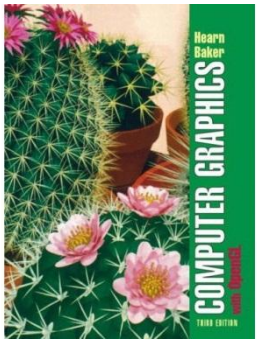
Grafika Komputerowa 1: Wstęp

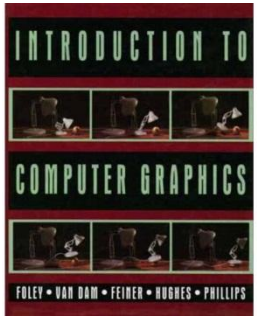
Wykładowca:

- Prof. dr hab. inż. Ryszard S. Choraś
- choras@utp.edu.pl

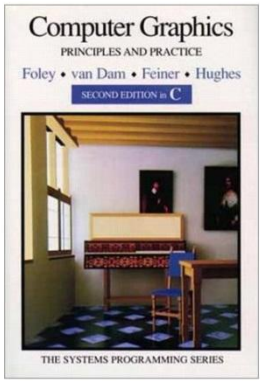
Literatura



“Computer Graphics with OpenGL”, D. Hearn & M. P. Baker, Prentice Hall, 2003



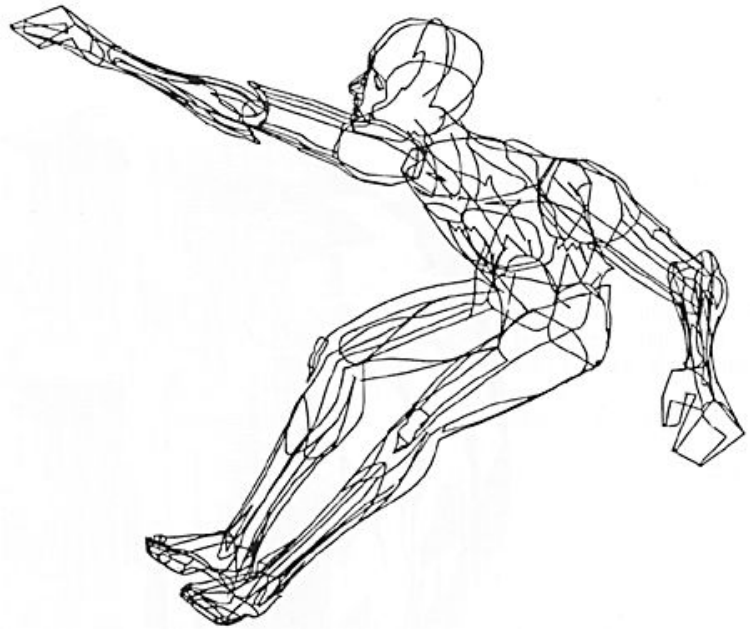
“Introduction to Computer Graphics”, J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes & R.L. Phillips, Addison Wesley, 1997



“Computer Graphics: Principles and Practice”, J.D. Foley, A. van Dam, S.K. Feiner & J.F. Hughes, Addison Wesley, 1995

Co to jest Grafika Komputerowa?

Termin Grafika Komputerowa (C**omputer graphics)** został wprowadzony w 1960 przez William Fetter do opisu nowej metody projektowania dla koncernu Boeing



Fetter stworzył serię szeroko reprodukowanych obrazów na ploter pokazujących projektowanie kokpitu, stosując 3D model ciała człowieka

Interaktywna GK

Mouse
Tablet and stylus
Force feedback device
Scanner
Live video streams

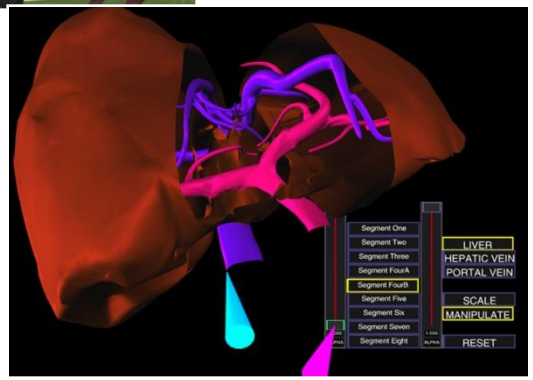
Screen
Paper-based printer
Video recorder
Projector
VR/AR headset

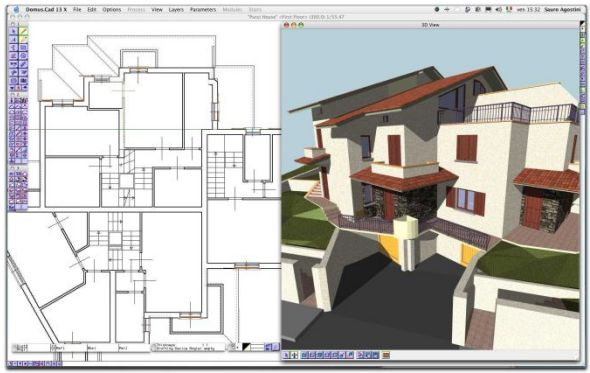
Sketchpad, rozwinięty w 1960, był pierwszą interaktywną aplikacją GK.

Wykorzystywał pióro świetlne, pad i monitor pozwalający użytkownikowi tworzyć dokładne rysunki

Dr. Ivan E. Sutherland developed Sketchpad as part of his PhD work. He went on to be a hugely influential computer scientist working in areas as diverse as graphics, circuit design, robotics and computer hardware

AUBURN
Take Snap Shot


Segment One
Segment Two
Segment Three
Segment FourA
Segment FourB
Segment Five
Segment Six
Segment Seven
Segment Eight
LIVER
HEPATIC VEIN
PORTAL VEIN
SCALE
MANIPULATE
RESET

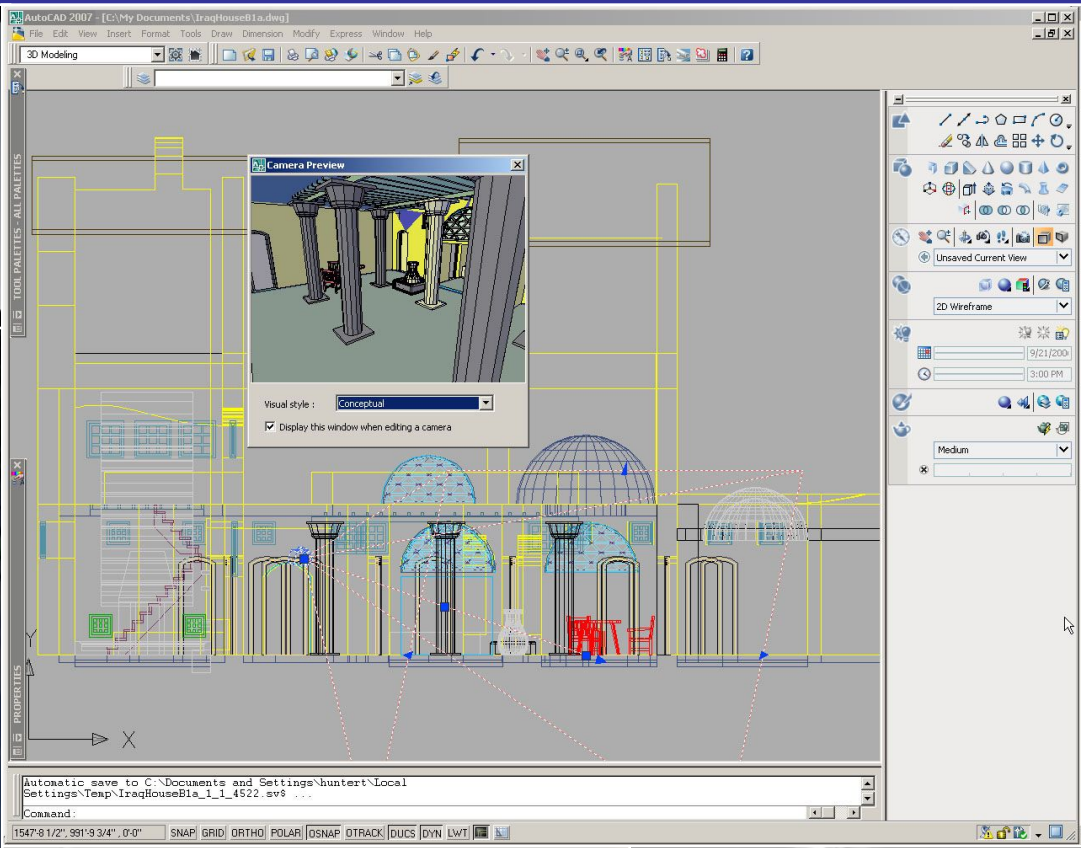


Aplikacje GK

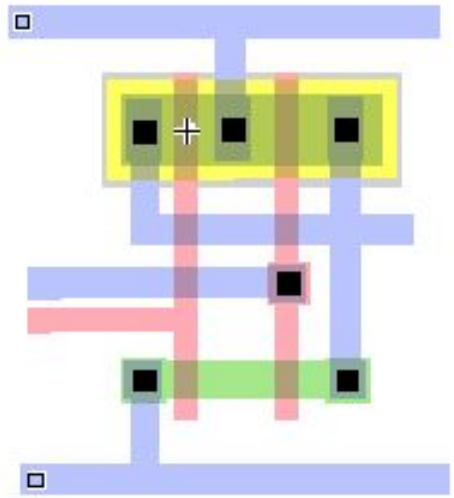
Przykładowe aplikacje w których wykorzystujemy GK :

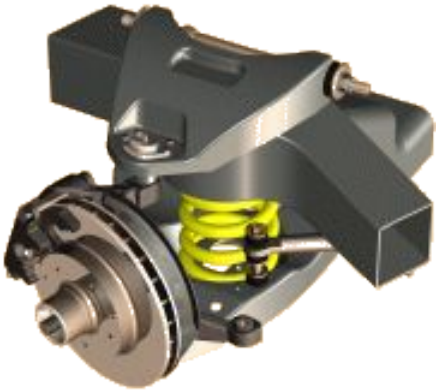
- Computer aided design
- Scientific visualisation
- Films
- Games
- Virtual/Augmented Reality

Computer Aided Design

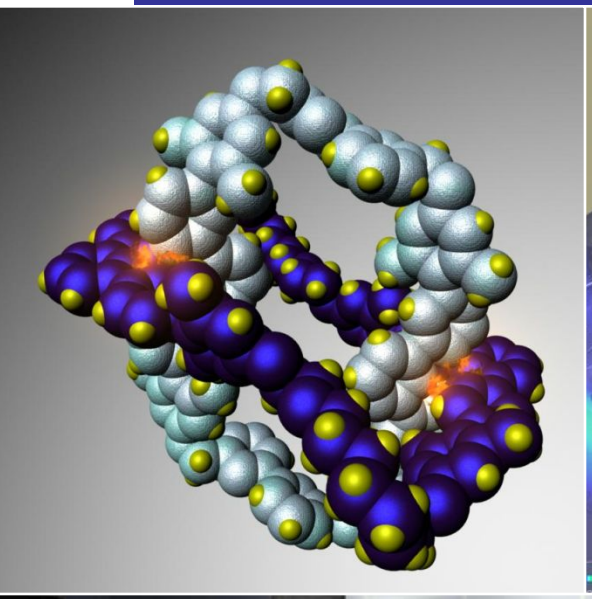


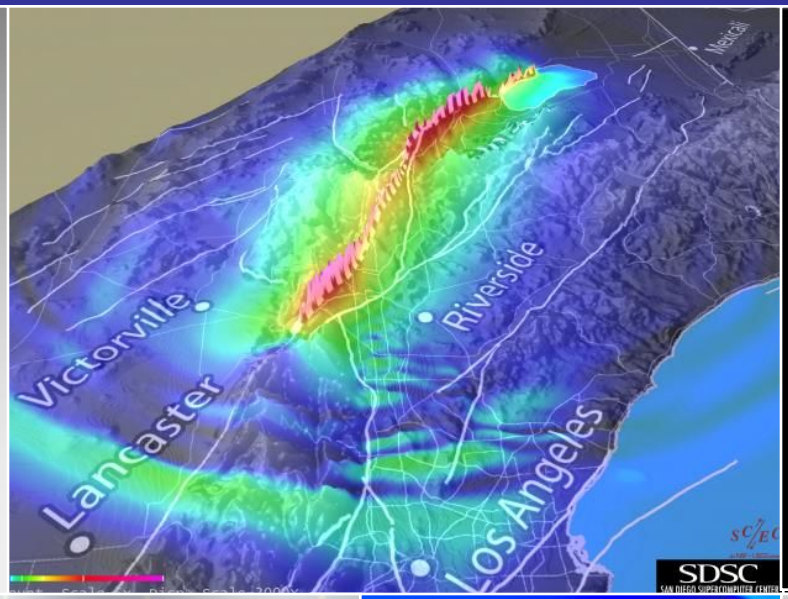
Computer Aided Design

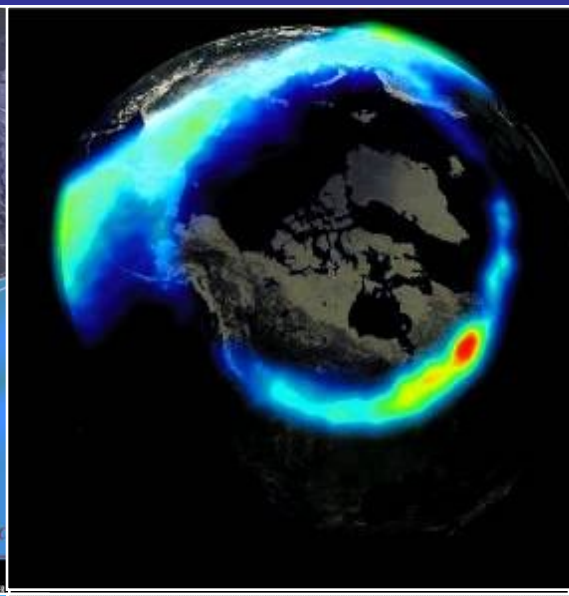


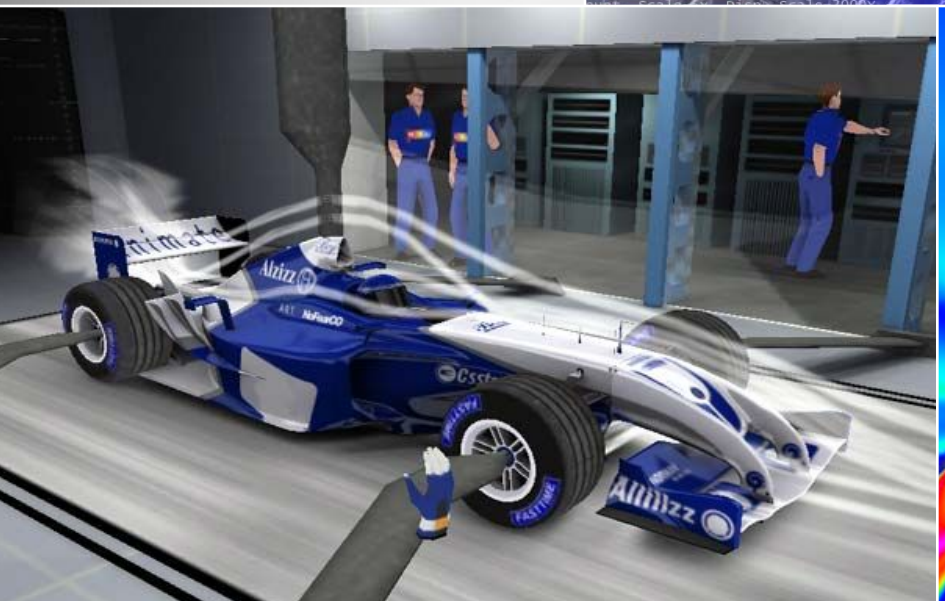


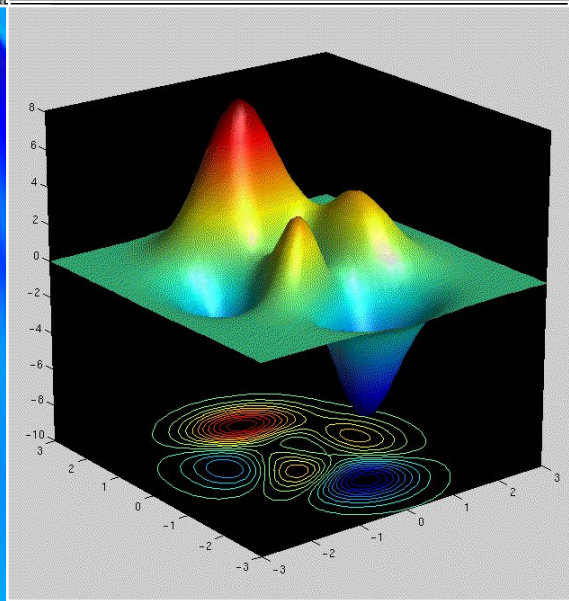
Scientific Visualisation



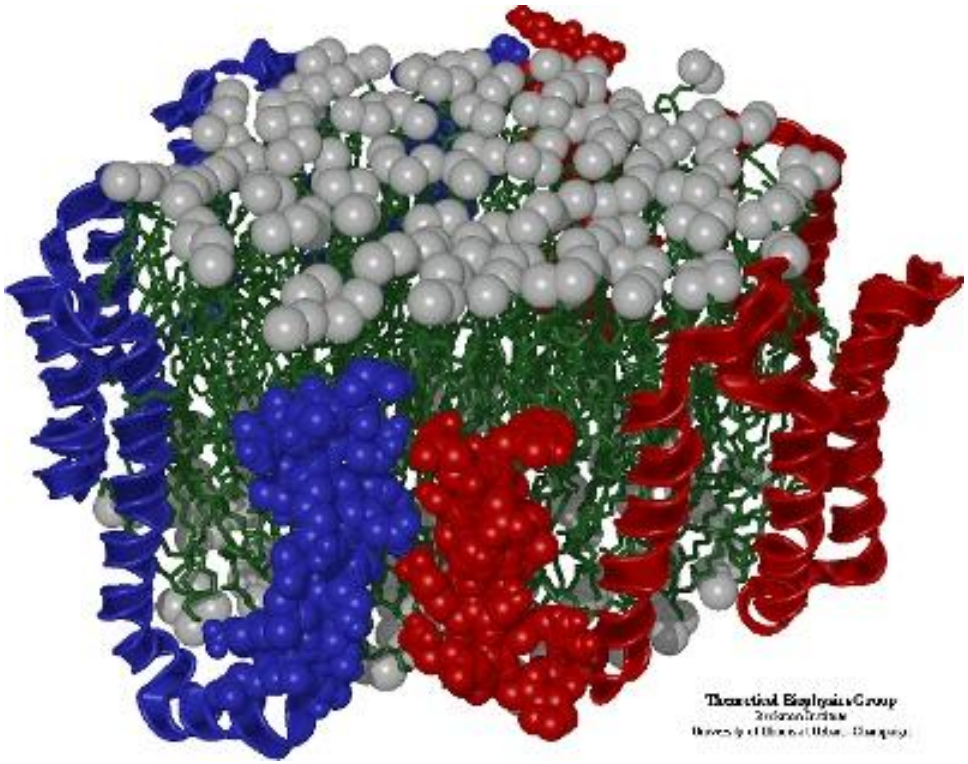


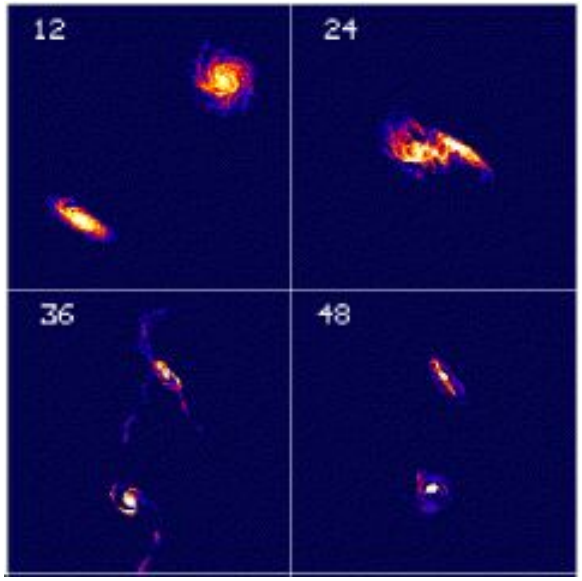


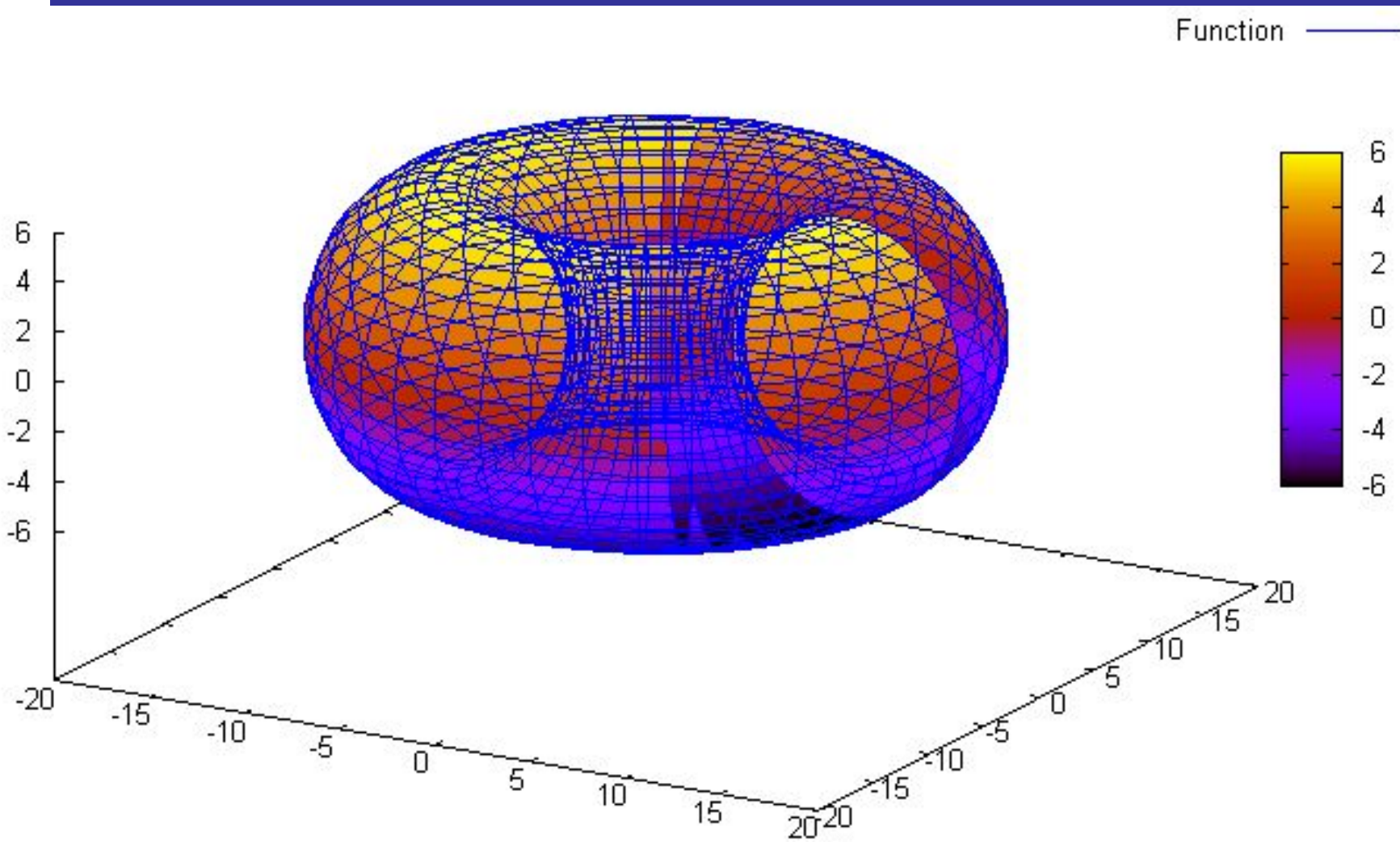




Scientific Visualisation





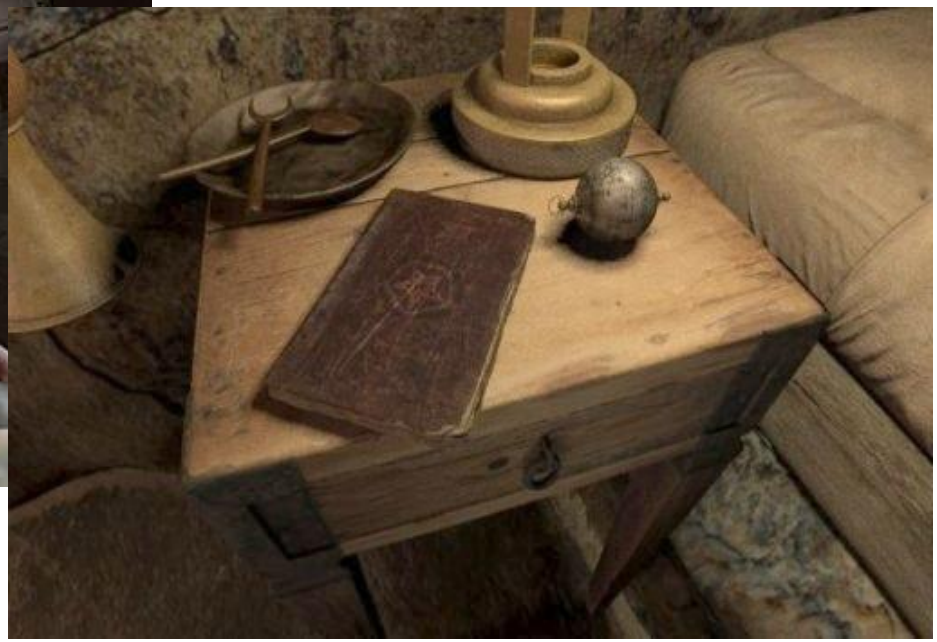

Function
6
4
2
0
-2
-4
-6
-20
-15
-10
-5
0
5
10
15
20
20
15
10
5
0
-5
-10
-15
-20

Films

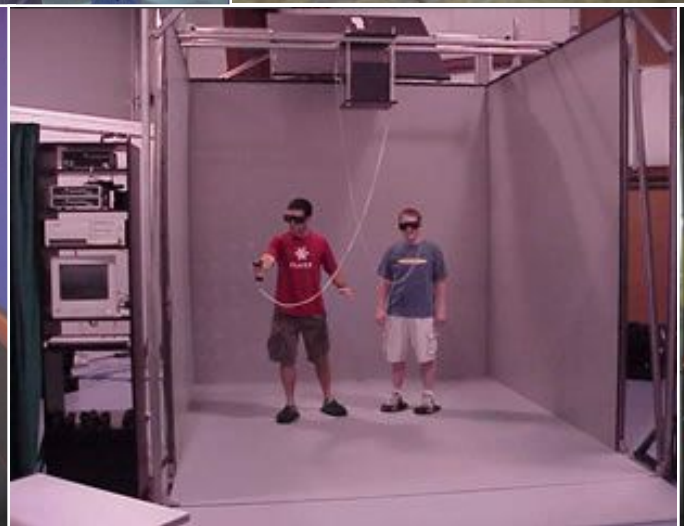
Animacje w Grafice Komputerowej (Filmy)

