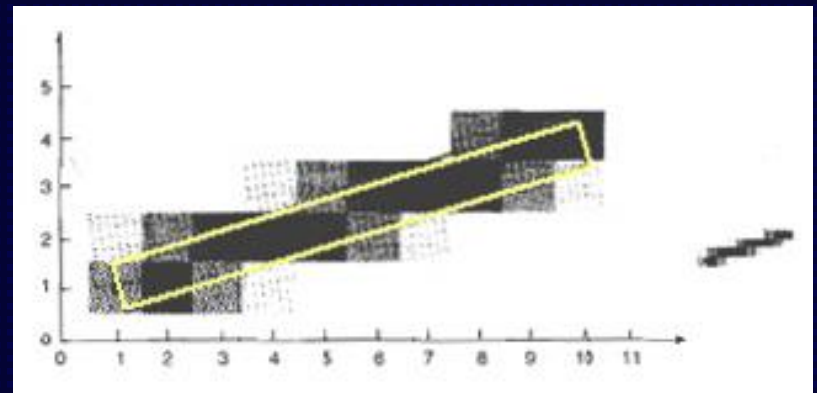
Metody usuwania zakłóceń

Antyaliasing

„Idealny” odcinek o zerowej grubości w obrazie rastrowym przyjmuje minimalną grubość 1 piksela.

Można go wyobrazić sobie jako prostokąt o szerokości 1. Odcinki pionowe i poziome pokrywają się z siatką rastra.

Dla odcinków ukośnych przeprowadzana jest analiza, w jakim procencie prostokąt pokrywa dany piksel (pole kwadratu). Obliczona składowa barwy jest mieszana z barwą tła.



Bezwagowe próbkowanie powierzchni

Metody usuwania zakłóceń

Antyaliasing

Wagowe próbkowanie powierzchni

Założenie – powierzchnia położona blisko środka piksela powinna mieć większy wpływ na jego barwę niż taka sama powierzchnia znajdująca się w większej odległości.