Games

Gry są b. ważne w Grafice Komputerowej



Virtual Reality



Zobrazowanie medyczne

GK2:
Matematyka

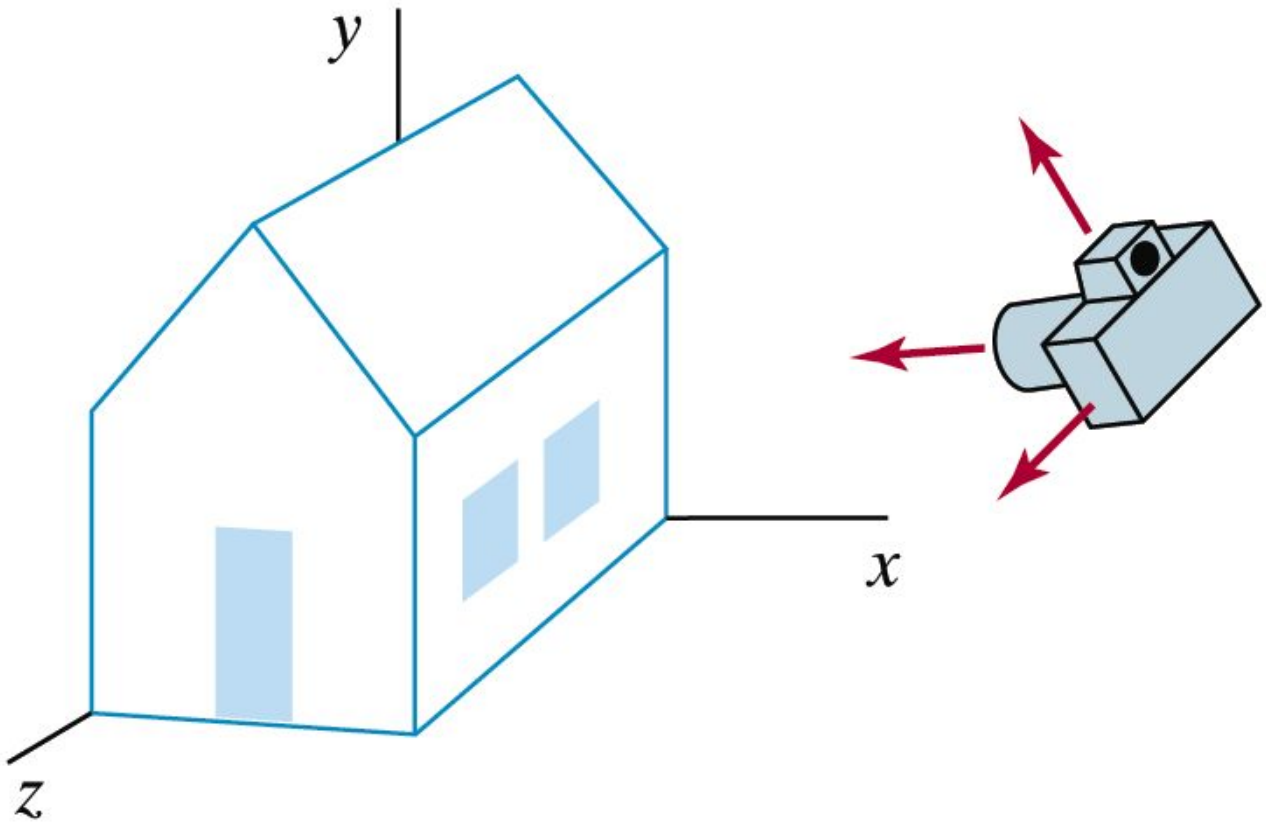
Wstęp

Grafika komputerowa to matematyka!

Dziasiaj omówimy zagadnienia:

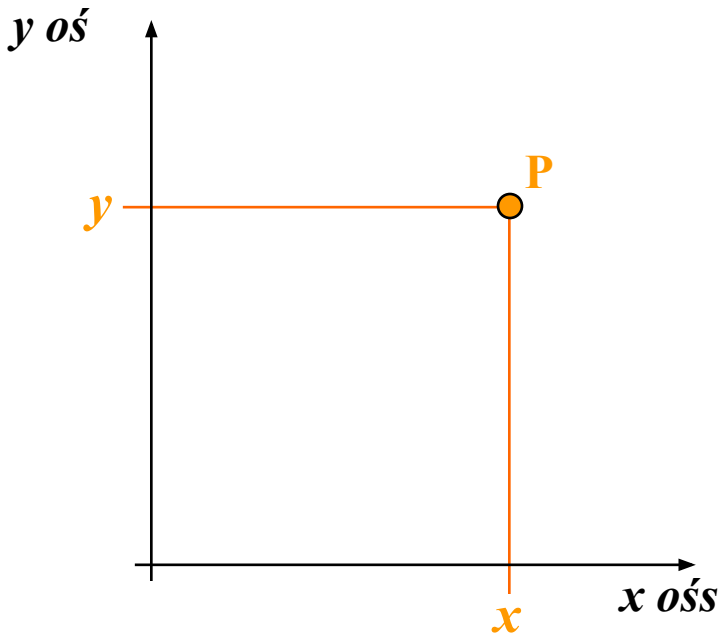
- Układ współrzędnych
- Punkty & linie
- Wektory
- Macierze

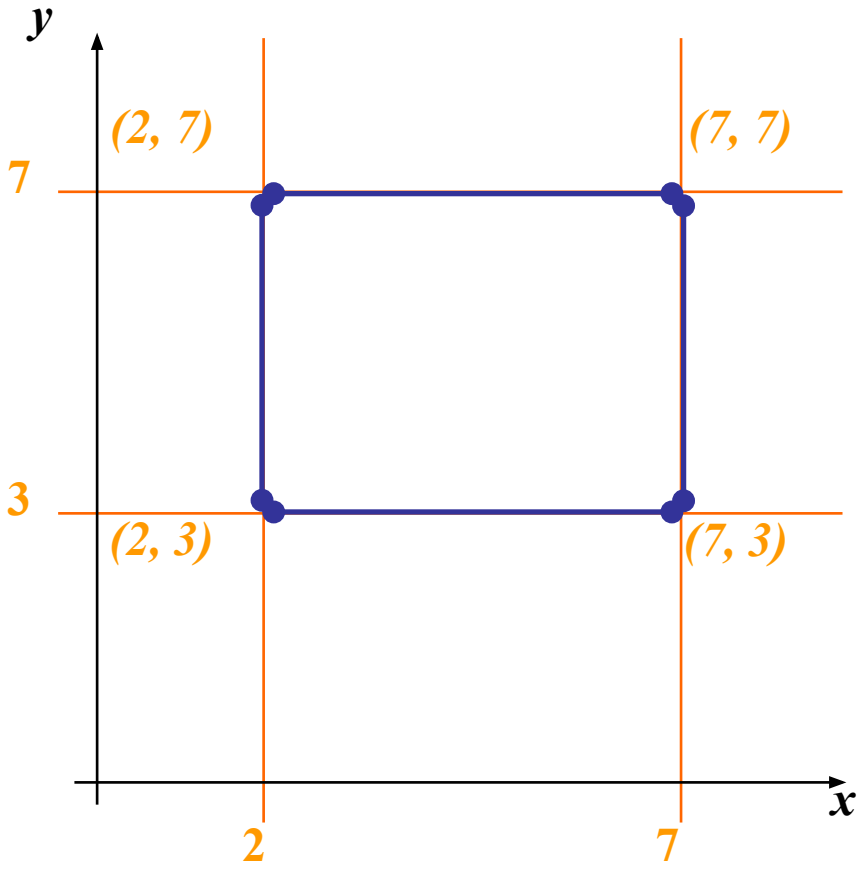
Idea



Układ współrzędnych – 2D

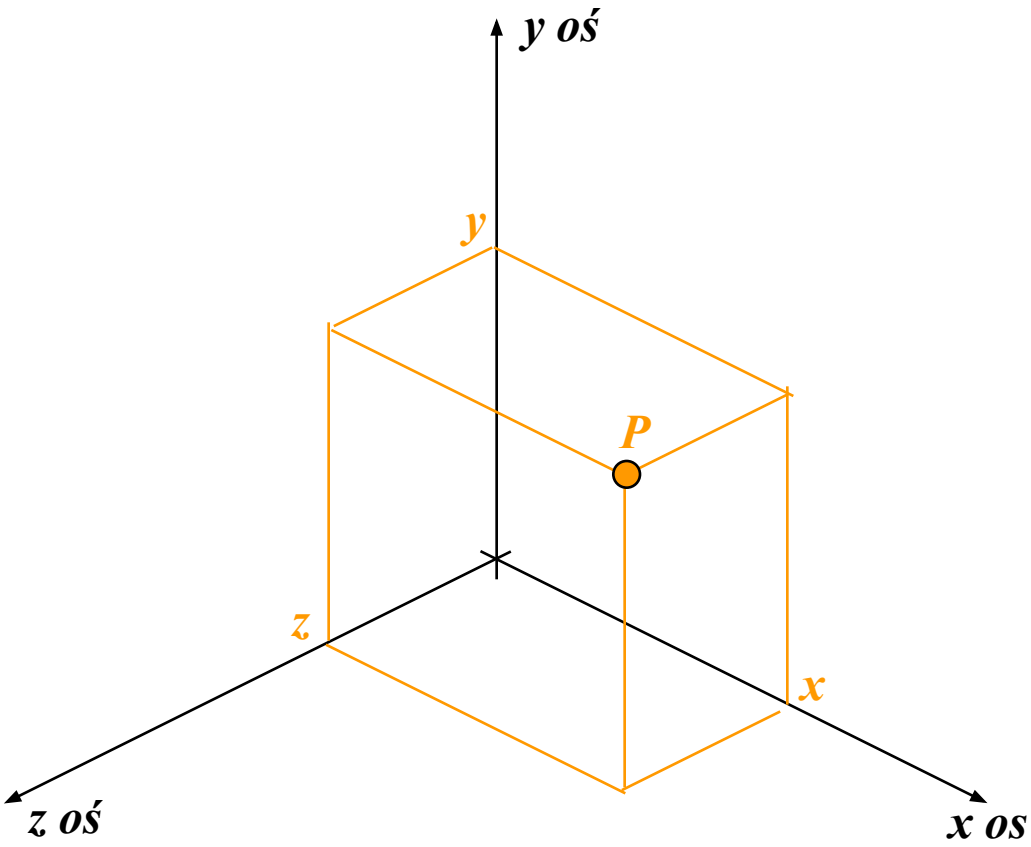
Dla 2D scen my stosujemy dwu wymiarowy ukł. Współrzędnych kartezjanskich.




y
(2, 7)
(7, 7)
7
3
(2, 3)
(7, 3)
x
2
7

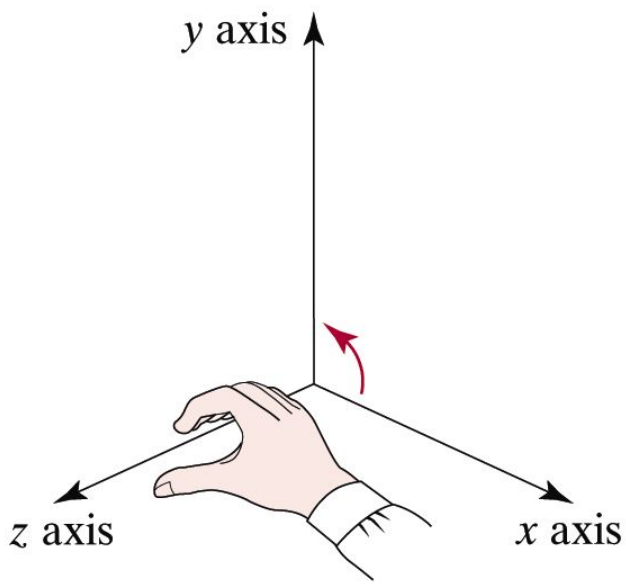
Układ współrzędnych – 3D

Dla trój wymiarowych scen stosujemty ukł. współrzędnych



Lefoskrętny lub Prawoskretny?

Mamy dwa różne podejścia do 3D współrzędnych – lewoskretny lub prawoskrętny



Right-Hand
Reference System

Zazwyczaj stosujemy prawoskretny ukł.

Left-Hand
Reference System

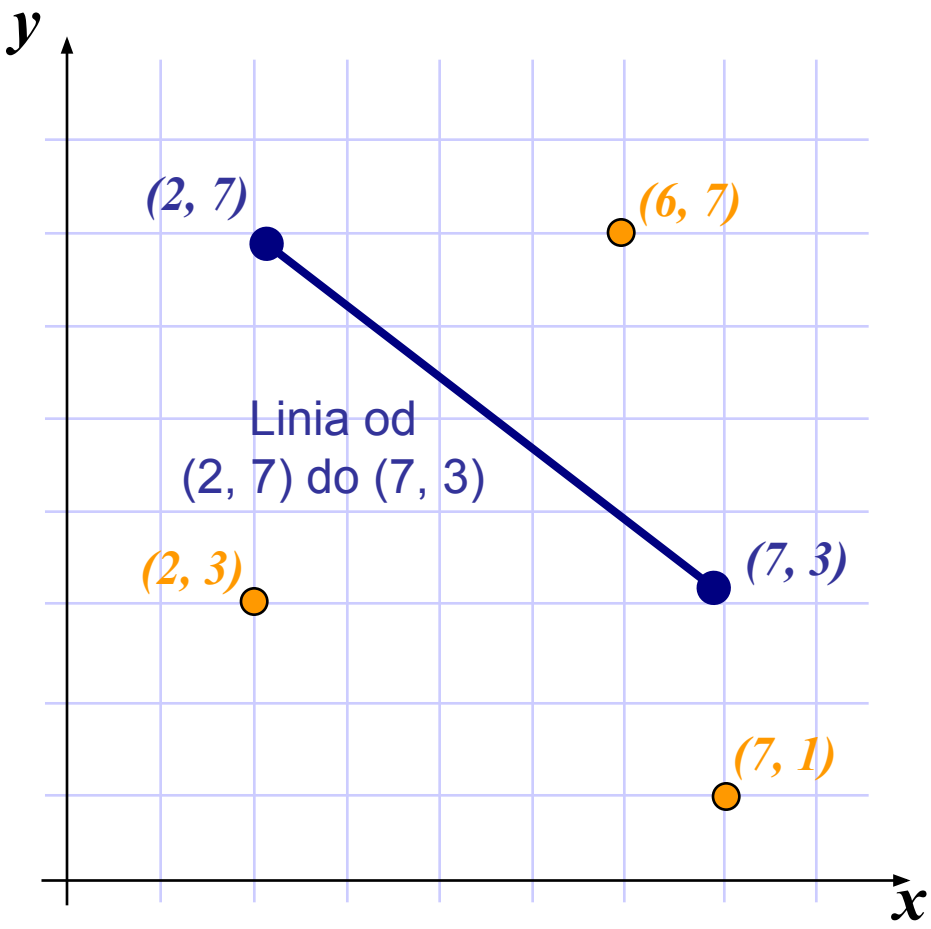
Punkty & Linie

Punkty:

- Punkt w 2D jest okreslony przez (x, y)
- W 3D punkt jest określony przez trójkę (x, y, z)

Linie:

- linia jest zdefiniowana przez punkt startowy i punkt końcowy
  - W 2d: (x_{start}, y_{start}) do (x_{end}, y_{end})
  - W 3d: $(x_{start}, y_{start}, z_{start})$ do $(x_{end}, y_{end}, z_{end})$


y
(2, 7)
(6, 7)
Linia od
(2, 7) do (7, 3)
(2, 3)
(7, 3)
(7, 1)
x

Równanie Linii

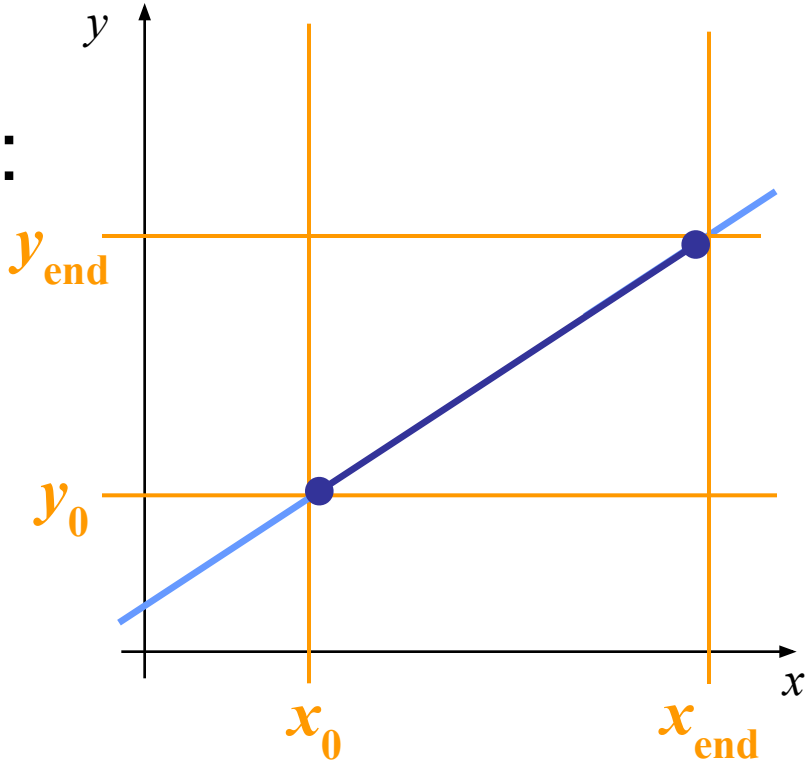
Równanie linii w terminach nachylenia:

$$y = m \cdot x + b$$

gdzie:

$$m = \frac{y_{end} - y_0}{x_{end} - x_0}$$

$$b = y_0 - m \cdot x_0$$



Ex.

Narysuj odcinek linii dany przez równanie:

$$y = \frac{3}{5}x + \frac{4}{5}$$

Okreslamy y współrzędną dla każdej x współrzędnej

Ex.

Dla każdej wartości x okreslamy wartość y :

$$y(2)=\frac{3}{5}\cdot 2+\frac{4}{5}=2$$

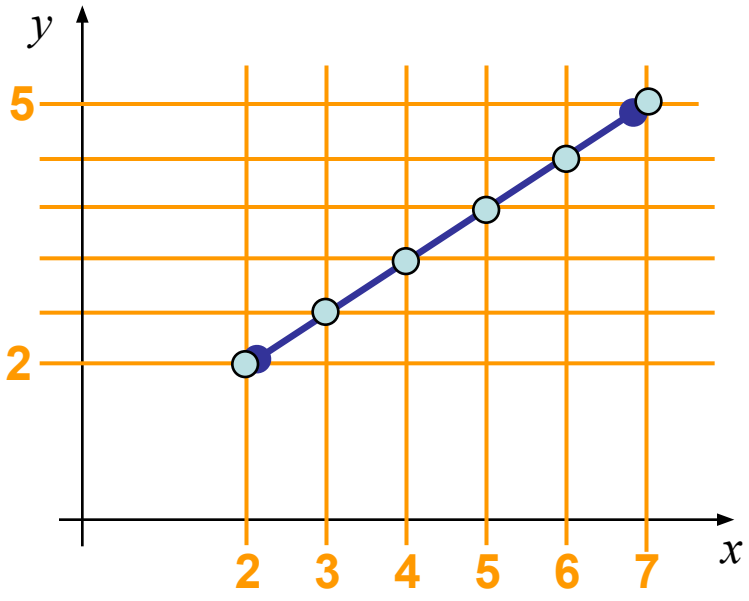
$$y(3)=\frac{3}{5}\cdot 3+\frac{4}{5}=2\frac{3}{5}$$

$$y(4)=\frac{3}{5}\cdot 4+\frac{4}{5}=3\frac{1}{5}$$

$$y(5)=\frac{3}{5}\cdot 5+\frac{4}{5}=3\frac{4}{5}$$

$$y(6)=\frac{3}{5}\cdot 6+\frac{4}{5}=4\frac{2}{5}$$

$$y(7)=\frac{3}{5}\cdot 7+\frac{4}{5}=5$$



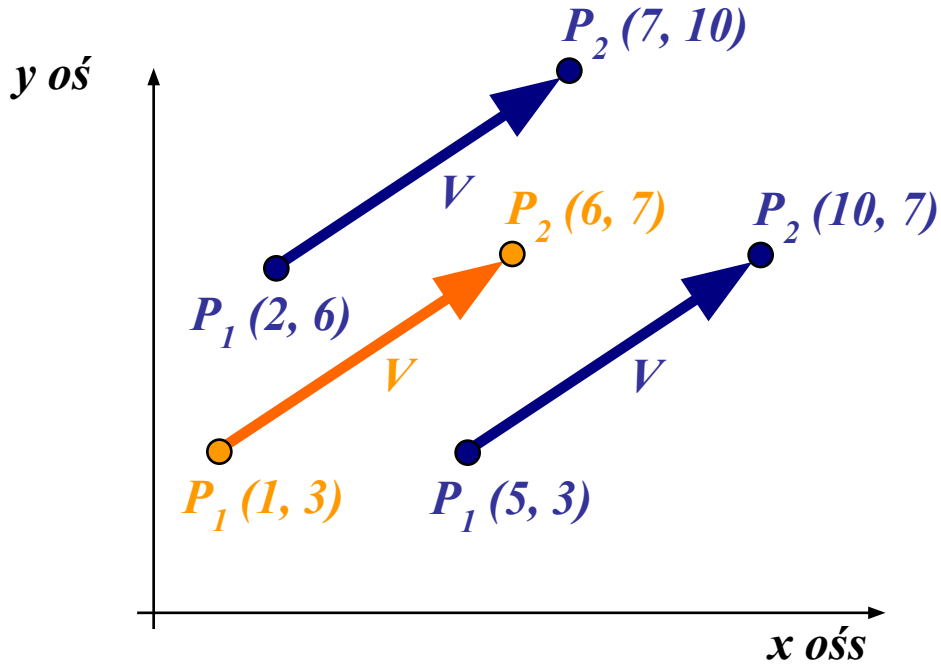
Wektory

Wektory:

- Wektor jest definiowany jako różnica między dwoma punktami
- Ważną cechą wektora jest kierunek i długość

Wektory (2D)

Do określenia wektorów między dwoma punktami



$$V = P_2 - P_1$$

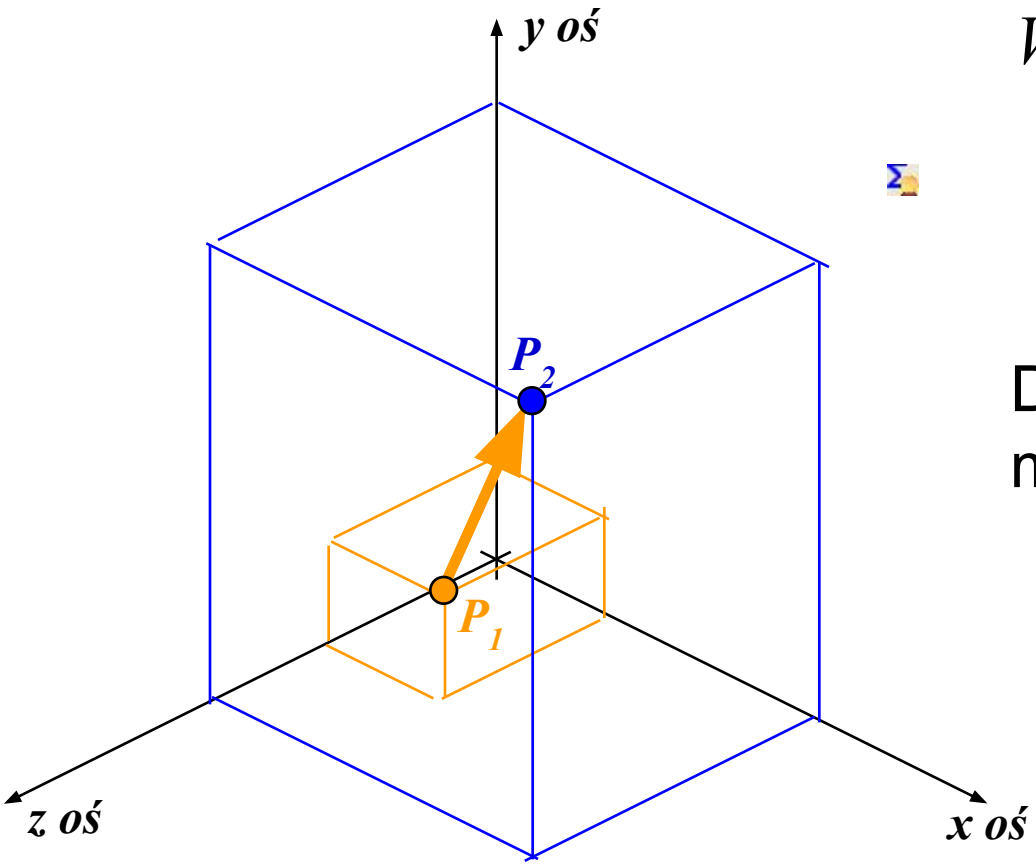
$$= (x_2 - x_1, y_2 - y_1)$$

$$= (6 - 1, 7 - 3)$$

$$= (5, 4)$$

Wektory (3D)

W 3D wektor jest obliczany podobnie



$$V = P_2 - P_1$$

$$= (x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

$$= (V_x, V_y, V_z)$$

Dla (2, 1, 3) to (7, 10, 5) my mamy

$$= (7-2, 10-1, 5-3)$$

$$= (5,9,2)$$

Operacje na wektorach

Mamy ważne operacje na wektorach:

- Obl. Dł. wektora
- Dodawanie wektorów
- Skalarne mnożenie wektorów
- Skalarny produkt
- Wektorowy produkt

Dł. wektora

Dł. Wektora jest obliczana w 2D jak:

$$|V| = \sqrt{V_x^2 + V_y^2}$$

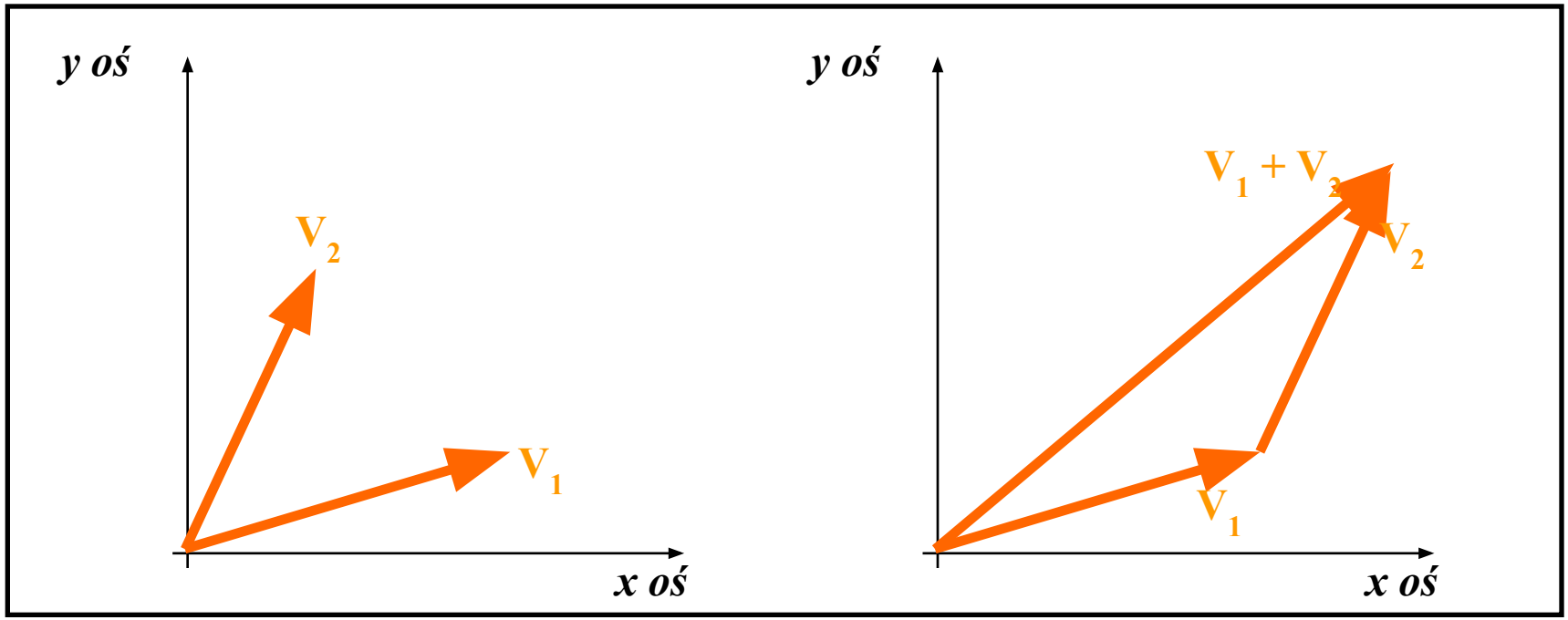
i w 3D jak:

$$|V| = \sqrt{V_x^2 + V_y^2 + V_z^2}$$

Dodawanie wektorów

Suma dwóch wektorów jest obliczana jak proste dodawanie odpowiednich współrzędnych

$$V_1 + V_2 = (V_{1x} + V_{2x}, V_{1y} + V_{2y})$$

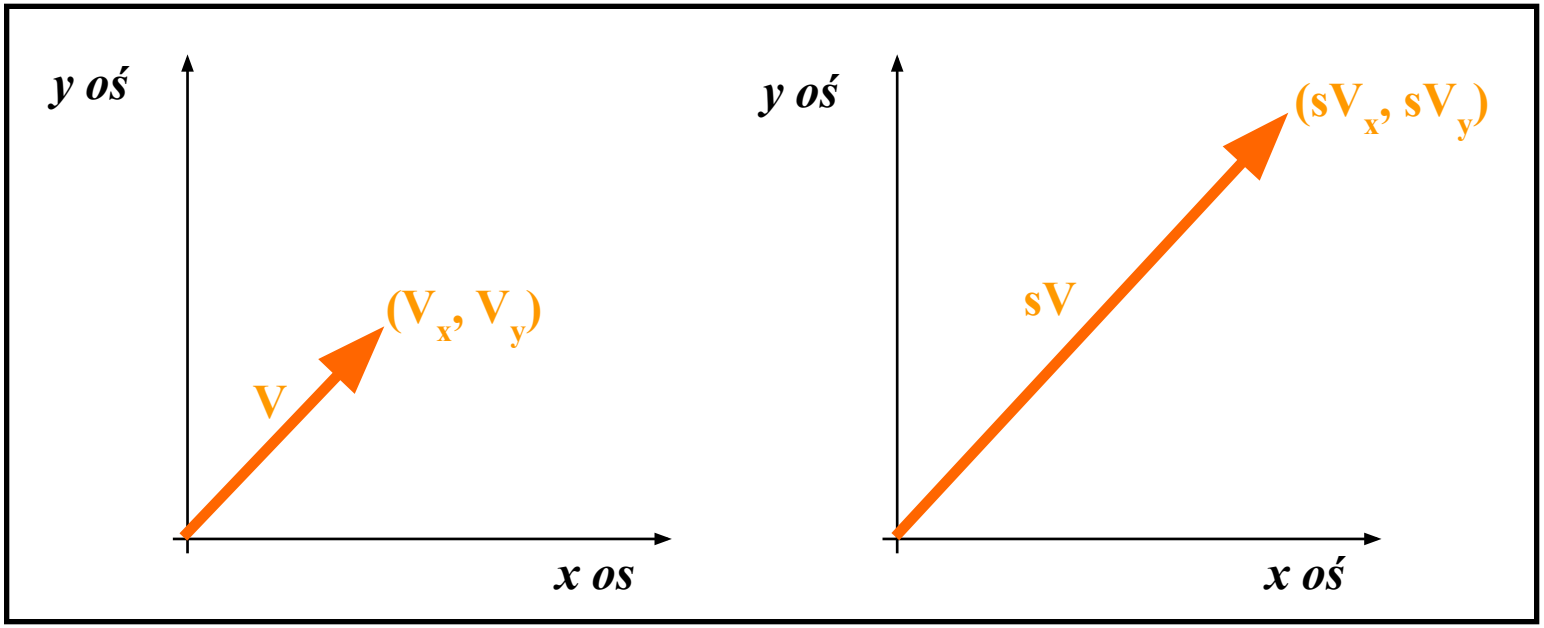


Podobnie w 3D

Skalarne mnożenie

Mnożenie wektora przez skalar polega na mnożeniu każdej współrzędnej wektora przez skalar

$$sV = (sV_x, sV_y)$$



– Skalar produkt
– Wektor produkt

Macierze

Macierz jest tablicą liczb

$$\begin{bmatrix} 1 & 11 & 13 \\ 10 & 4 & -3 \\ 2 & 0 & 6 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix} \quad \begin{bmatrix} 4.3 \\ 6.7 \\ 1.2 \end{bmatrix} \quad \begin{bmatrix} 4 & 8 & 15 \\ 16 & 23 & 42 \end{bmatrix}$$

Ważne operacje na macierzach:

- Skalarne mnożenie
- Dodawanie macierzy
- Mnożenie macierzy
- Transponowanie macierzy
- Wyznacznik (Determinant) macierzy
- Odwrotność macierzy

Mnożenie skalarne

$$s*\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} s*a & s*b & s*c \\ s*d & s*e & s*f \\ s*g & s*h & s*i \end{bmatrix}$$

Ex.:

$$3*\begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \\ 14 & 16 & 18 \end{bmatrix} = \begin{bmatrix} 6 & 12 & 18 \\ 24 & 30 & 36 \\ 42 & 48 & 54 \end{bmatrix}$$

Dodawanie

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} + \begin{bmatrix} r & s & t \\ u & v & w \\ x & y & z \end{bmatrix} = \begin{bmatrix} a+r & b+s & c+t \\ d+u & e+v & f+w \\ g+x & h+y & i+z \end{bmatrix}$$

Ex.:

$$\begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \\ 14 & 16 & 18 \end{bmatrix} + \begin{bmatrix} 3 & 5 & 7 \\ 9 & 11 & 13 \\ 15 & 17 & 19 \end{bmatrix} = \begin{bmatrix} 5 & 9 & 13 \\ 17 & 21 & 25 \\ 29 & 33 & 37 \end{bmatrix}$$

Obie macierze mają taki sam wymiar

Mnożenie macierzy

Dwie macierze **A** i **B**

Mamy m przez n macierz **A** i n przez q macierz **B** wtedy mnożenie:

C=AB

gdzie **C** jest m przez q macierz której elementy są obl. następująco:

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{ki}$$

Ex.:

$$\begin{bmatrix} 0 & -1 \\ 5 & 7 \\ -2 & 8 \end{bmatrix}\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 0*1+(-1)*3 & 0*2+(-1)*4 \\ 5*1+7*3 & 5*2+7*4 \\ -2*1+8*3 & -2*2+8*4 \end{bmatrix} = \begin{bmatrix} -3 & -4 \\ 26 & 38 \\ 22 & 28 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}\begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 1*4+2*5+3*6 \end{bmatrix} = \begin{bmatrix} 32 \end{bmatrix}$$

$$\begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}\begin{bmatrix} 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 4*1 & 4*2 & 4*3 \\ 5*1 & 5*2 & 5*3 \\ 6*1 & 6*2 & 6*3 \end{bmatrix} = \begin{bmatrix} 4 & 8 & 12 \\ 5 & 10 & 15 \\ 6 & 12 & 18 \end{bmatrix}$$

$$AB \neq BA$$

Transpozycja

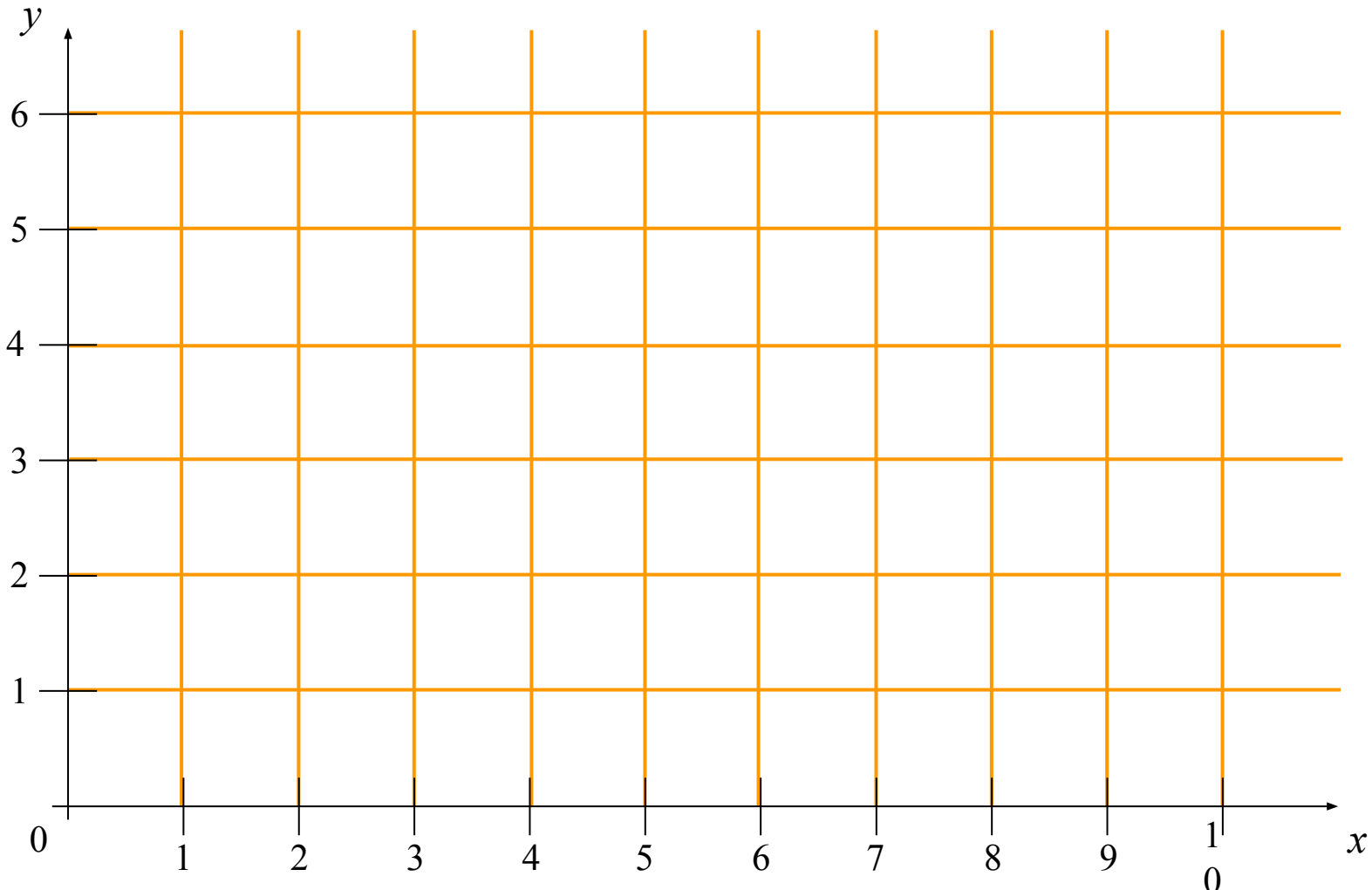
Ex.:

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}^T = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

– Wyznacznik (Determinant) macierzy
– Odwrotność macierzy

Ex. 1

Narysuj linię $y = \frac{1}{2}x + 2$ od $x = 1$ do $x = 9$



Ex. 2

Zrób następujące dodawania macierzy:

$$\begin{bmatrix} -11 & -19 & -15 & 5 \end{bmatrix} + \begin{bmatrix} -1 & -14 & -5 & 1 \end{bmatrix} = \begin{bmatrix} __ & __ & __ & __ \end{bmatrix}$$

$$\begin{bmatrix} 3 & 20 & -11 & 0 \\ -14 & 5 & -3 & 6 \\ 15 & 2 & 9 & -18 \\ -19 & 4 & -15 & 10 \end{bmatrix} + \begin{bmatrix} 16 & 10 & -12 & -11 \\ 10 & -15 & 15 & 5 \\ -1 & 14 & -9 & 0 \\ 3 & -3 & -16 & -5 \end{bmatrix} = \begin{bmatrix} __ & __ & __ & __ \\ __ & __ & __ & __ \\ __ & __ & __ & __ \\ __ & __ & __ & __ \end{bmatrix}$$

$$\begin{bmatrix} 13 & -3 & -4 \\ 19 & 9 & 8 \end{bmatrix} + \begin{bmatrix} -1 & -15 & -7 \\ -14 & 5 & 17 \end{bmatrix} = \begin{bmatrix} __ & __ & __ \\ __ & __ & __ \end{bmatrix}$$

Ex. 3

Mnożenie macierzy:

$$\begin{bmatrix} 8 & 15 & 19 & 4 \\ 7 & -4 & 12 & 3 \end{bmatrix} * \begin{bmatrix} -15 & 19 \\ -12 & -19 \\ 0 & -13 \\ 10 & 7 \end{bmatrix} = \begin{bmatrix} \underline{\qquad\qquad} & \underline{\qquad\qquad} \\ \underline{\qquad\qquad} & \underline{\qquad\qquad} \end{bmatrix} = \begin{bmatrix} \underline{\quad} & \underline{\quad} \\ \underline{\quad} & \underline{\quad} \end{bmatrix}$$

$$\begin{bmatrix} 16 & 10 & -12 & -11 \\ 10 & -15 & 15 & 5 \\ -1 & 14 & -9 & 0 \\ 3 & -3 & -16 & -5 \end{bmatrix} * \begin{bmatrix} 4 \\ 11 \\ 6 \\ 3 \end{bmatrix} = \begin{bmatrix} \underline{\qquad\qquad\qquad\qquad} \\ \underline{\qquad\qquad\qquad\qquad} \\ \underline{\qquad\qquad\qquad\qquad} \\ \underline{\qquad\qquad\qquad\qquad} \end{bmatrix} = \begin{bmatrix} \underline{\quad} \\ \underline{\quad} \\ \underline{\quad} \\ \underline{\quad} \end{bmatrix}$$

Ex. 4

Mnożenie macierzy przez skalar

$$6*\begin{bmatrix} 15 & 19 \\ 2 & 5 \\ 0 & -1 \\ 1 & 7 \end{bmatrix} = \begin{bmatrix} __ & __ \\ __ & __ \\ __ & __ \\ __ & __ \end{bmatrix}$$

Oblicz macierz transponowaną

$$\begin{bmatrix} 3 & 11 \\ 6 & 4 \\ 23 & 7 \end{bmatrix}^T = \begin{bmatrix} __ & __ & __ \\ __ & __ & __ \end{bmatrix}$$

Nachylenie:

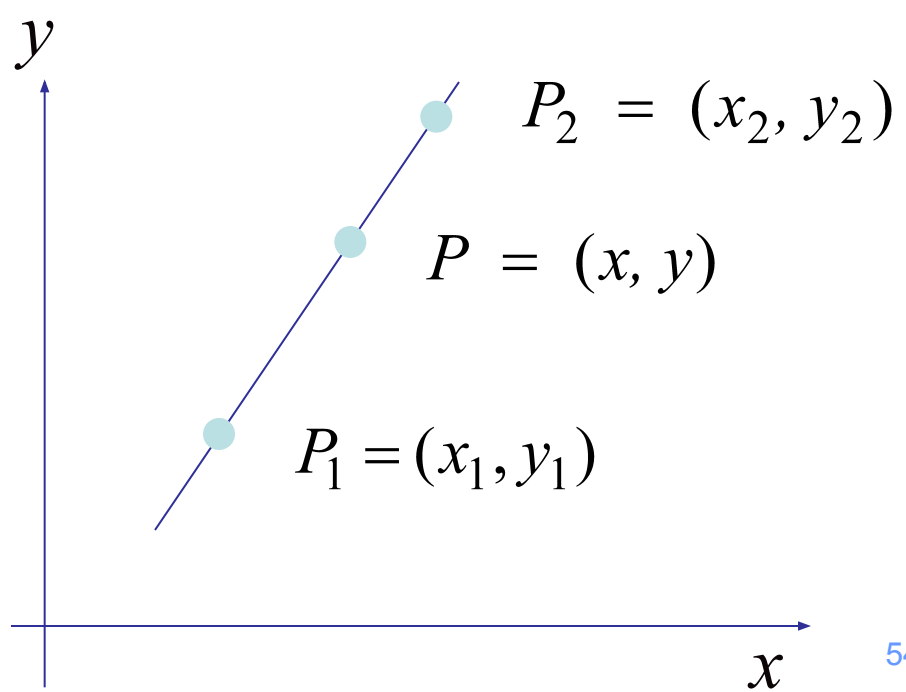
$$m = (y - y_1) / (x - x_1)$$
$$= (y_2 - y_1) / (x_2 - x_1)$$

Rozwiązanie dla y:

$$y = [(y_2 - y_1)/(x_2 - x_1)]x$$
$$+ [-(y_2 - y_1)/(x_2 - x_1)]x_1 + y_1$$

lub:

$$y = mx + b$$

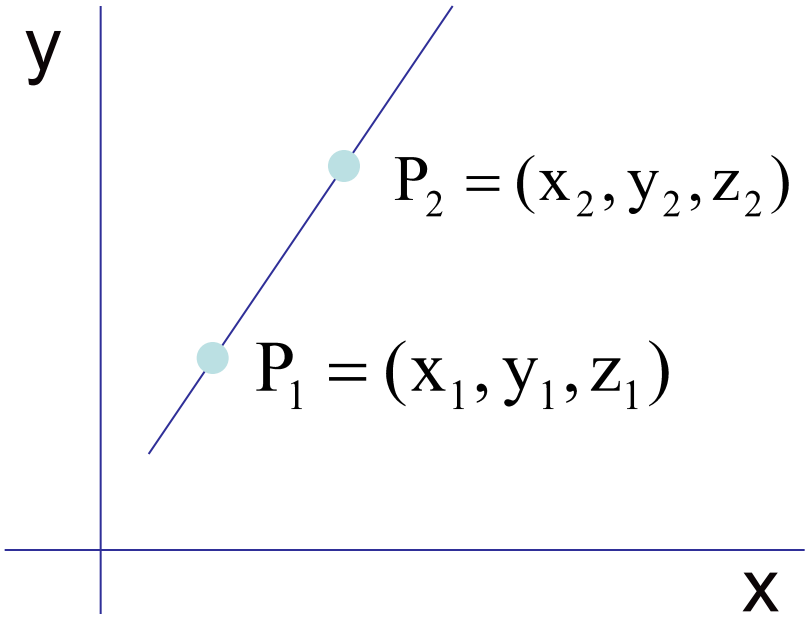


Mamy punkty $P_1 = (x_1, y_1, z_1), \quad P_2 = (x_2, y_2, z_2)$

$$x = x_1 + u(x_2 - x_1)$$
$$y = y_1 + u(y_2 - y_1)$$
$$z = z_1 + u(z_2 - z_1)$$

Kiedy:

- u=0, mamy (x_1, y_1, z_1)
- u=1, mamy (x_2, y_2, z_2)
- (0<u<1), mamy punkty między (x_1, y_1, z_1) i (x_2, y_2, z_2)



Długość = $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

Midpoint $\boldsymbol{P_2}$ between $\boldsymbol{P_1}$ and $\boldsymbol{P_3}$:

$$\boldsymbol{P_2} = \left\{ \frac{x_1 + x_3}{2}, \quad \frac{y_1 + y_3}{2} \right\}$$

2 linie są prostopadłe jeżeli:

$$m_1 = -1 / m_2$$

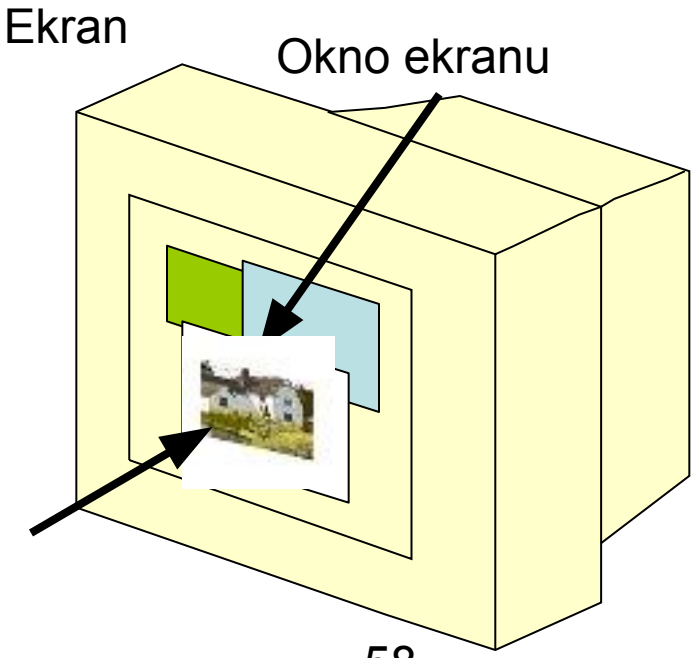
Kosinus kąta między nimi wynosi 0.

Pytania

Są pytania?

Okna i widoki

Świat (to co widzimy, świat rzeczywisty)



Okna i widoki

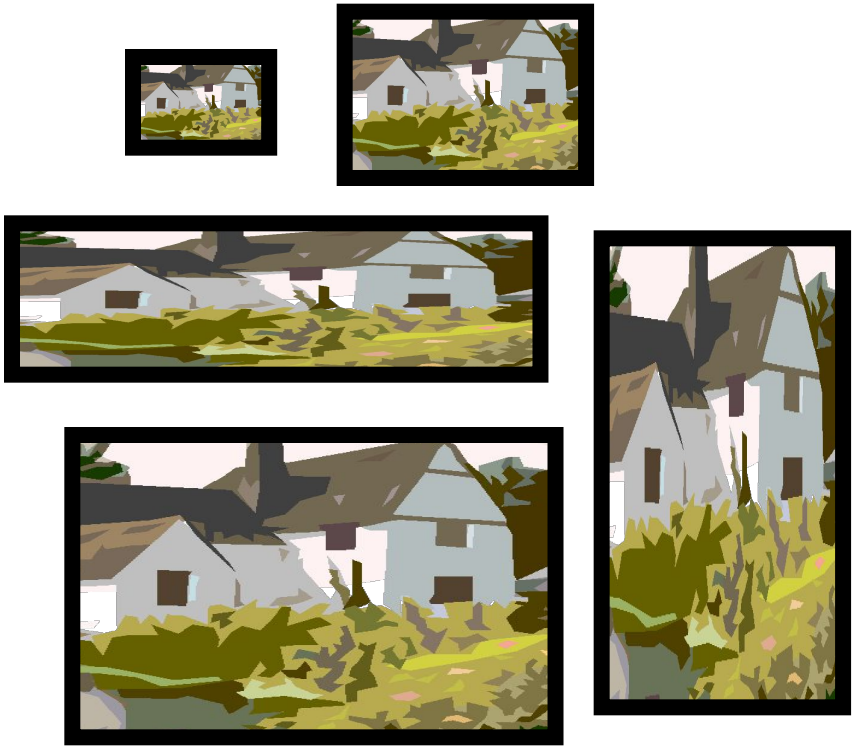
Okno jest prostokątne.

Widok jest prostokątny.

Mogą nie mieć takiego samego rozmiaru lub tzw. aspect ratio.

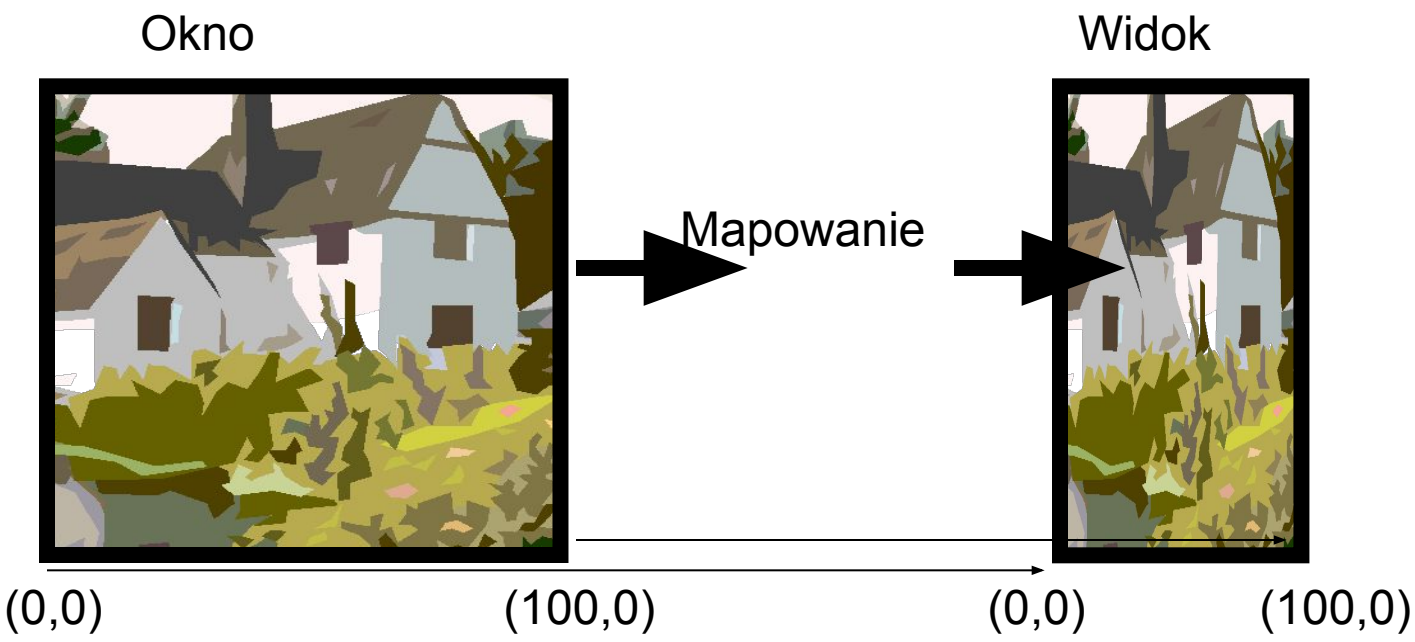
Okna i widoki

Okno



Przykłady widoków

Okna i widoki

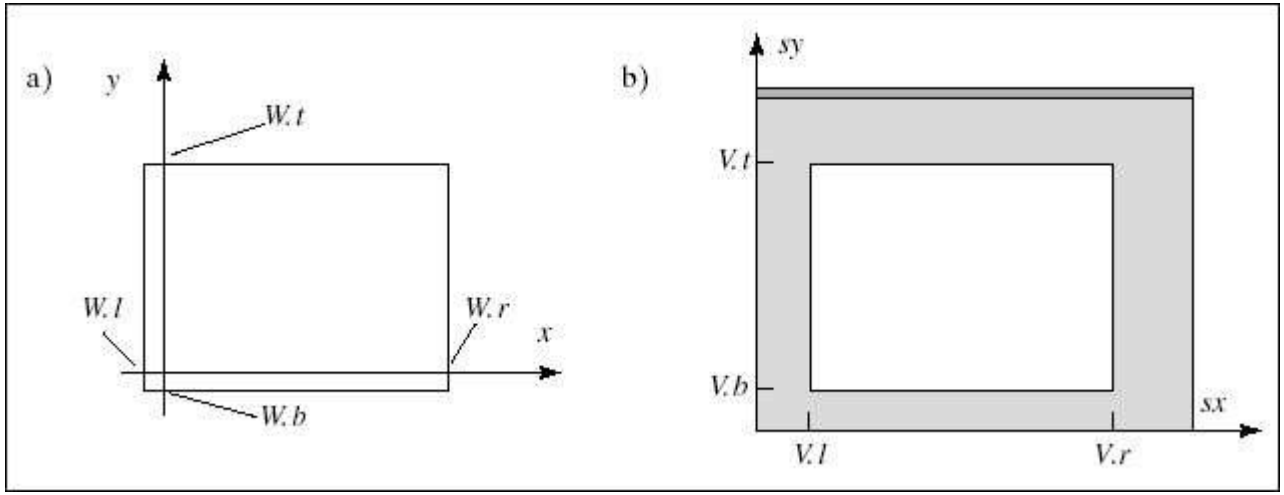


Okna i widoki

$x' = Ax + B$

$y' = Cy + D$

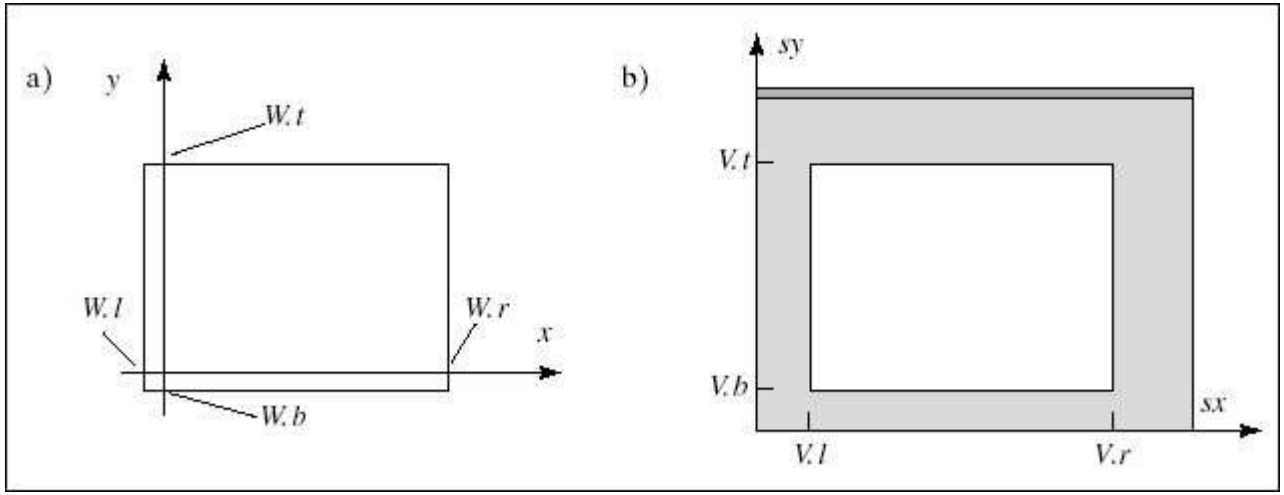
Okna i widoki



W.t = Góra okna
W.l = lewa str. okna
W.b = Dół okna
W.r = prawa str. okna

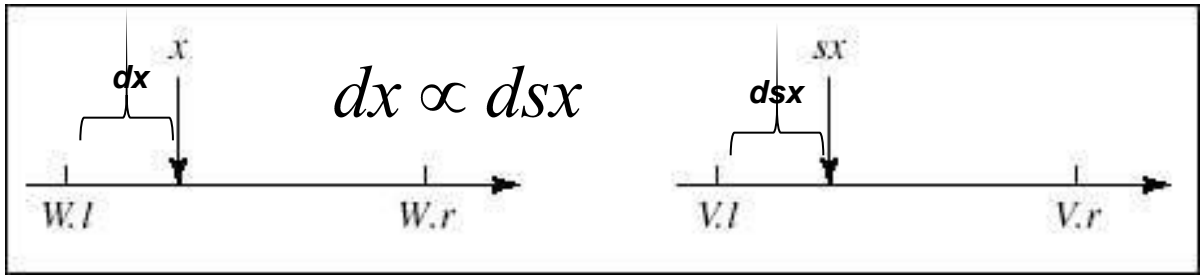
V.t = Góra widoku
V.l = Lewa str. widoku
V.b = Dół widoku
V.r = prawa str. widoku

Okna i widoki



xmin, ymin, xmax, ymax = W.l, W.b, W.r, W.t
SCREENWIDTH = V.r – V.l
SCREENHEIGHT = V.t – V.b

Okna i widoki



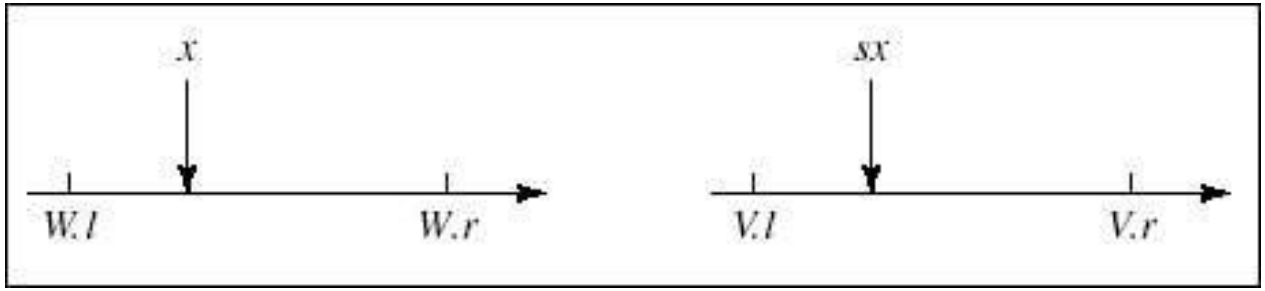
- Jak daleko jest x od W.l?

  dx = x – W.l

  mamy

$$\frac{x - W.l}{W.r - W.l}$$

Okna i widoki



- Wtedy:

$$\frac{sx - V.l}{V.r - V.l} = \frac{x - W.l}{W.r - W.l}$$

- lub

A C

$$sx = \frac{V.r - V.l}{W.r - W.l} x + \left(V.l - \frac{V.r - W.l}{W.r - W.l} W.l \right)$$

Okna i widoki

- Dla y:

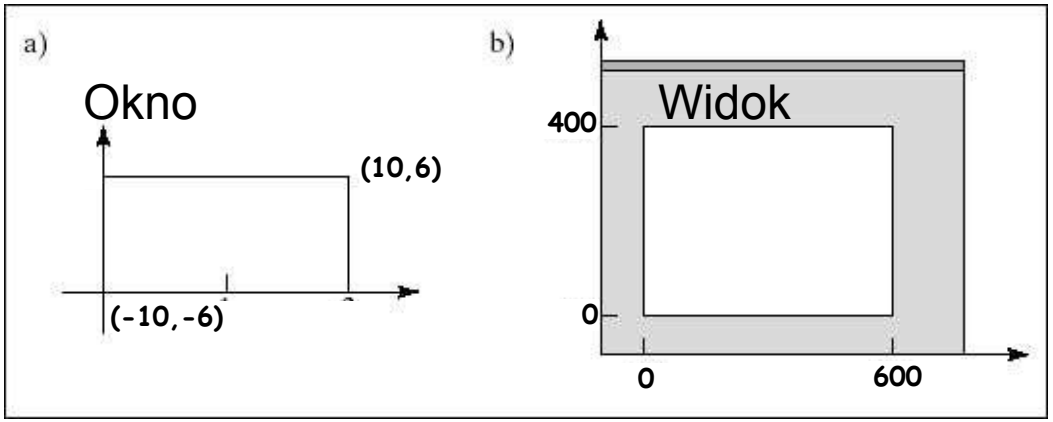
$$\frac{sy - V.b}{V.t - V.b} = \frac{x - W.b}{W.t - W.b}$$

- lub

$$sy = \overbrace{\frac{V.t - V.b}{W.t - W.b}}^{B} y + \overbrace{\left(V.b - \frac{V.t - W.b}{W.t - W.b} W.b \right)}^{D}$$

Okna i widoki

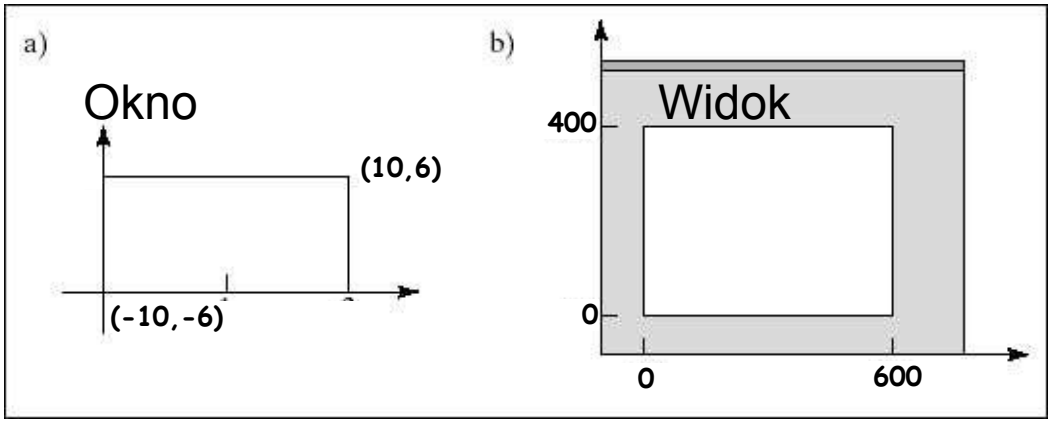
Np.



Jakie są A, B, C oraz D ?

Okna i widoki

Np.



A = 600/20 = 30

B = 400/12 = 33.3333

C = 0 – 30 * -10 = 300

D = 0 – 33.3333 * -6 = 200

Jeśli to jest poprawne,
(-10,-6) to (0,0) i (10,6) -> (600,400)
sx = A * -10 + C = 0 ☺
sy = B * -6 + D = 0 ☺
sx = A * 10 + C = 600 ☺
xy = B * 6 + D = 400 ☺

Widok

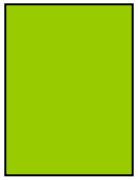
Mamy

$$\frac{W_{window}}{H_{window}} = \frac{W_{viewport}}{H_{viewport}}$$

Widok

Jeżeli aspect ratio prostokąta jest < 0, prostokąt jest

Np. W/H = 3/5



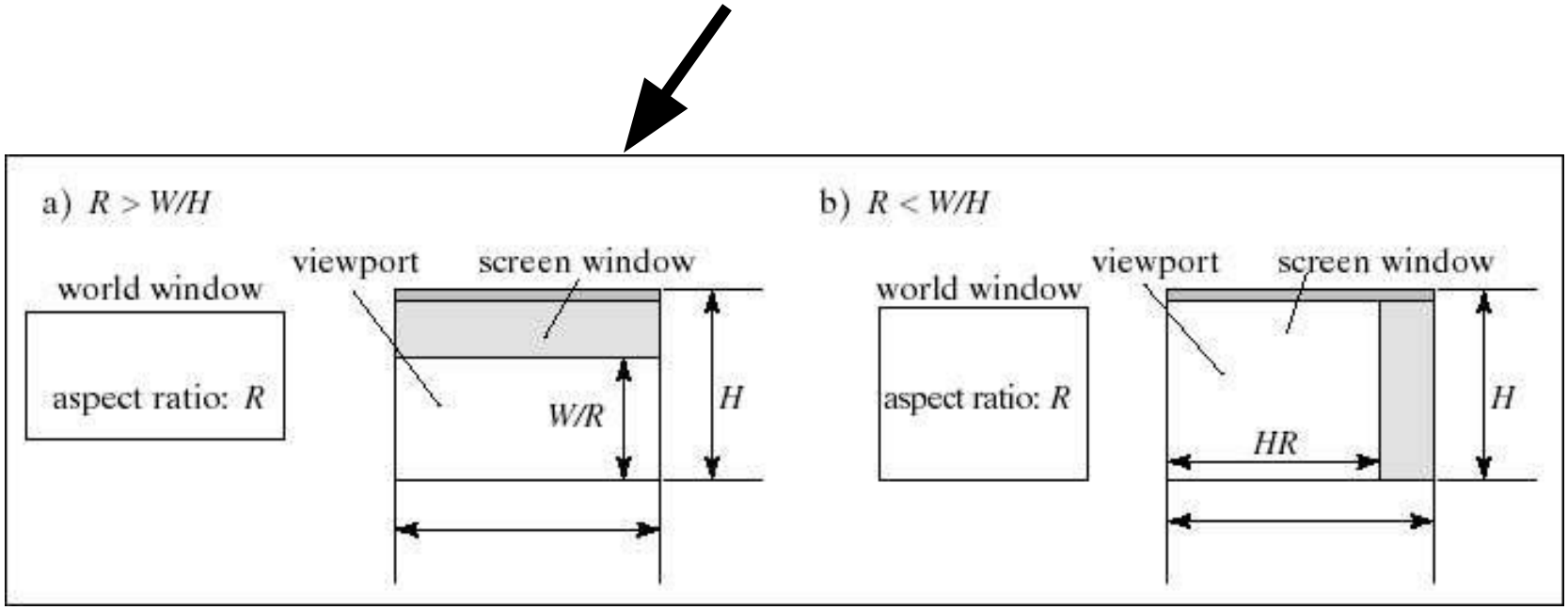
Jeżeli aspect ratio jest > 0, the mamy

Np. W/H = 5/3

Widok

a) R > W/H (R jest aspect ratio okna rzeczywistego)

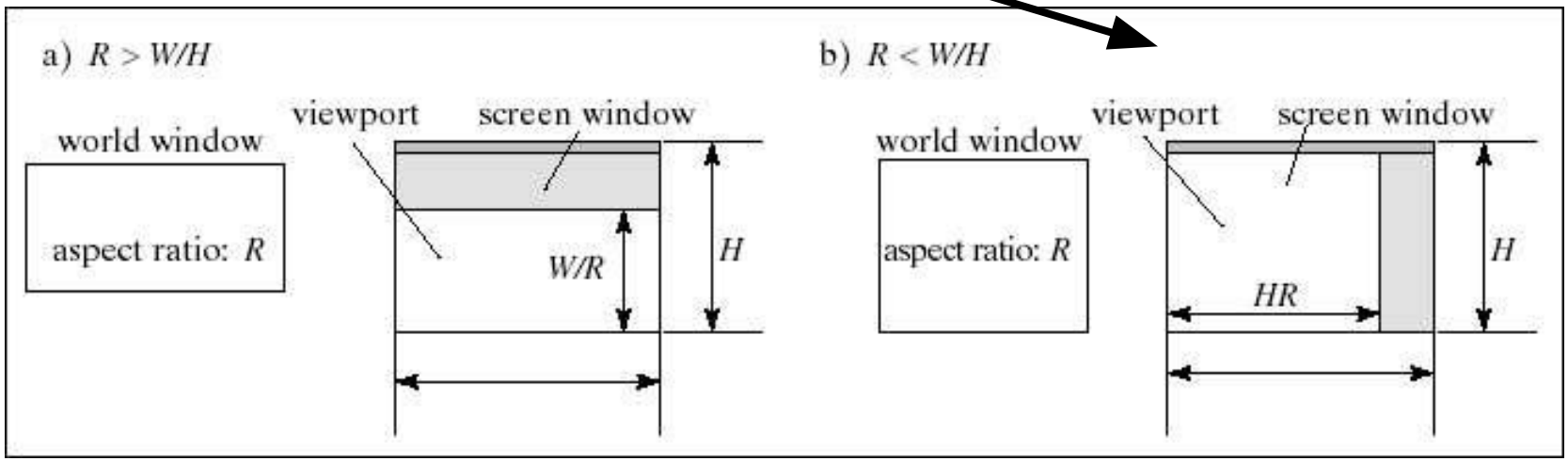
- setViewport(0,W,0,W/R);



Widok

b) R < W/H (R aspect ratio okna)

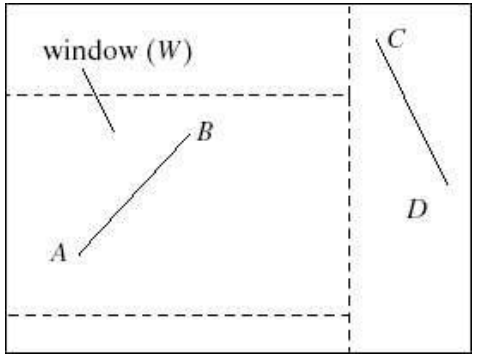
setViewport(0,H*R,0,H);



Obcinanie linii

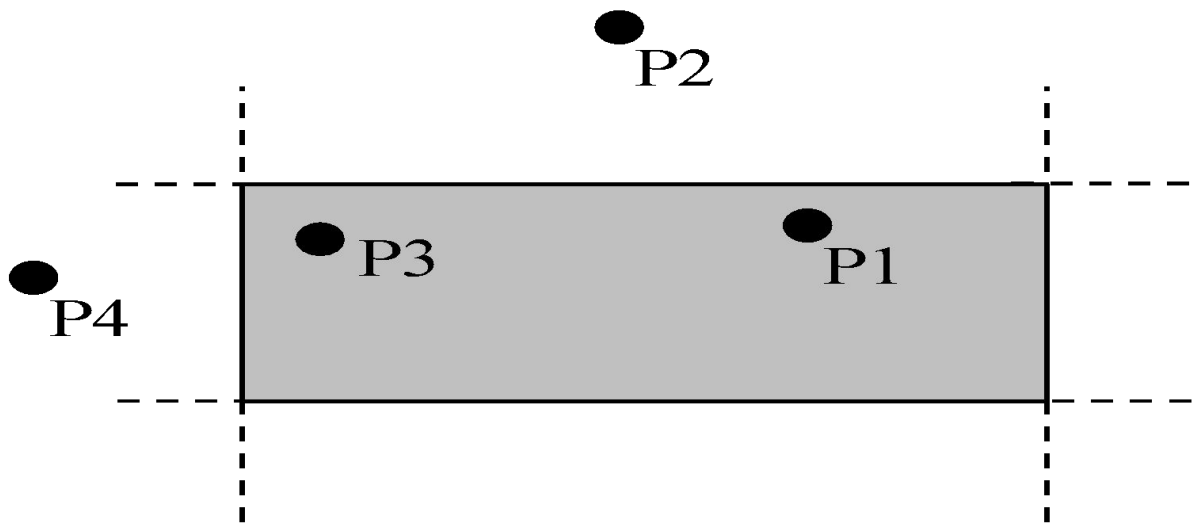
Algorytm Cohena-Sutherlanda

1. Przypadek trywialny Akceptacja lub Odrzucenie
   - Jeśli oba końce linii leżą wewnątrz okna (np. AB) wtedy linia jest trywialnie akceptowana i nie jest obcinana.
   - Jeśli oba końce linii leżą na zewnątrz okna (np. CD) wtedy linia jest trywialnie odrzucana.

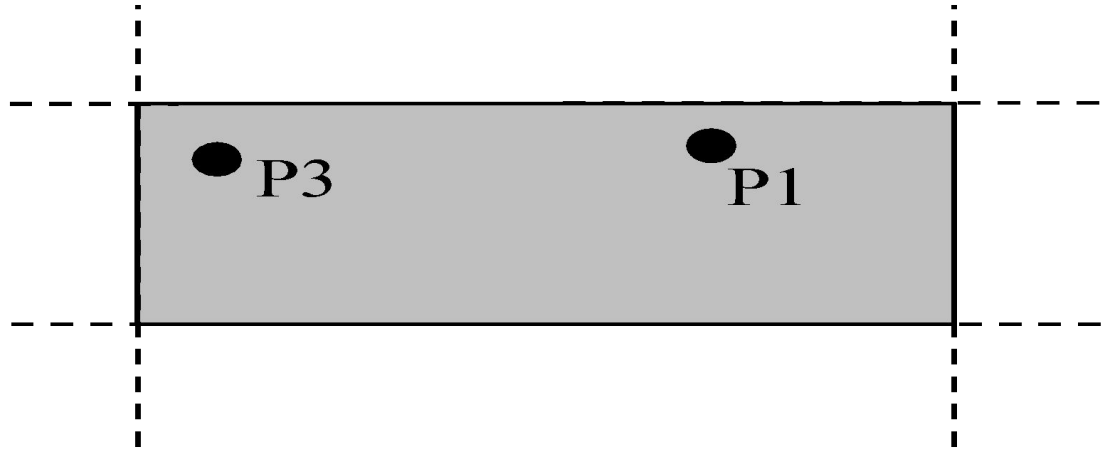


Przed i po obcinaniu PUNKTU

Przed



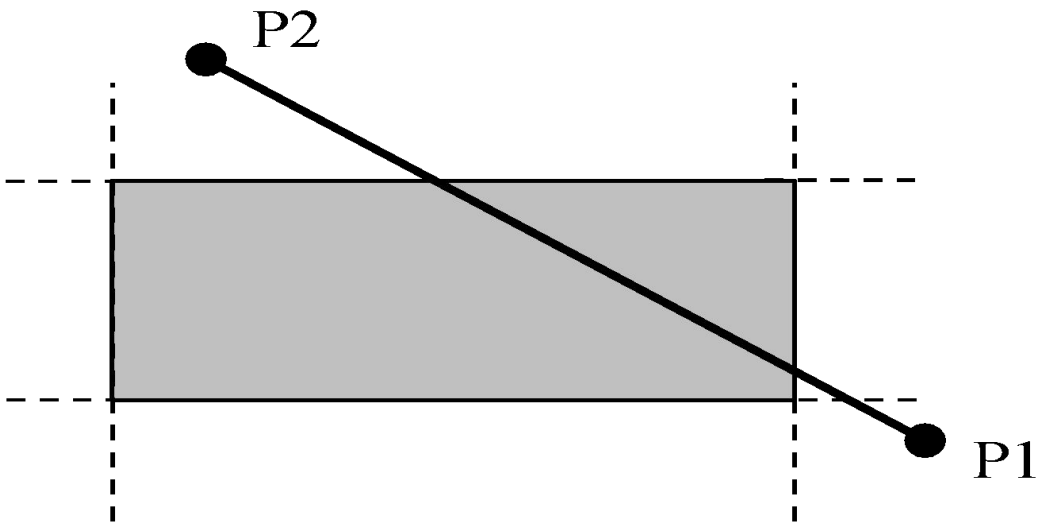
Po



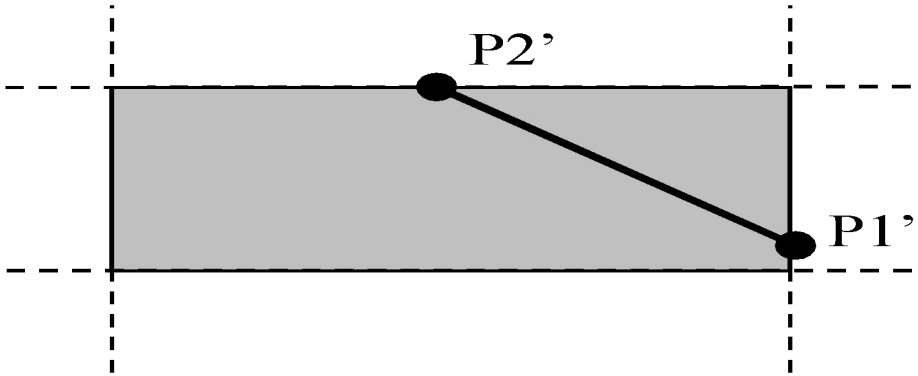
*

Przed i po obcinaniu LINII

Przed



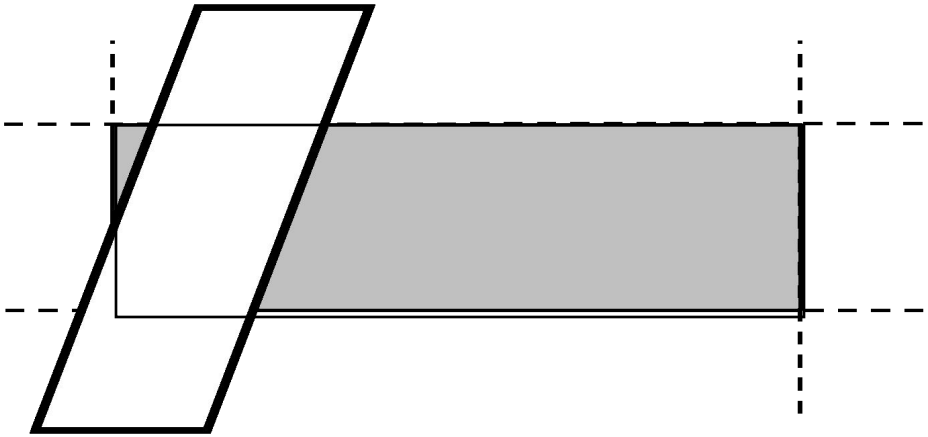
Po



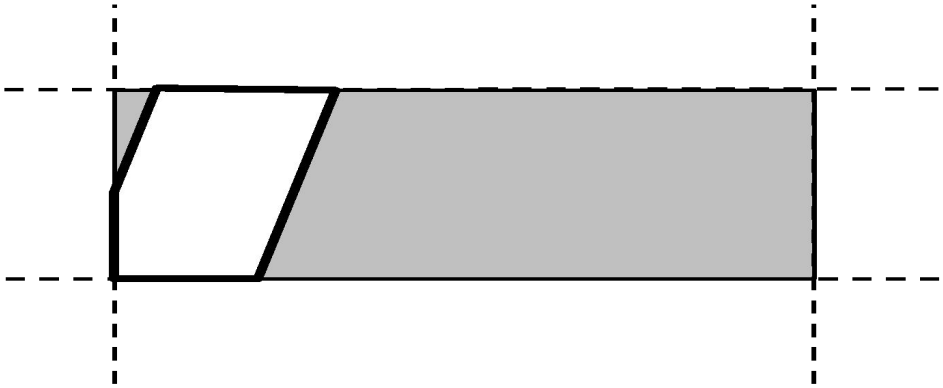
*

Przed i po obcinaniu WIELOKĄTA

Przed



Po

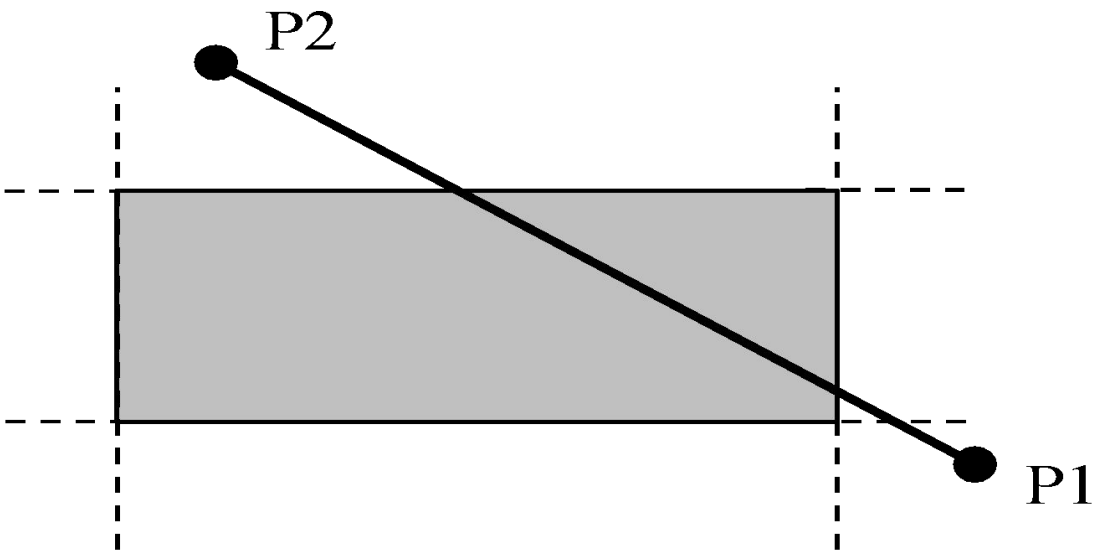


*

Algorytm 2D obcinania Cohena-Sutherlanda

4 obszary są zdefiniowane – na zewnątrz okna

- GÓRA
- DÓŁ
- LEWA
- PRAWA



*

4-bitowy kod

Przypisujemy 4-bitowy kod każdemu punktowi

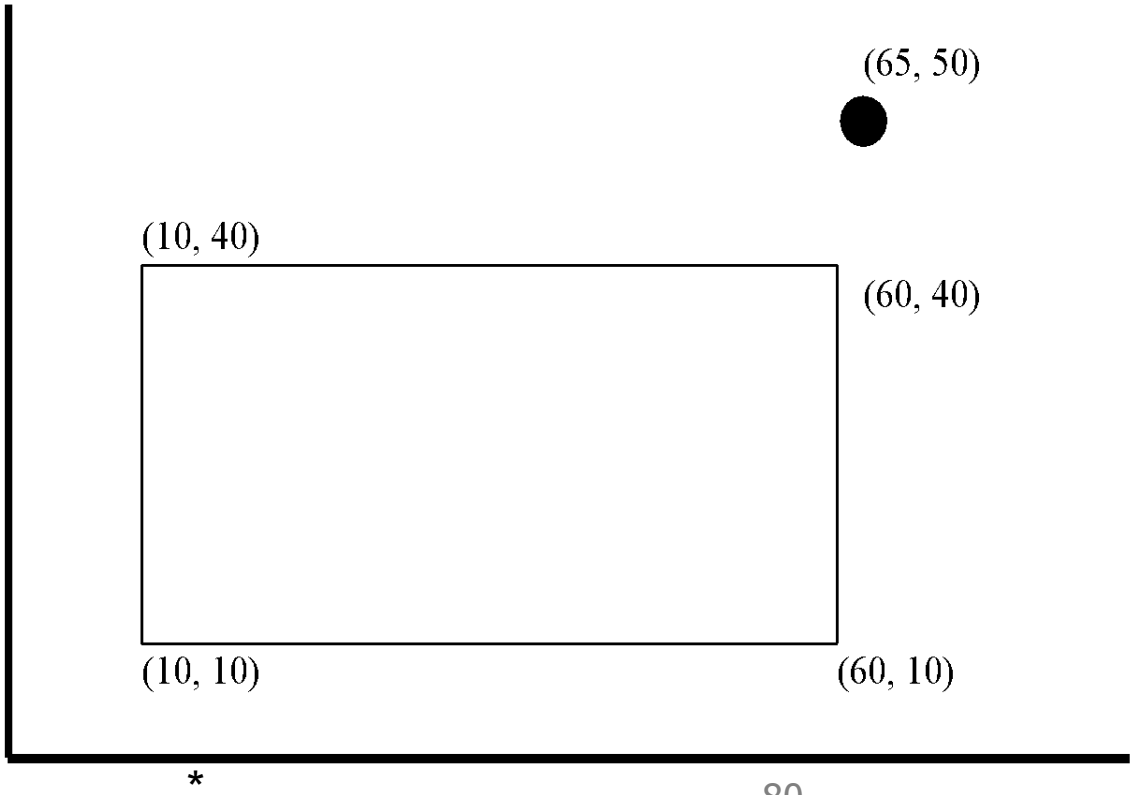
Kod nazywamy obszarowym kodem Cohena-Sutherlanda

```
LEWY   bit 1 = binary 0001
PRAWY  bit 2 = binary 0010
DÓŁ    bit 3 = binary 0100
GÓRA   bit 4 = binary 1000
```

*

Punkt (65, 50) – góra – prawo względem okna,

Kod: 1010



1000 pokazuje obszar GÓRA

0010 pokazuje obszar PRAWO

4 bitowy kod Cohena-Sutherlanda tworzymy przez logiczną operację OR wszystkich obszarów

- 1000 GÓRA
- OR0010 PRAWO
- ----------
- = 1010

*

Decyzje

- Jeżeli kod (0000) punkt wewnątrz okna
- Jeżeli kod jest nie-zerwy punkt na zewnątrz okna

Dla dwóch punktów końcowych linii

- Jeżeli kody obu punktów końcowych są zerowe, cała linia wewnątrz okna
- Jeżeli (logiczna) AND kodów końcowych jest nie-zerowa, punkty końcowe muszą mieć wspólny obszar…
  Cała linia musi być na zewnątrz okna

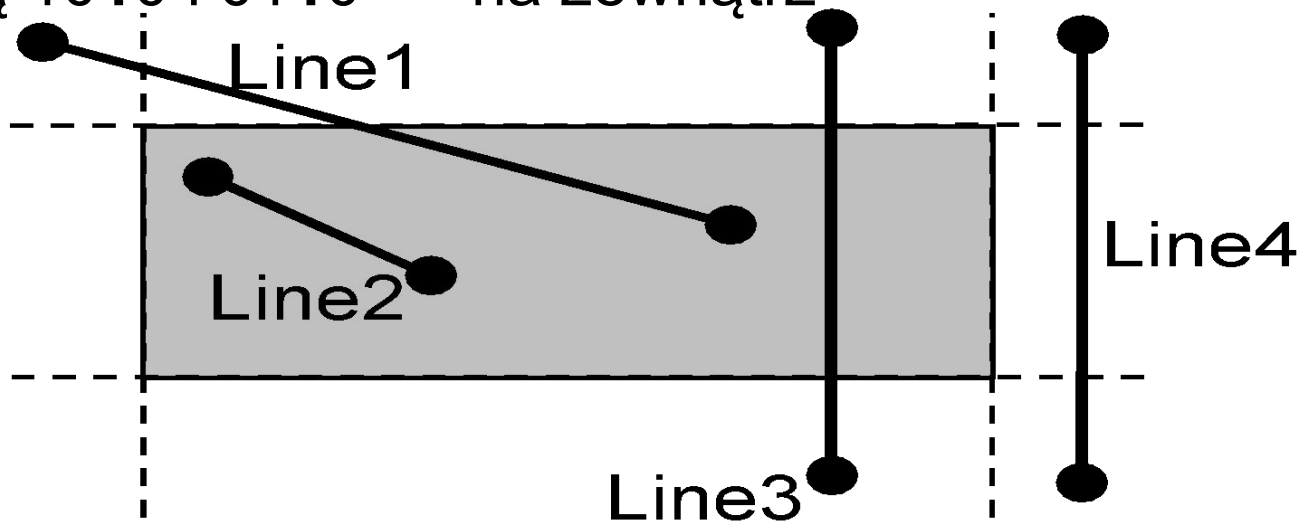
GÓRA DÓŁ PRAWO LEWO

Linia 1 kody są 1001 i 0000

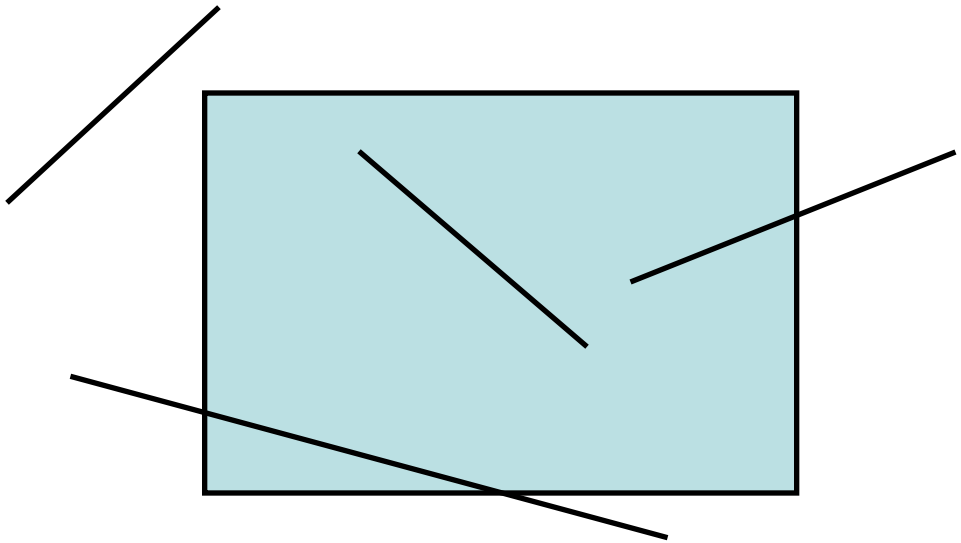
Linia 2 kody są 0000 i 0000 << wewnątrz

Linia 3 kody są 1000 i 0100

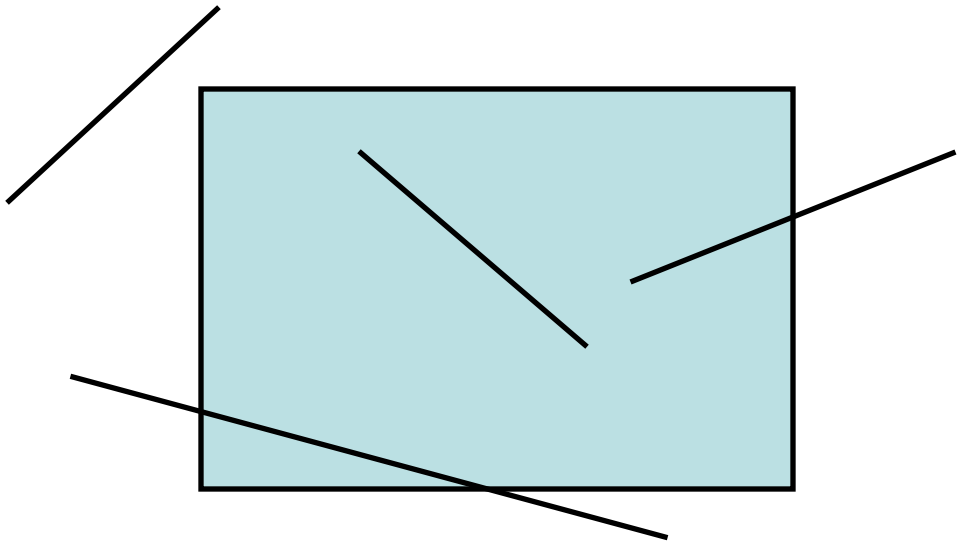
Linia 4 kody są 10**1**0 i 01**1**0 << na zewnątrz



*



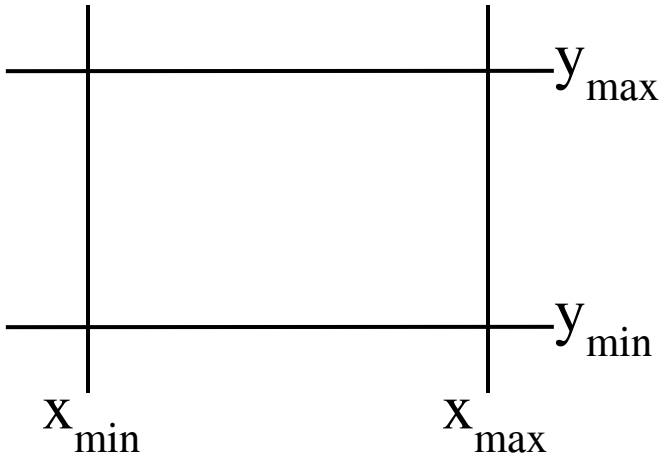
*



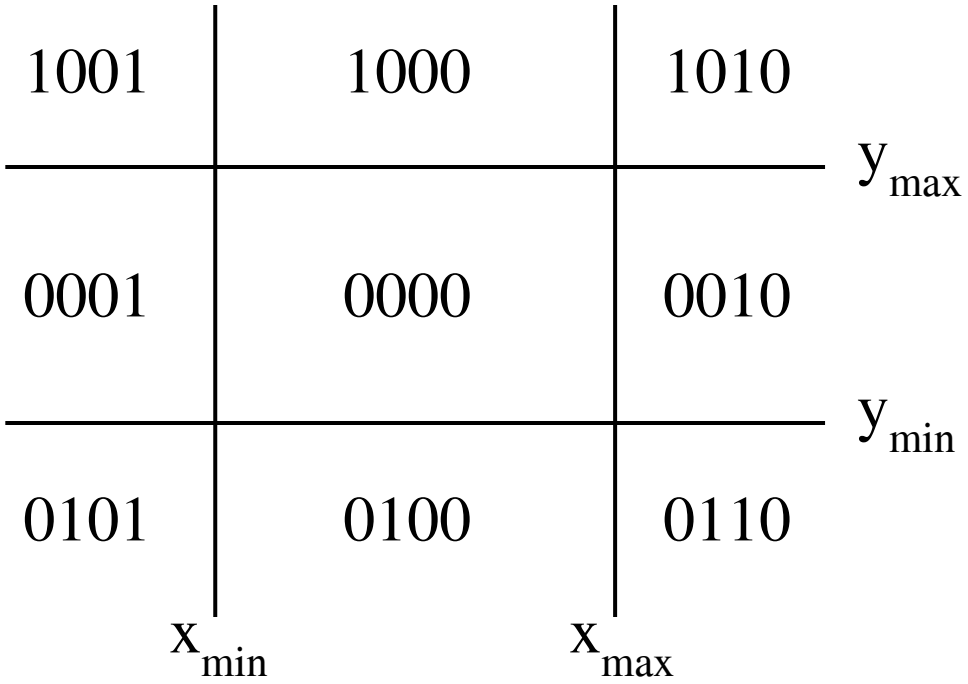
*

$$x_{\min} \le x \le x_{\max}$$

$$y_{\min} \le y \le y_{\max}$$



*


1001
1000
1010
y_{max}
0001
0000
0010
y_{min}
0101
0100
0110
x_{min}
x_{max}

*

1001	1000	1010
0001	0000	0010
0101	0100	0110

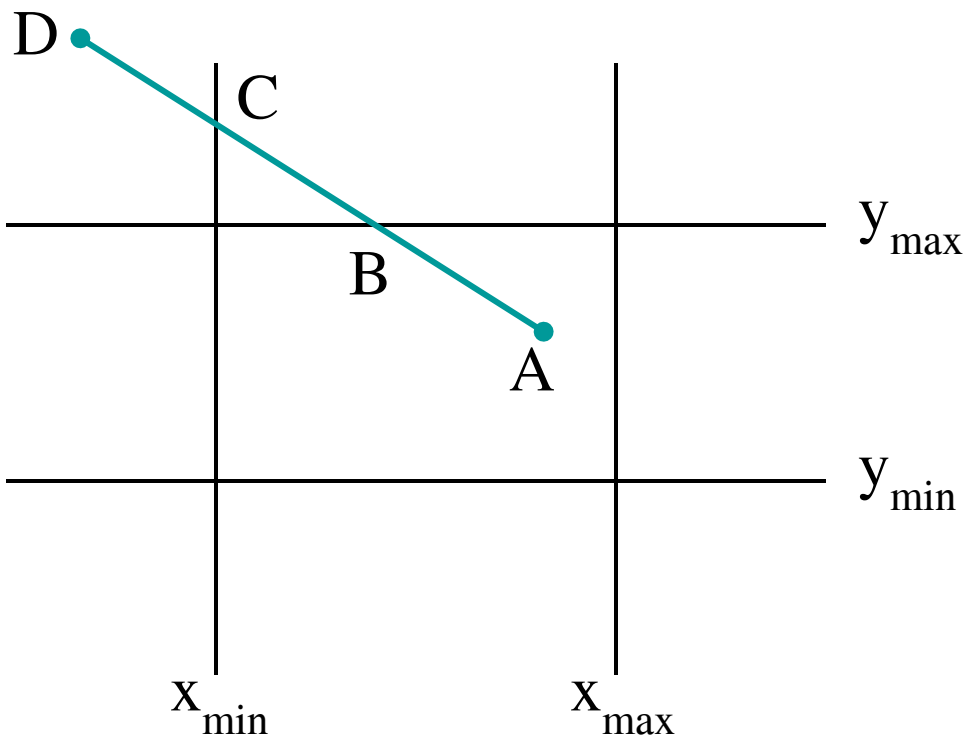
y_{max}

y_{min}

x_{min} x_{max}

bit	condition
1	$y > y_{max}$
2	$y < y_{min}$
3	$x > x_{max}$
4	$x < x_{min}$

*



Kod(A) = 0000

Kod(D) = 1001

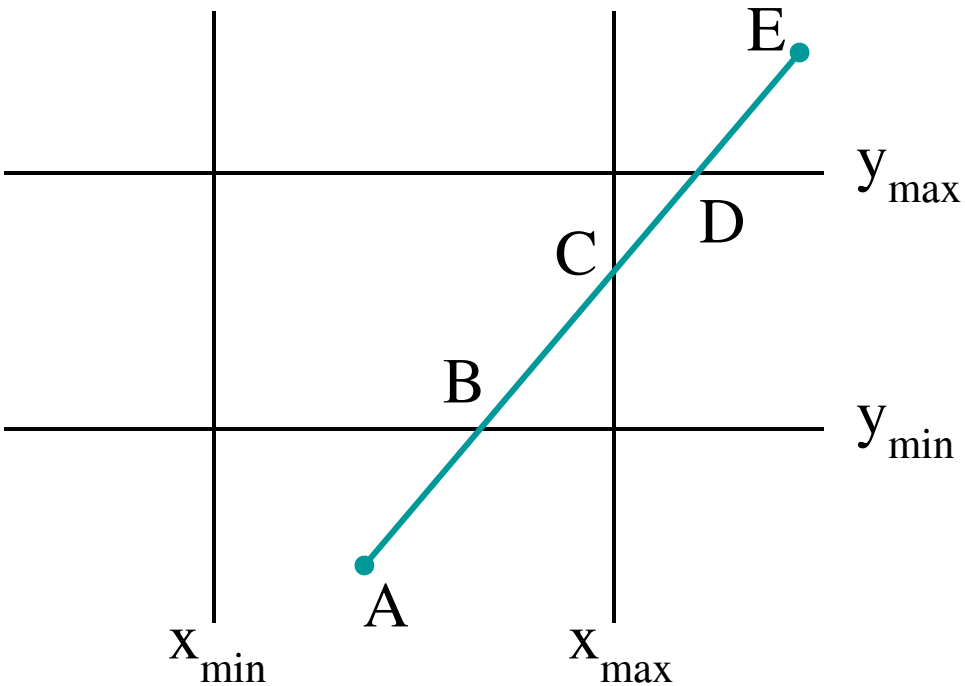
Nie trywialny przypadek akceptacja/odrzucenia

Obcinamy (A,D) przy $y = y_{max}$,

*

Kod(A) = 0100

Kod(E) = 1010



obcinamy (A,E) przy $y = y_{max}$,

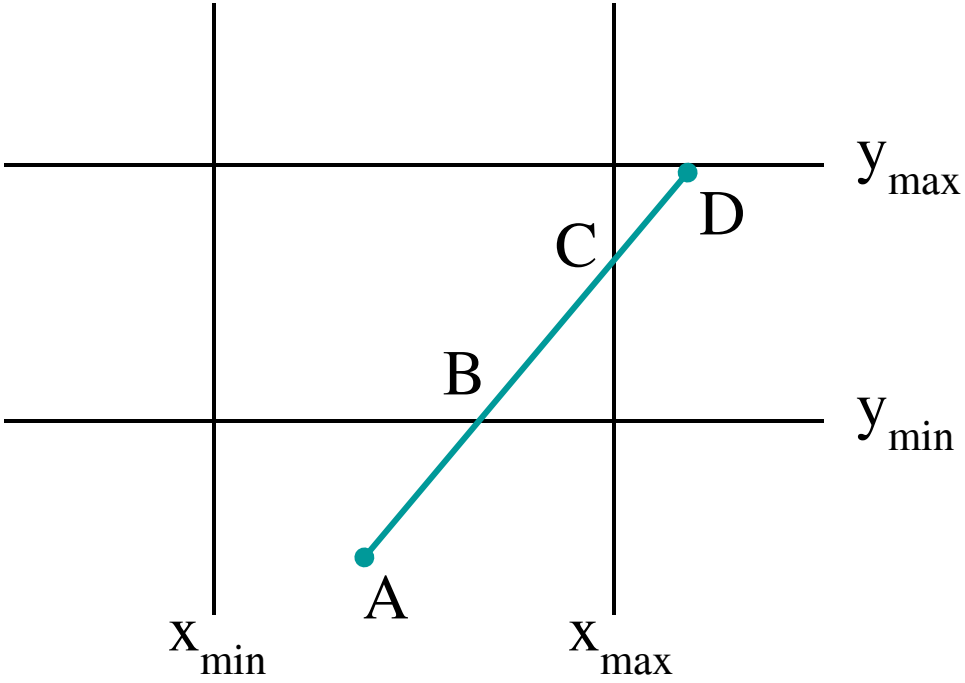
Odrzucamy (D,E)

Przetwarzamy (A,D)

*

Kod(A) = 0100

Kod(D) = 0010



Obcinamy (A,D) przy $y = y_{min}$,

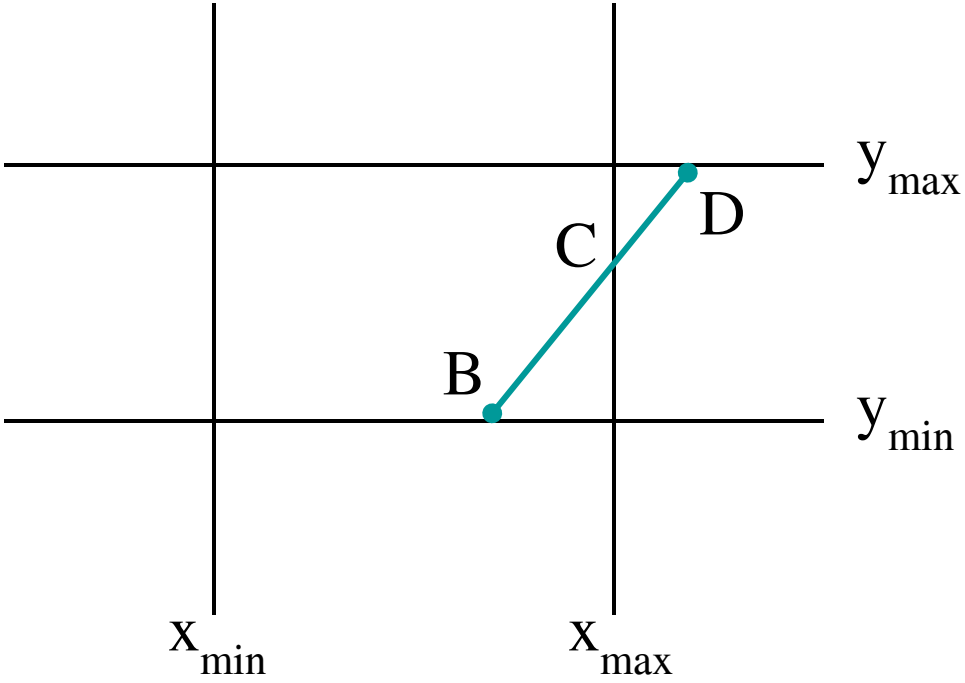
Odrzucamy (A,B)

Przetwarzamy (B,D)

*

Kod(B) = 0000

Kod(D) = 0010



Obcinamy (B,D) przy $x = x_{max}$,

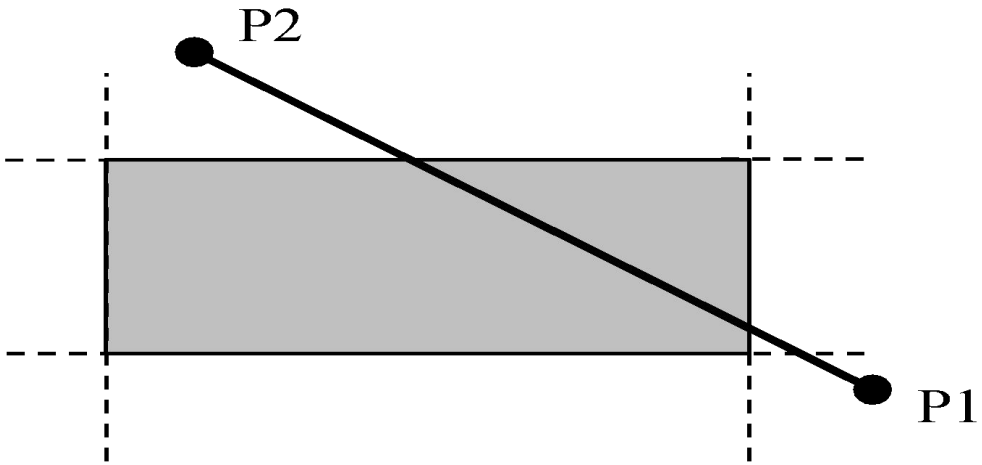
Odrzucamy (C,D)

Przetwarzamy (B,C)

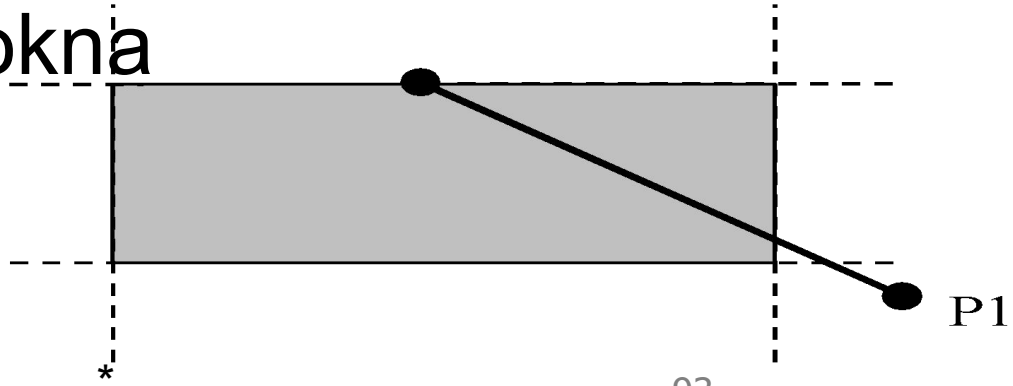
*

Obcinamy linię

Punkt P2 (kod 1000)

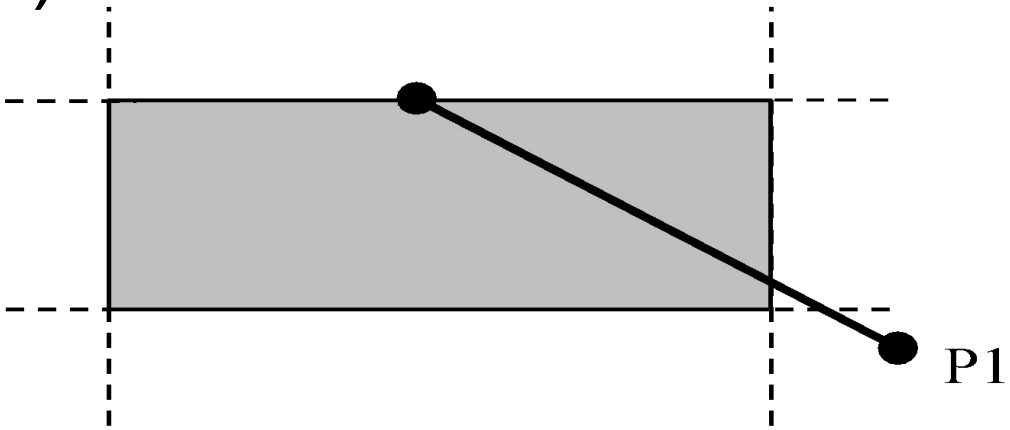


Obcinamy GÓRA okna



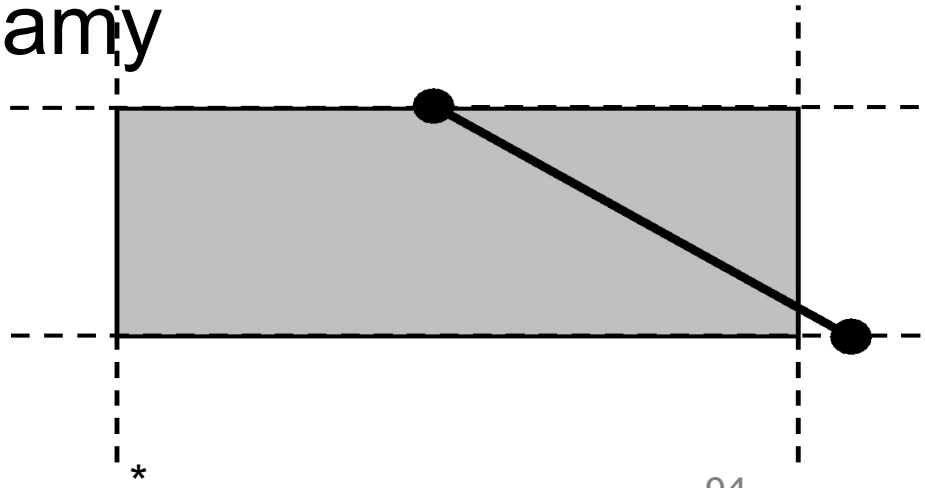
Clipping a line

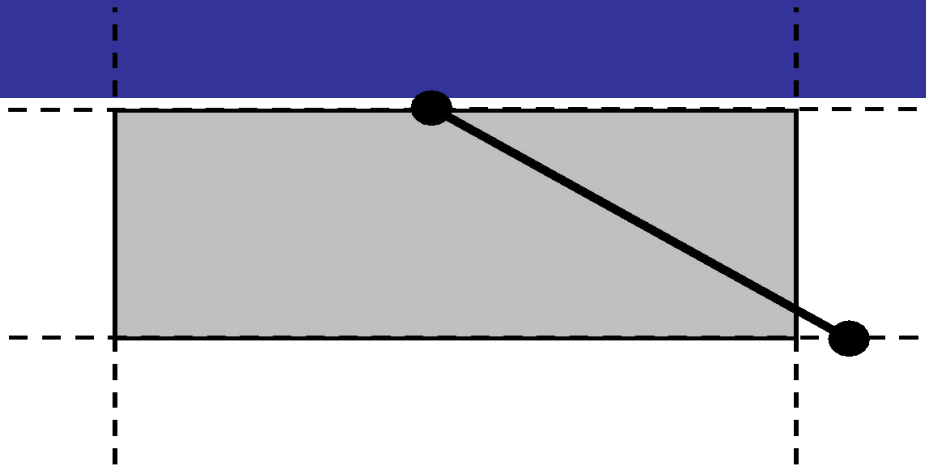
Punkt P1 (kod 0110)



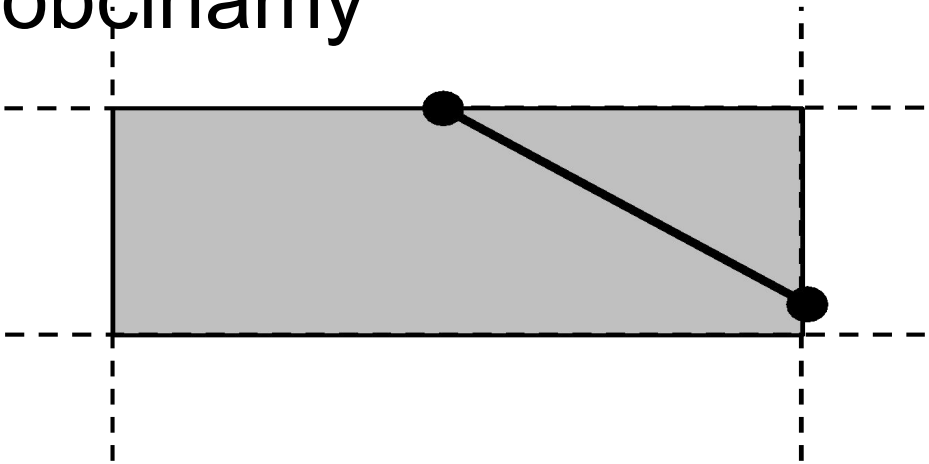
Bit 1 (GÓRA) nie

Bit 2 (DÓŁ) obcinamy





Bit 3 (PRAWO) nie obcinamy

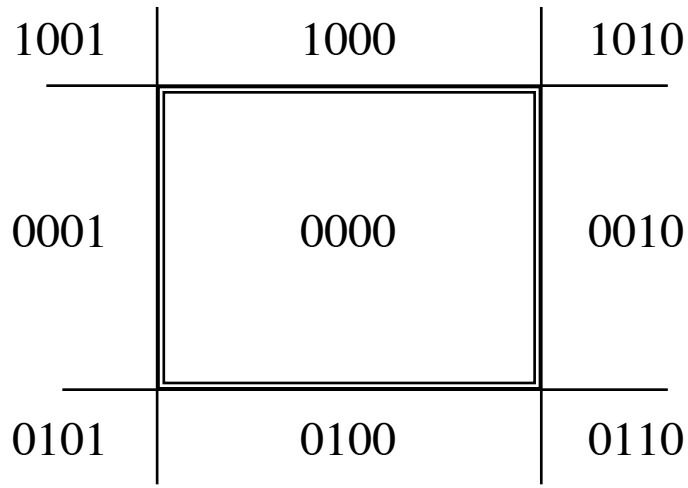


Bit 4 (LEWO) obcinanie zakończone

*

Obcinamy linię

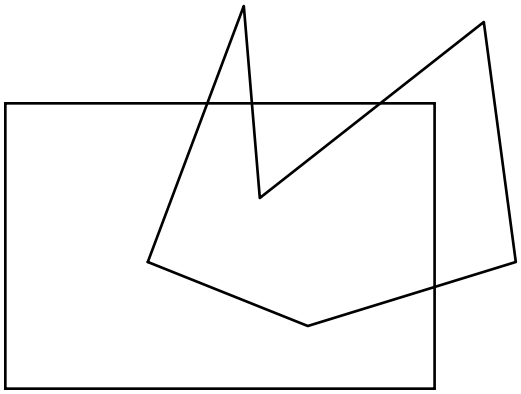
Bit 1: Góra
Bit 2: Dół
Bit 3: Prawo
Bit 4: Lewo

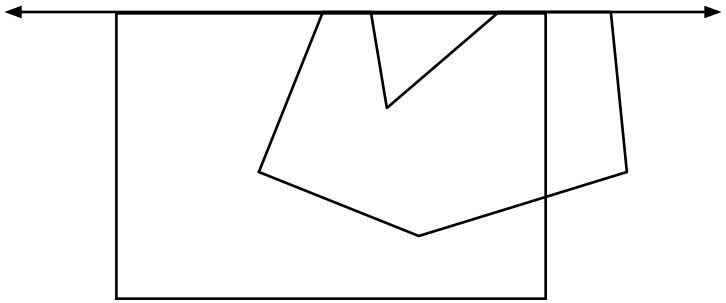


- Bit 1: 1 jeżeli $y > y_t$, wpp 0
- Bit 2: 1 jeżeli $y < y_b$, wpp 0
- Bit 3: 1 jeżeli $x > x_r$, wpp 0
- Bit 4: 1 jeżeli $x < x_l$, wpp 0

*

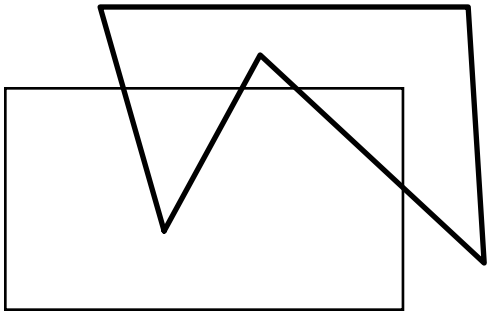
Obcinanie wielokata

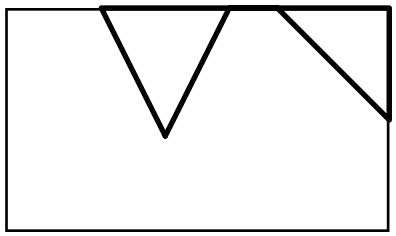




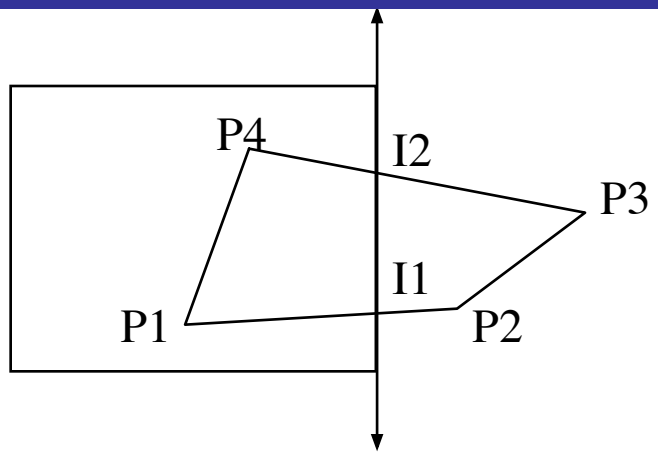
*

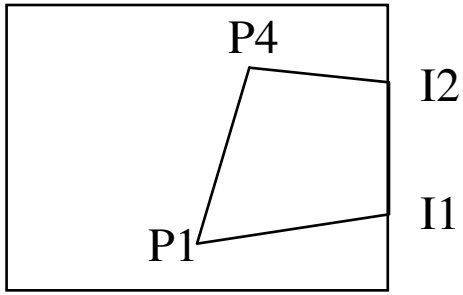
Obcinanie wielokata



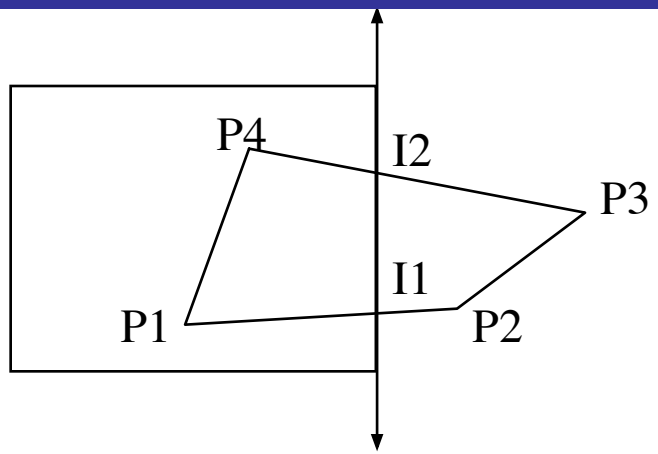


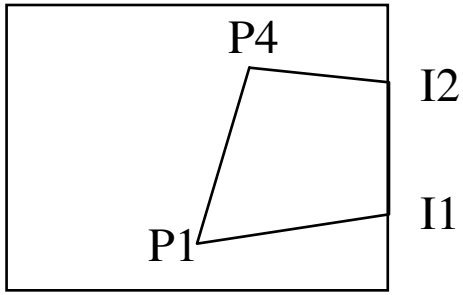
*


P4
I2
P3
I1
P1
P2

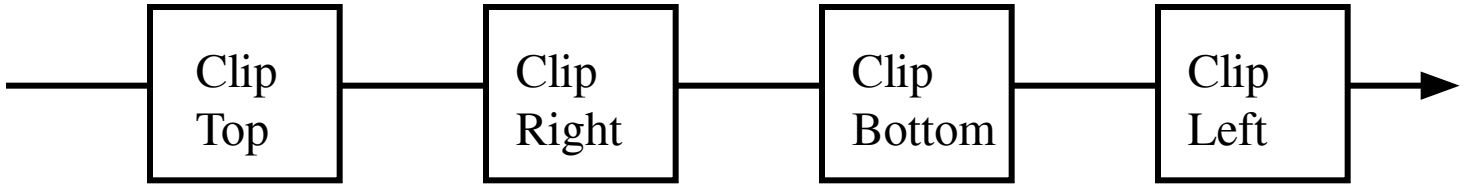

P4
I2
I1
P1

*


P4
I2
P3
I1
P1
P2


P4
I2
I1
P1

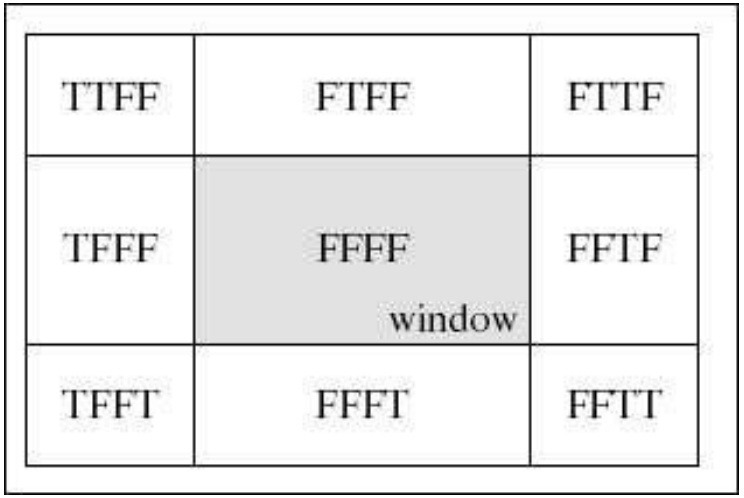
*


Clip
Top
Clip
Right
Clip
Bottom
Clip
Left

*

Obcinanie linii

Algorytm Cohena-Sutherlanda



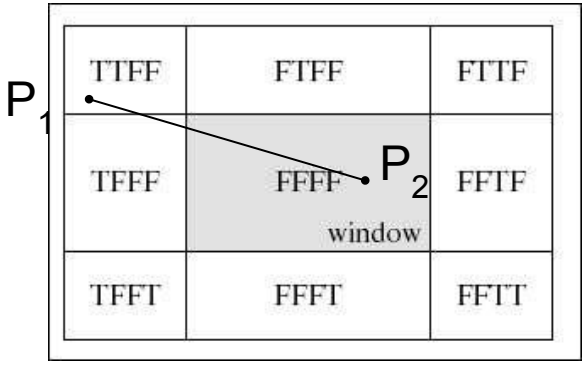
Obcinanie linii

Algorytm Cohena-Sutherlanda

1. Przydział kodu dla końcowych punktów linii.

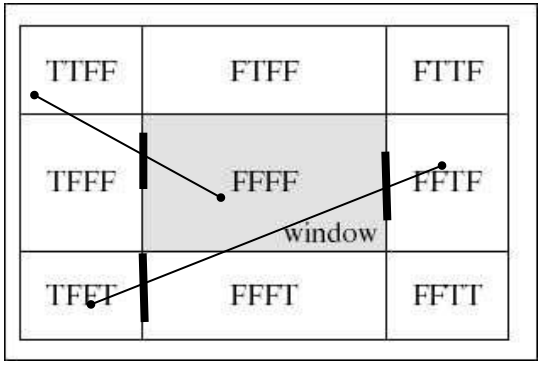
   P_1 = TTFF, P_2 = FFFF

- Trywialna Akceptacja jeśli oba punkty mają kod FFFF
- Trywialne Odrzucenie jeśli oba punkty mają T w takich samych pozycjach np. oba punkty są na lewo, prawo, ponad lub poniżej okna.



Obcinanie linii

Algorytm Cohena-Sutherlanda



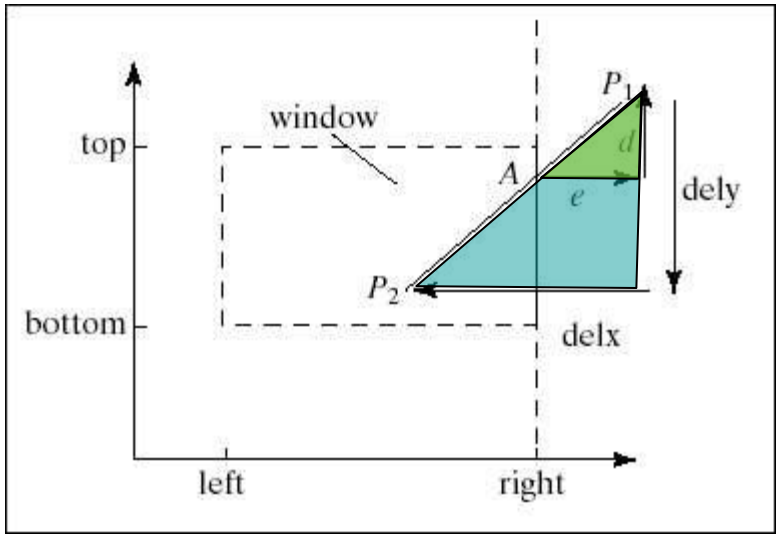
Obcinanie linii

Algorytm Cohena-Sutherlanda

Musimy określić położenie A.

- współrzędna x jest okno.prawa
- to współrzędna y może być obliczona z podobieństwa trójkątów

$$\frac{d}{dely} = \frac{e}{delx}$$



Obcinanie linii

Algorytm Cohena-Sutherlanda

Do określenia A mamy.

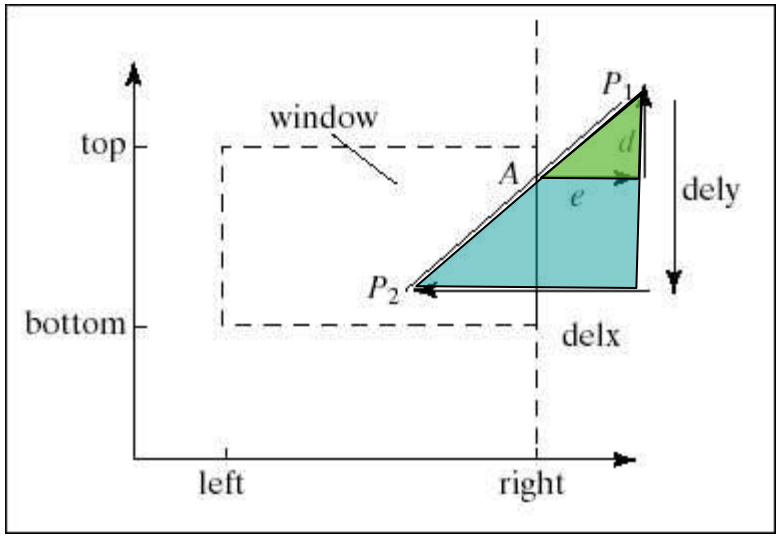
e = p1.x – W.right

delx = p2.x – p1.x;

dely = p2.y – p1.y;

d = e/delx * dely;

p1.y += (W.right – p1.x) * dely/delx

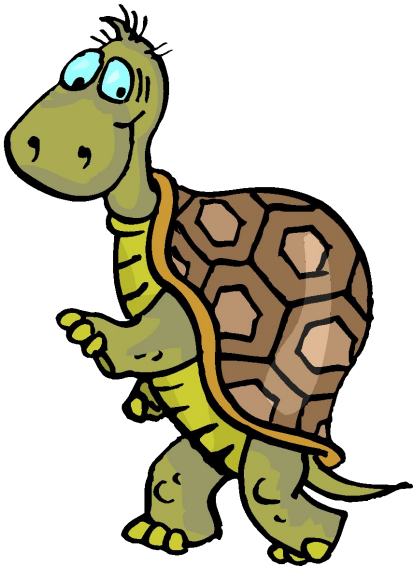


Grafika Żółwia

Rysujemy linie i punkty na ekranie.

Polecenia (pióro w ploterze).

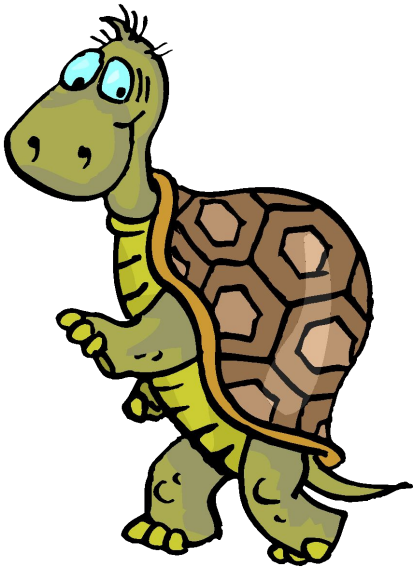
moveto(), forward(), turn()



Grafika żółwia

Np.: Polyspirals

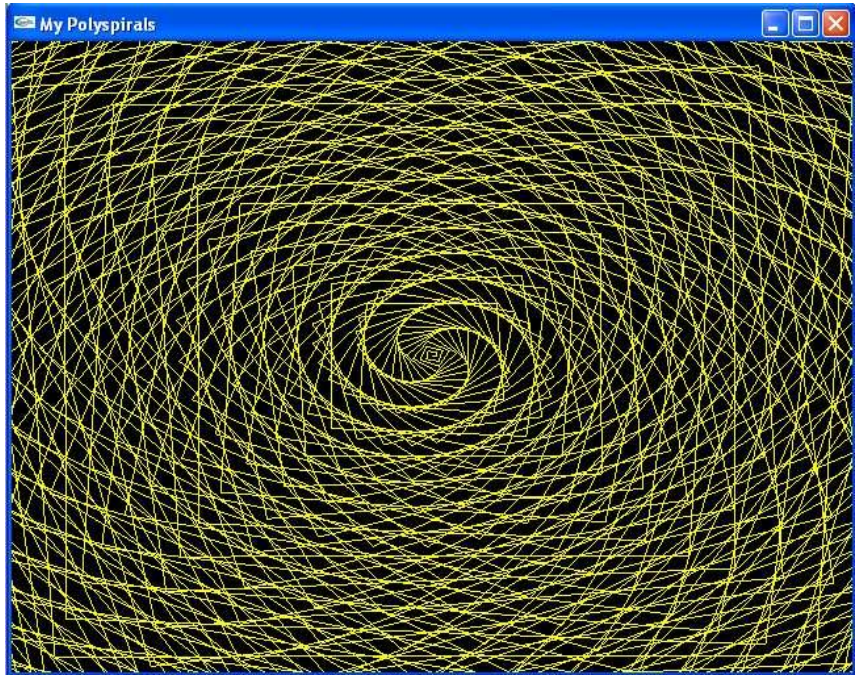
```
for( some iterations )
{
      forward(length, 1);
      turn(angle);
      length += increment;
}

- polyspiral.cpp
```

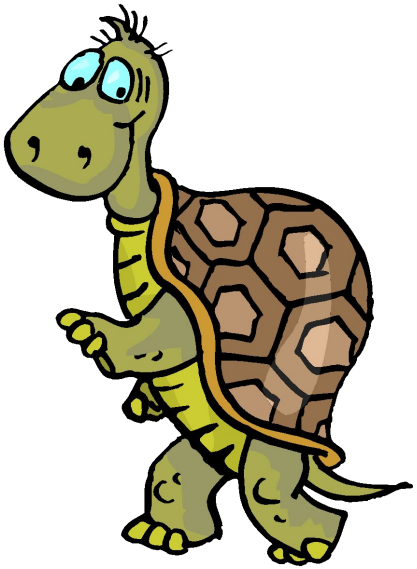


Grafika żółwia

Np.: Polyspirals

float angle = 87, inc = 0.5;

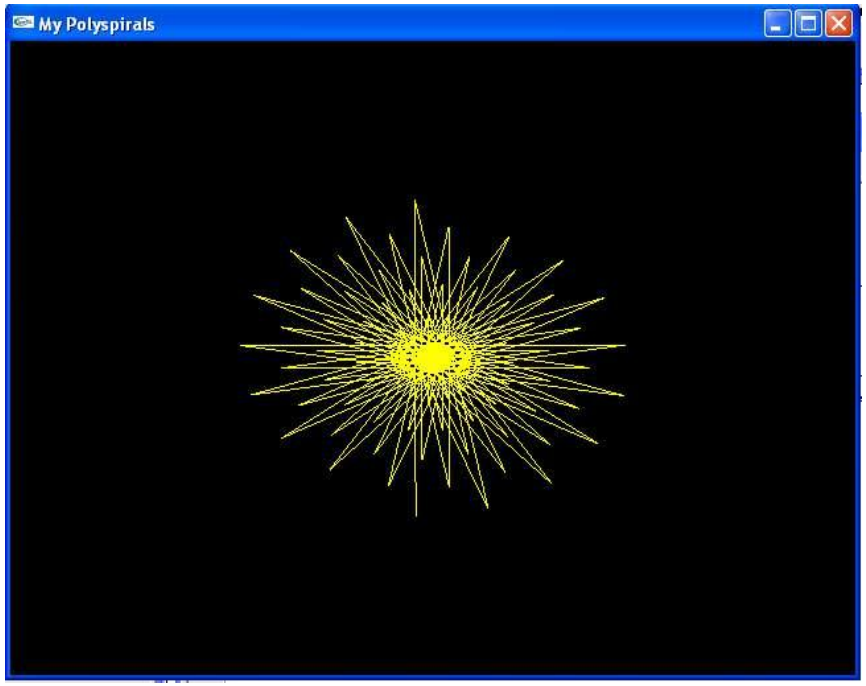


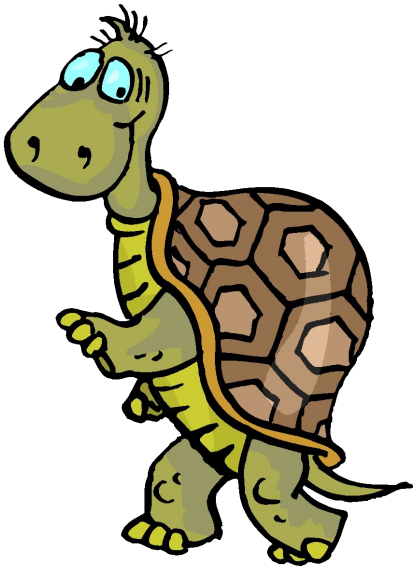


Grafika żółwia

Np.: Polyspirals

float angle = 170, inc = 1;

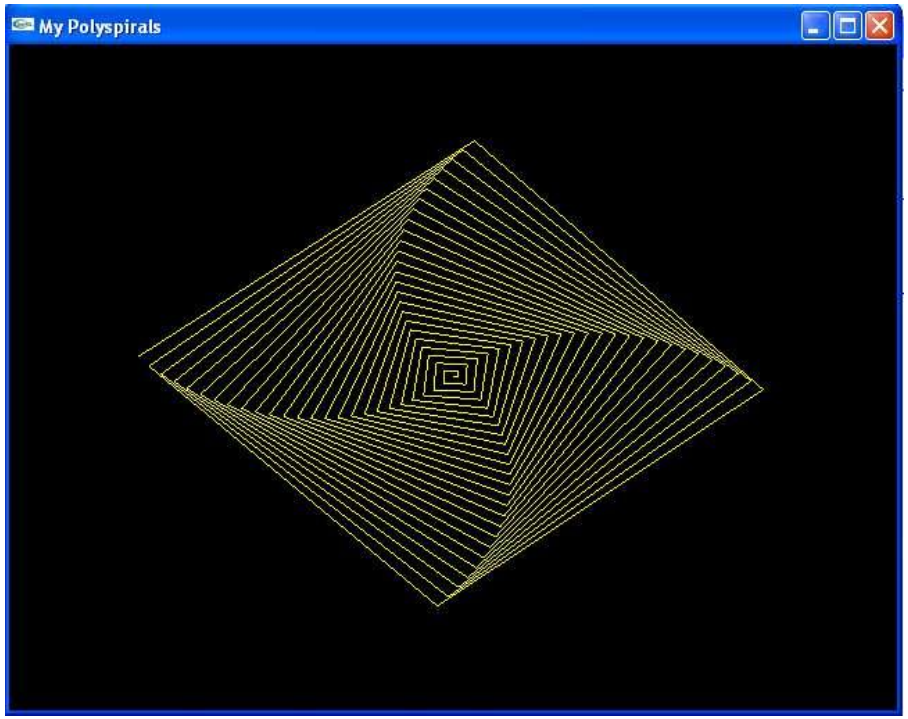


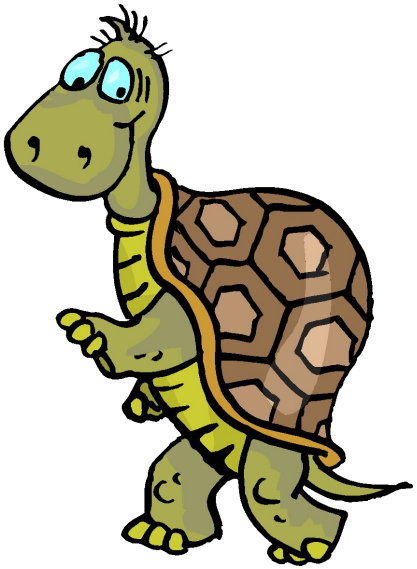


Grafika żółwia

Np.: Polyspirals

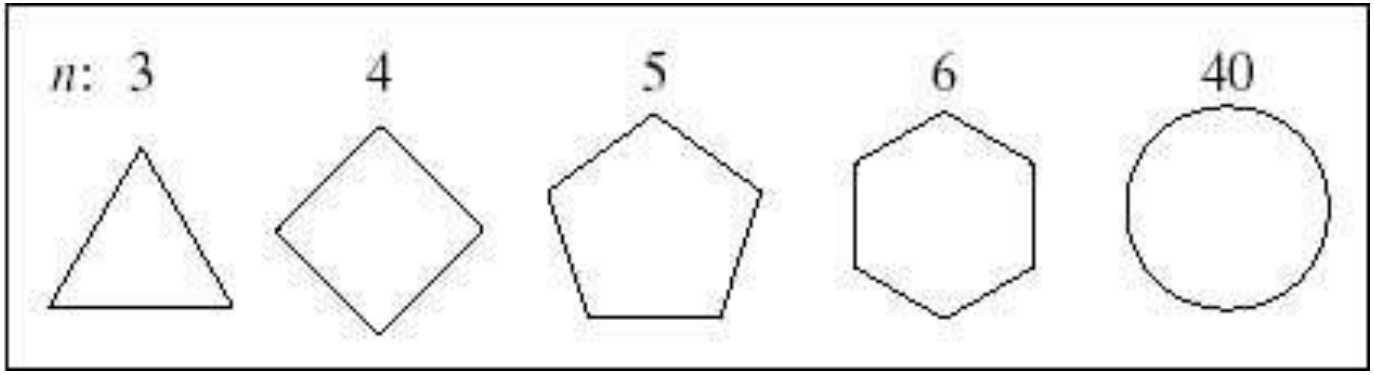
float angle = 89.5, inc = 1;





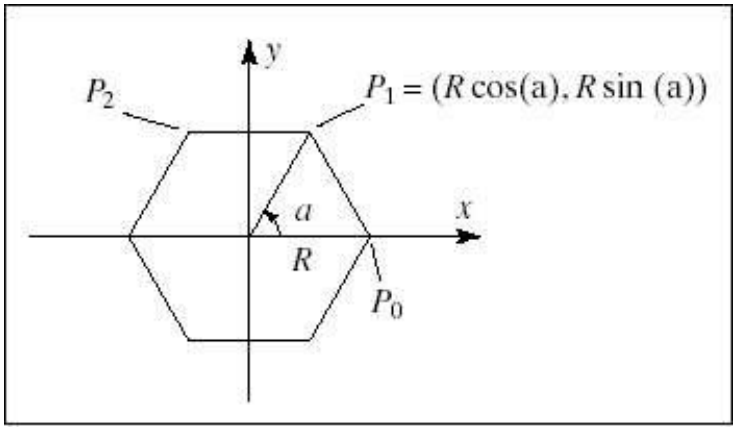
Regularne Wieloboki

Wielobok jest regularny jeśli wszystkie bokimają równą długość.



Regularne Wieloboki

Do rysowania regularnego wieloboku potrzeba określić środek, i kąt obrotu.



$$p_i = \left(R\cos(2\pi i / n), R\sin(2\pi i / n)\right)$$

Regularne Wieloboki

```
ngon(6,0.0,0.0,50.0, 20.0);
ngon(10,0.0,0.0,40.0, 30.0);
ngon(4,0.0,0.0,30.0, 10.0);
ngon(20,0.0,0.0,20.0, 5.0);
```

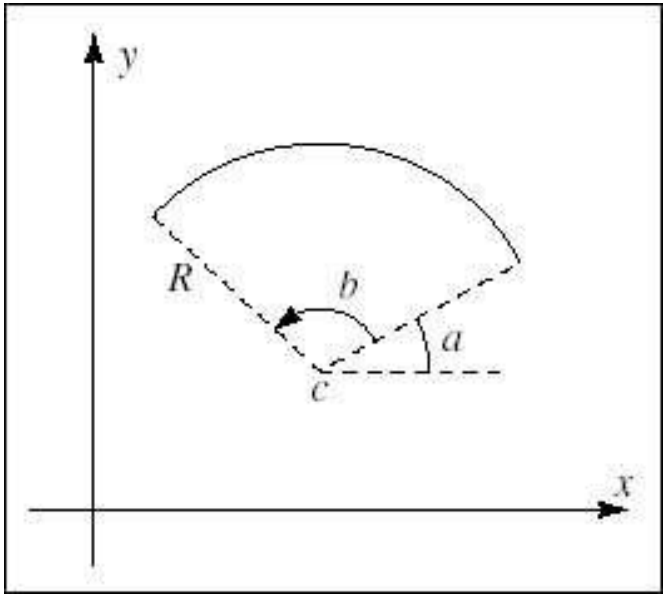
- ngon.cpp

Okręgi i łuki

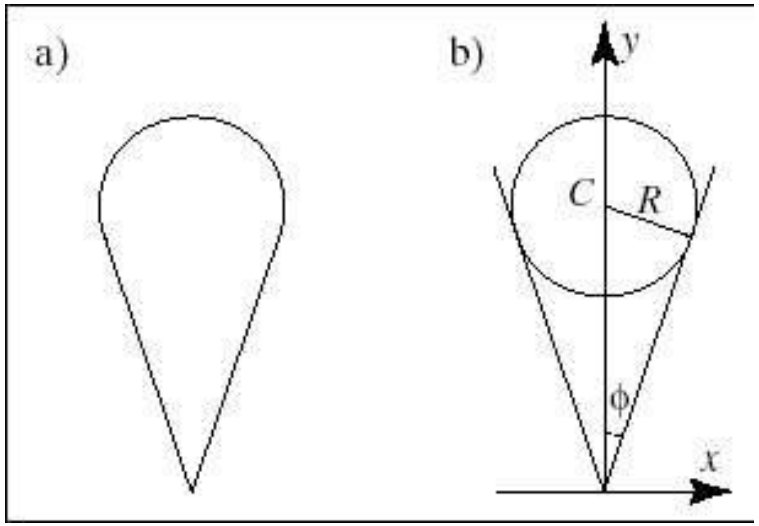
```
void drawCircle(Point2 center, float radius)
{
    const int numVerts = 50;
    ngon(numVerts, center.getX(), center.getY(), radius,
   0);
}
```

Okręgi i łuki

Musimy znać, środek, promień, kąt startowy (a) i kąt (b).



Okręgi i łuki



Krzywe Parametryczne

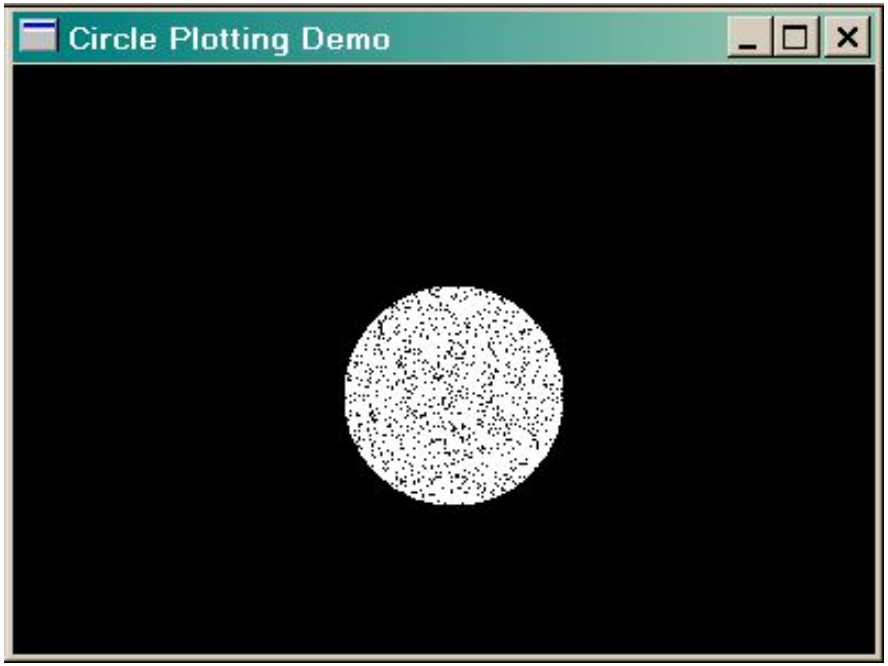
Krzywa może być opisana:

1. Implicitly
   - $F(x,y) = 0$;
   - $F(x,y) = (y - A_y)(B_x - A_x) - (x - A_x)(B_y - A_y)$
   - $F(x,y) = 0$ dla wszystkich punktów na krzywej.
   - $F(x,y) > 0$ dla wszystkich punktów na zewnątrz krzywej.
   - $F(x,y) < 0$ dla wszystkich punktów wewnątrz krzywej.

Krzywe Parametryczne

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

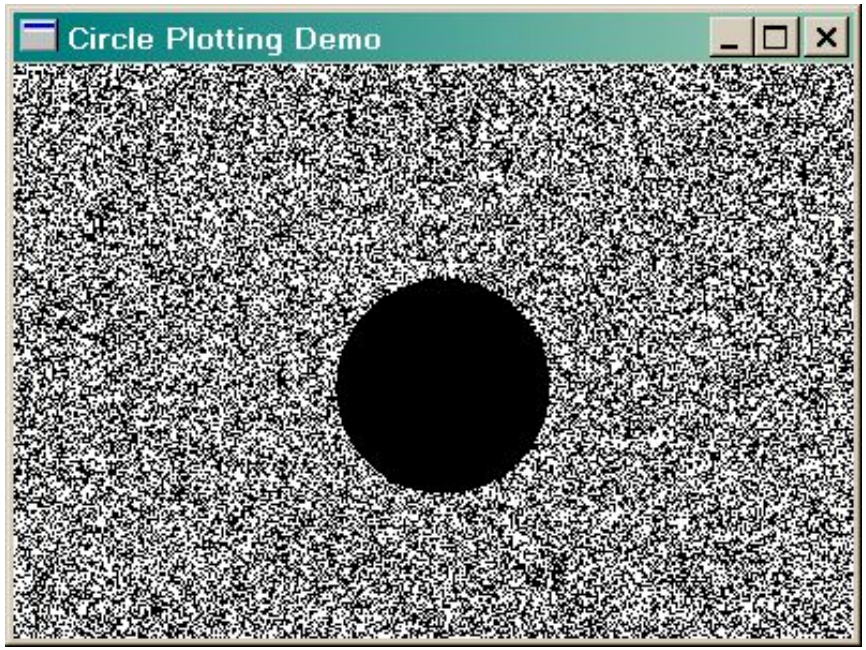
$F(x,y) < 0$ (wewnątrz)



Krzywe Parametryczne

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

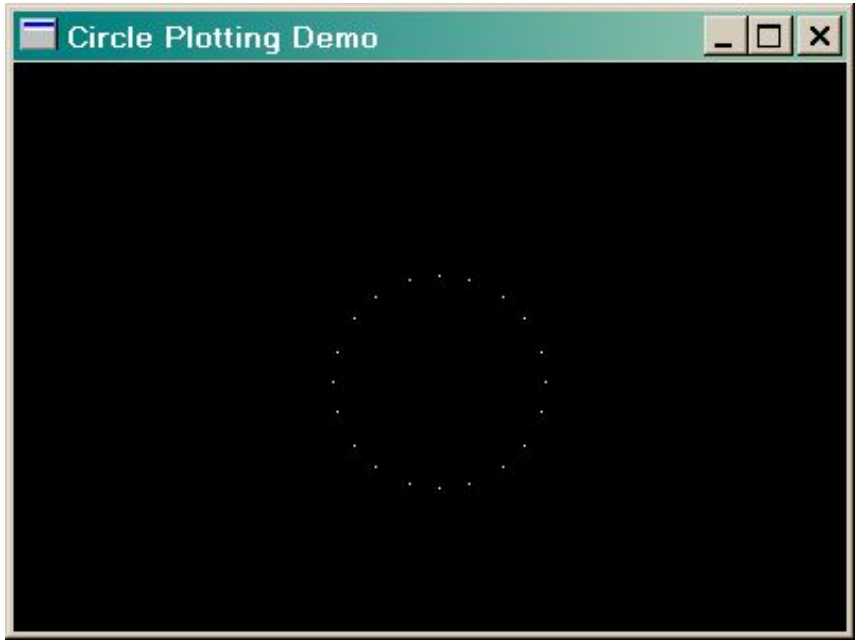
$F(x,y) > 0$ (na zewnątrz)



Krzywe Parametryczne

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

$F(x,y) = 0$ (na okręgu)



Krzywe Parametryczne

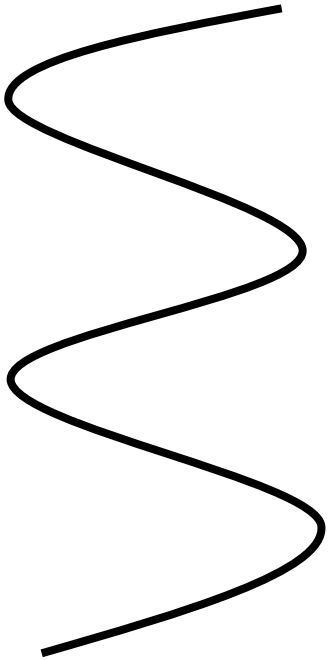
Problem z reprezentacją implicit

- Może być więcej niż jedna wartość x dla każdego y lub vice versa.

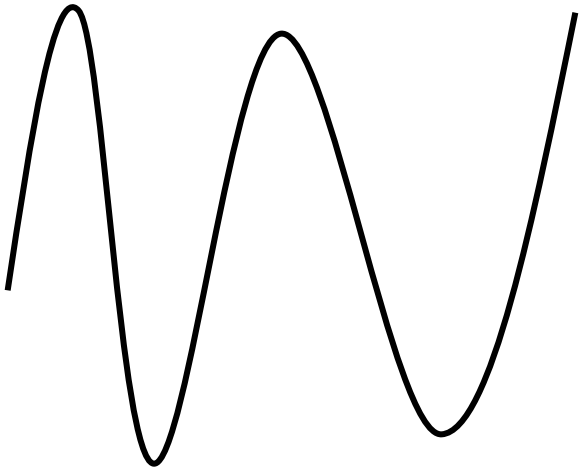
$$y = \pm\sqrt{R^2 - x^2}$$

- Pewne funkcje mają pojedyńcze wartości w x.
- Pewne funkcje mają pojedyńcze wartości w y.

Krzywe Parametryczne



Pojedyncza
wartość w y



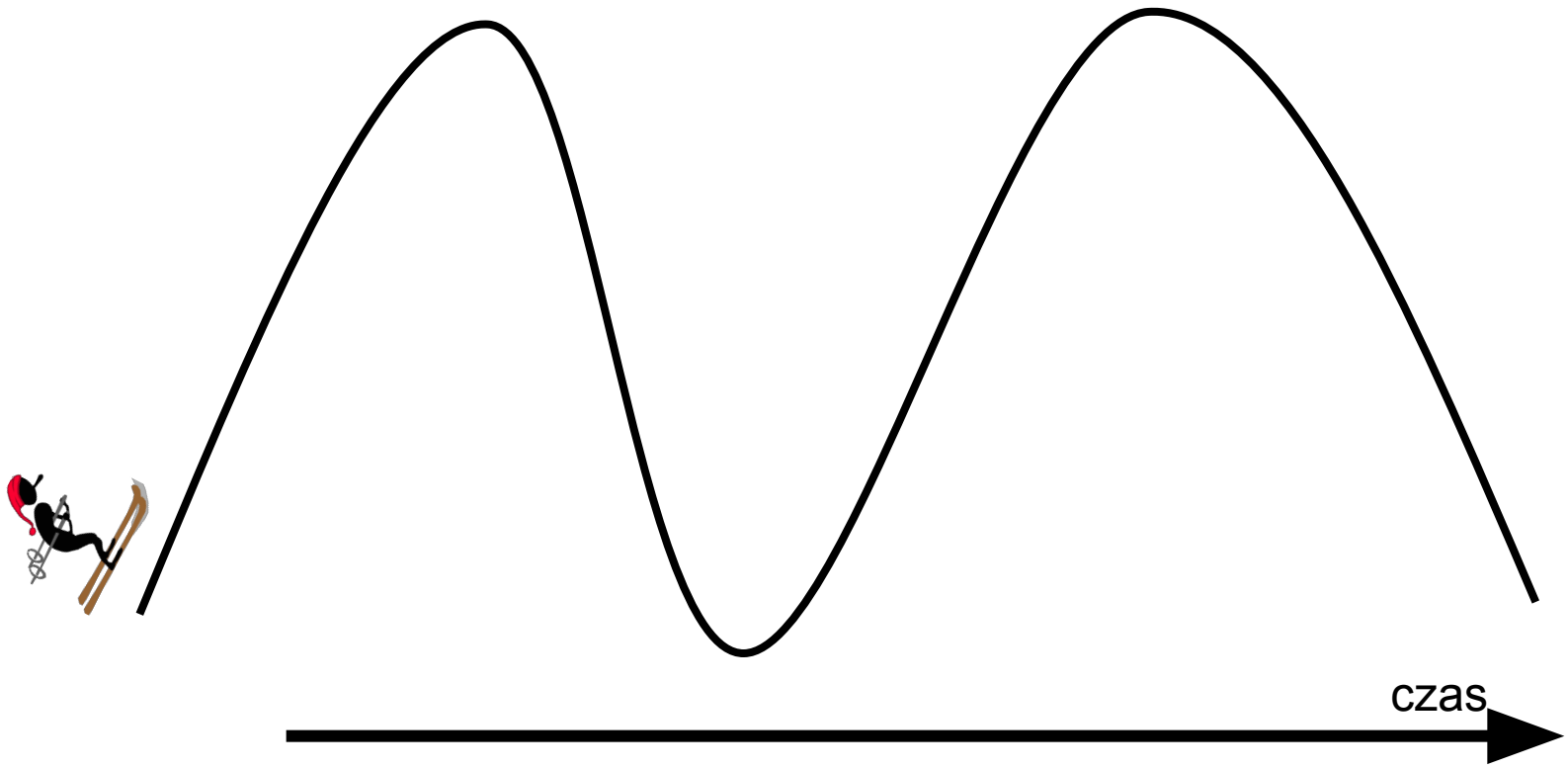
Pojedyncza
wartość w x

Krzywe Parametryczne

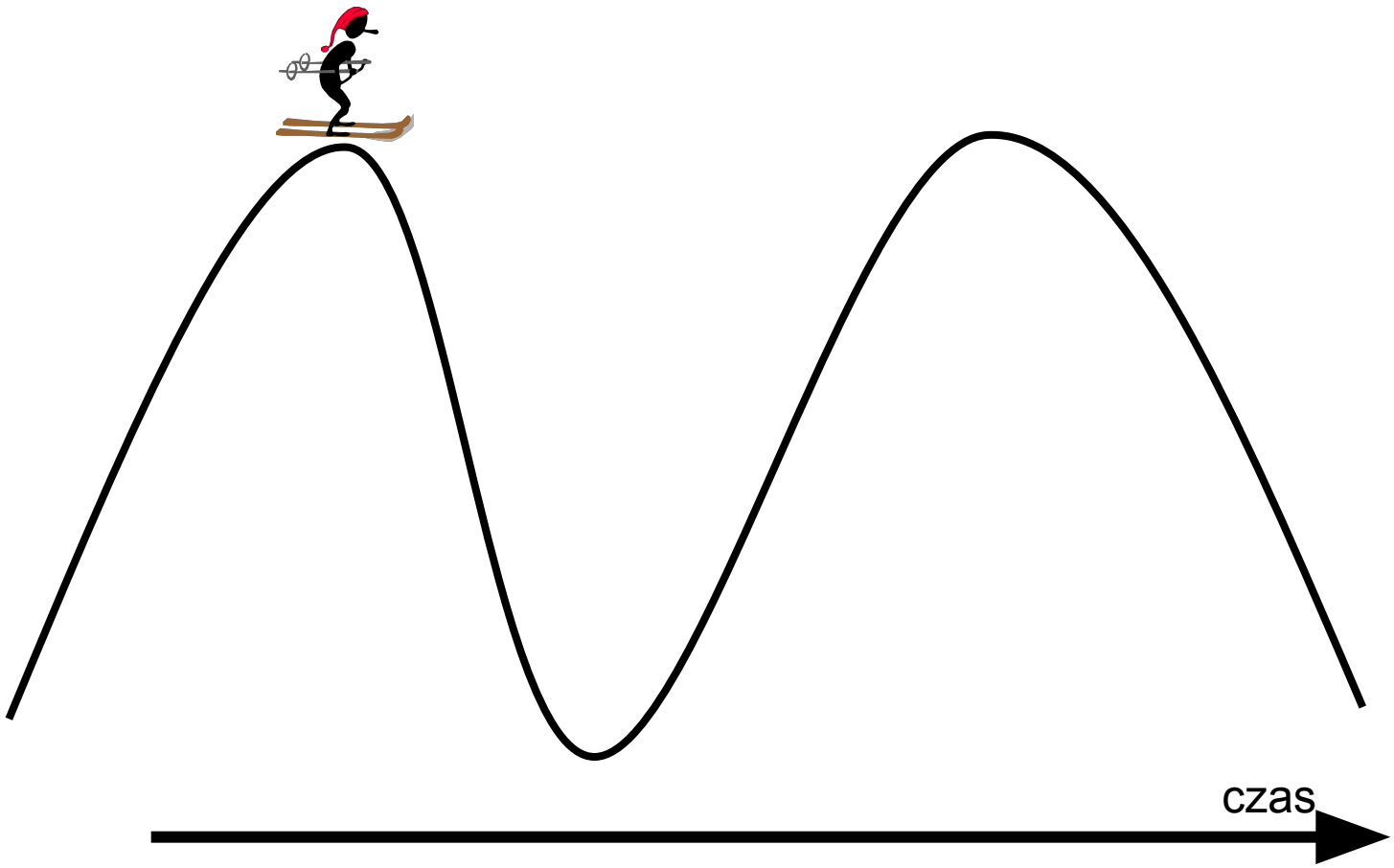
Opis:

2. Parametryczny

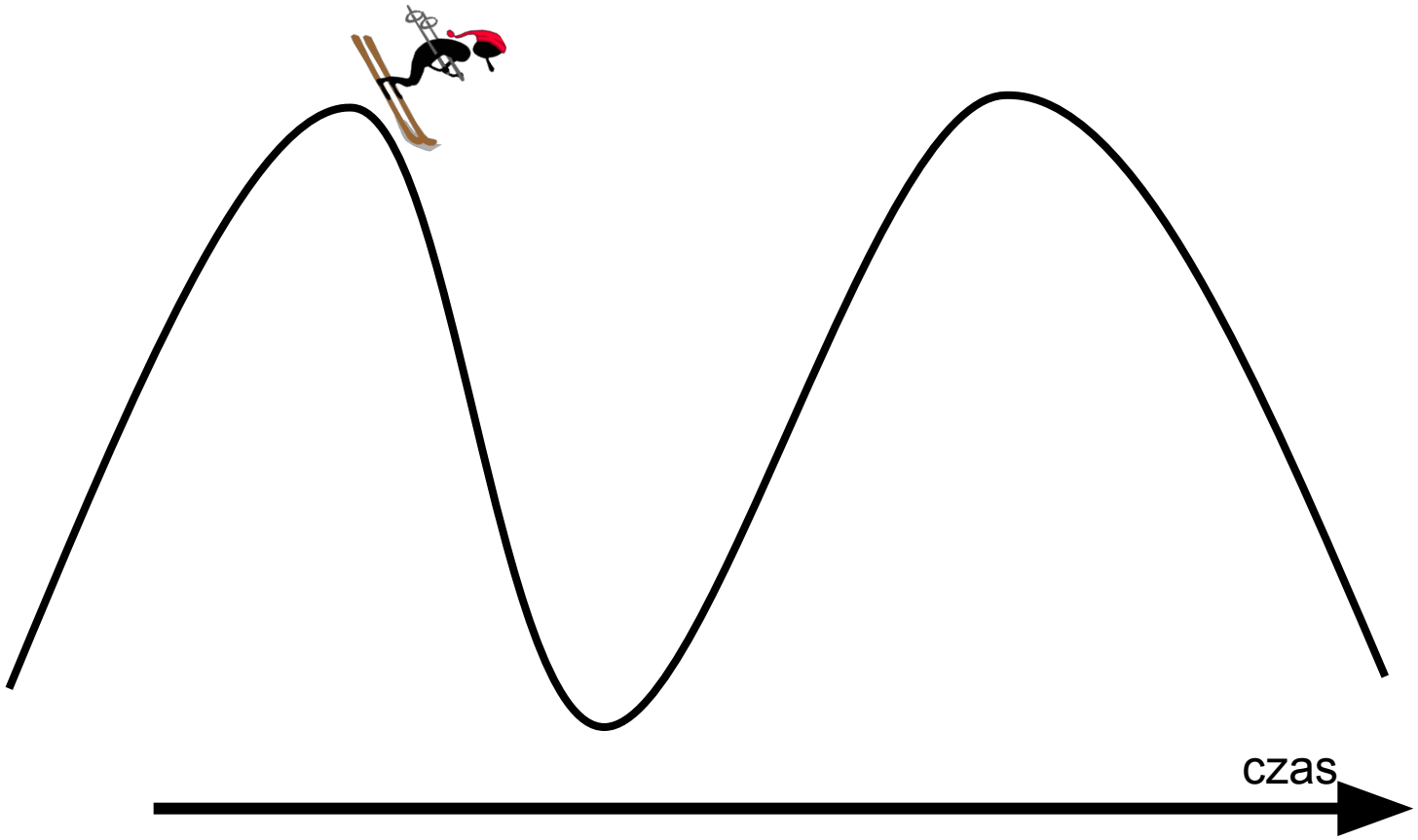
Krzywe Parametryczne



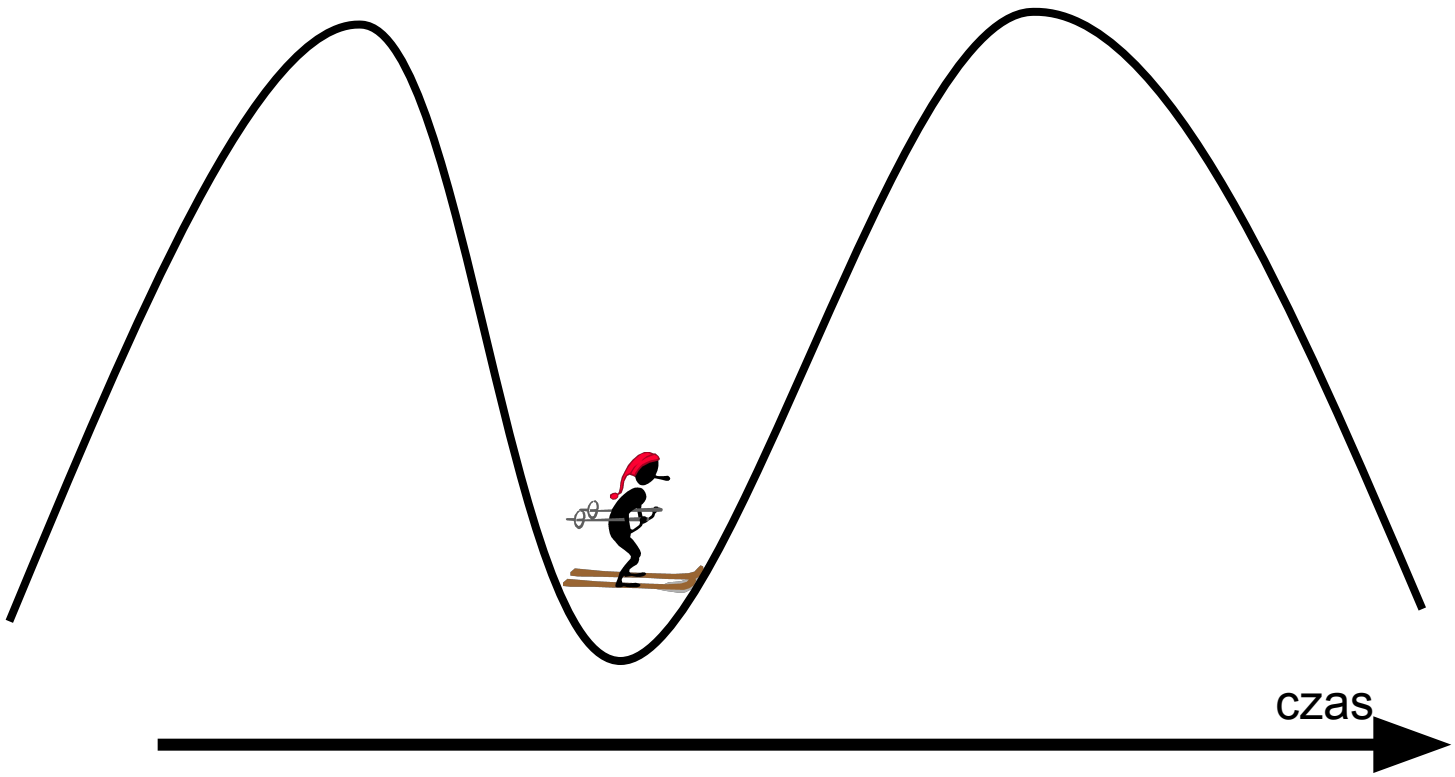
Krzywe Parametryczne



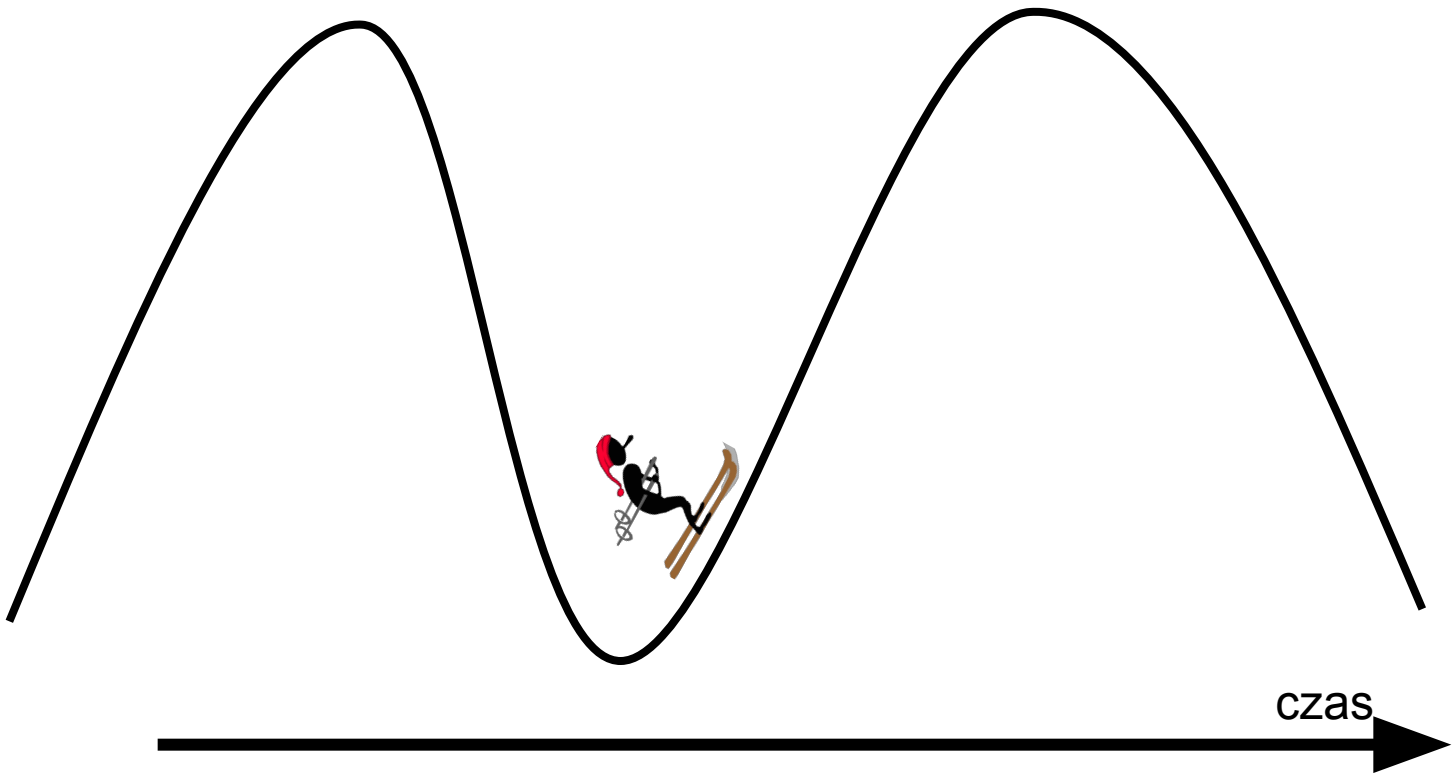
Krzywe Parametryczne



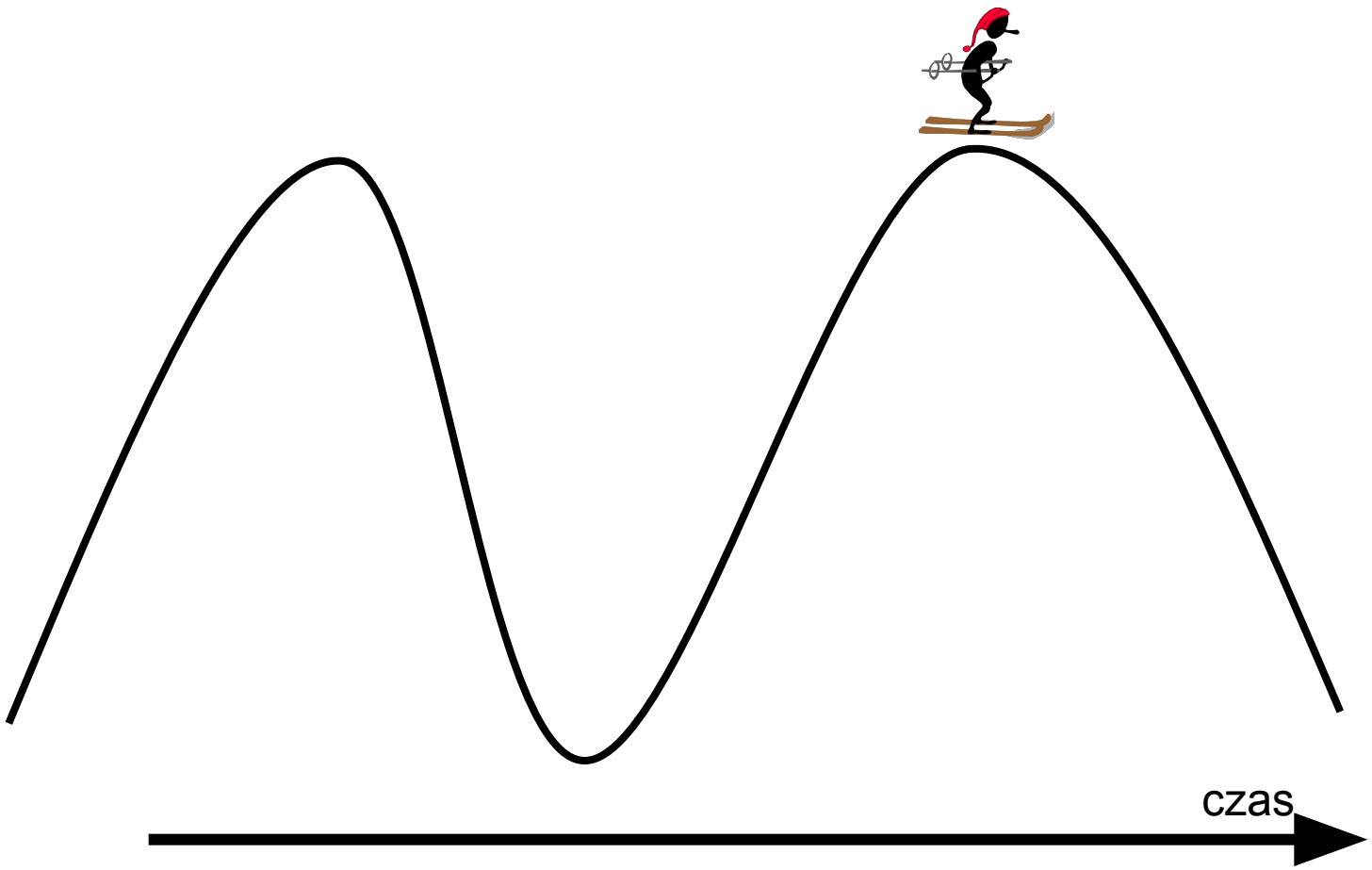
Krzywe Parametryczne



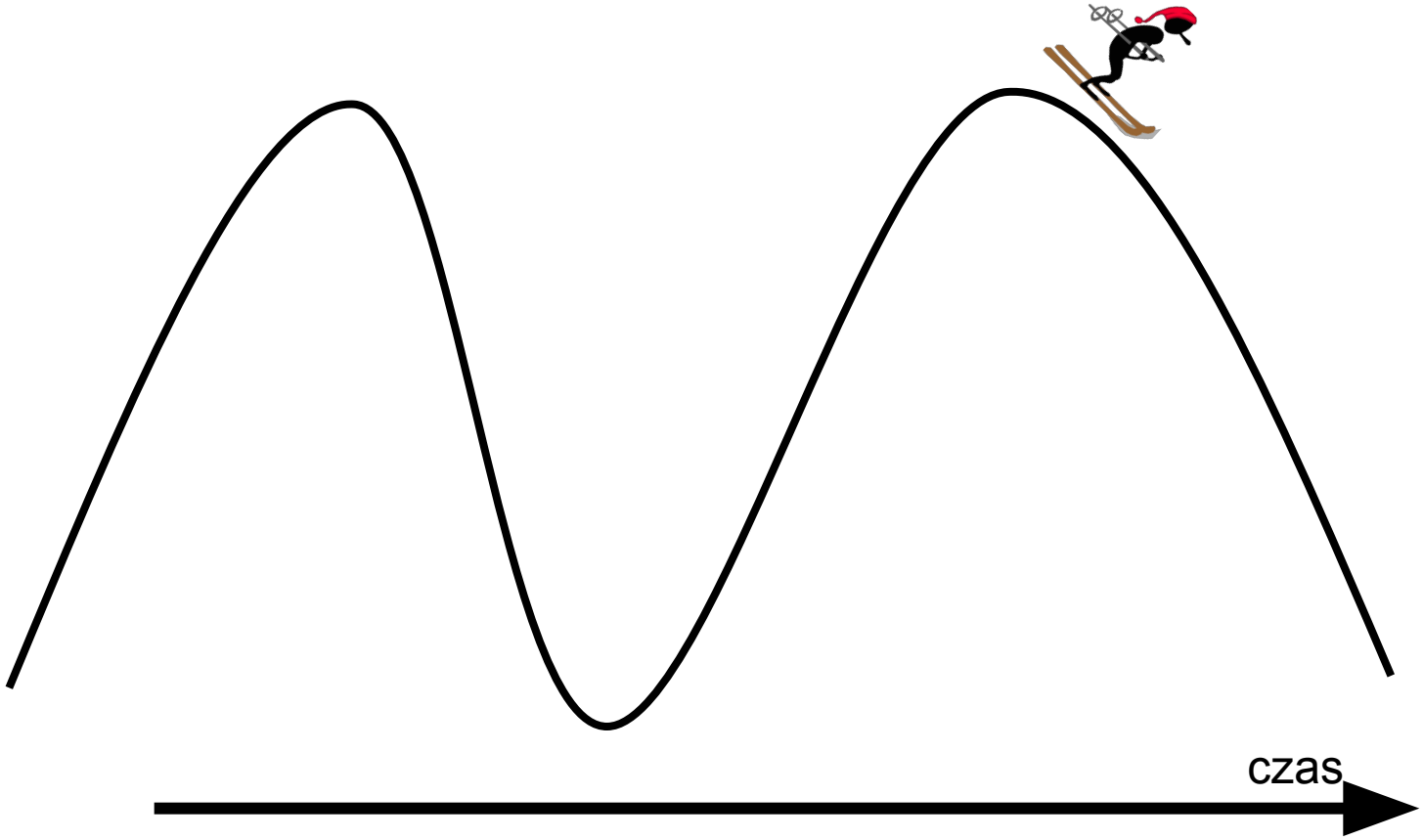
Krzywe Parametryczne



Krzywe Parametryczne



Krzywe Parametryczne



Krzywe Parametryczne

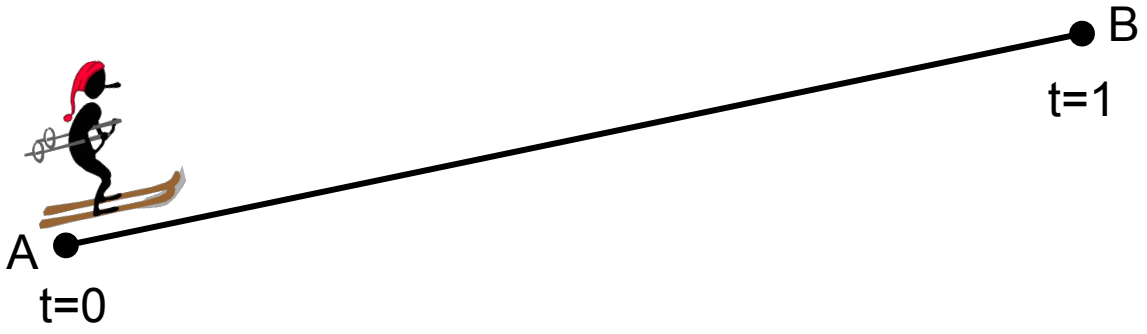
Mamy dwie funkcje, x() and y().

Położenie pewnego punktu w czasie jest określone przez (x(t), y(t)).

Krzywe Parametryczne

Np.:

Linia prosta od A do B. Dla czasu t=0 mamy A i dla t=1 mamy B.



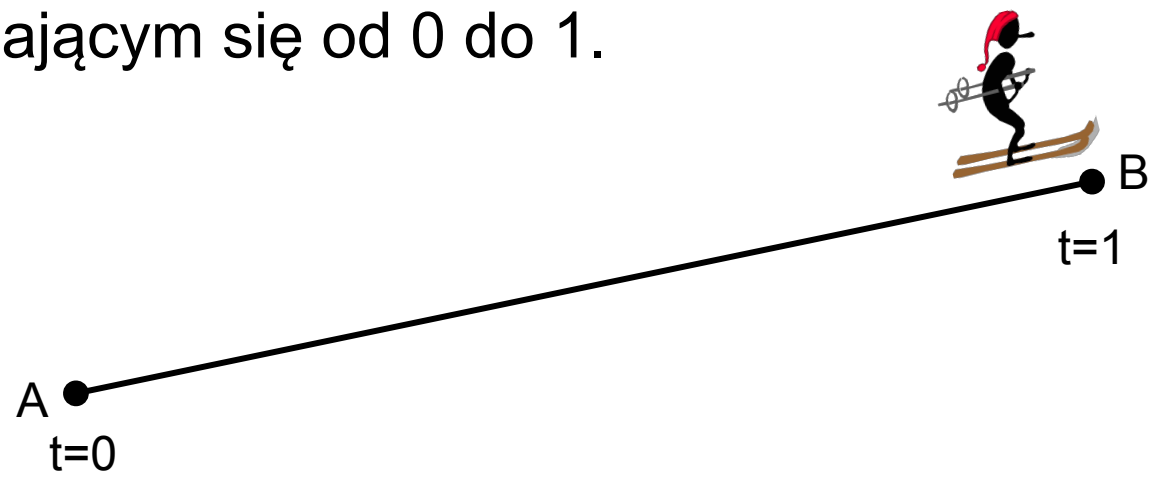
Krzywe Parametryczne

Teraz:

$x(t) = A_x + (B_x - A_x)t$

$y(t) = A_y + (B_y - A_y(t)$

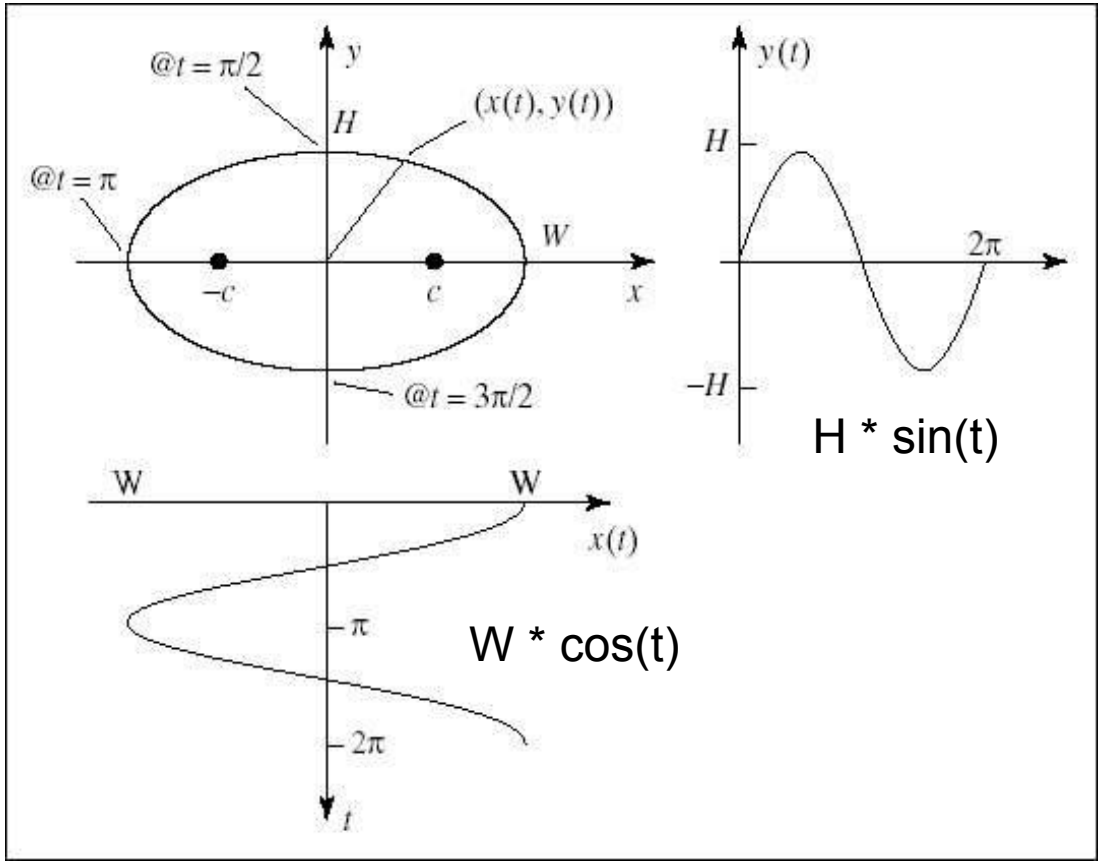
Punkt porusza się między A i B przy t zmieniającym się od 0 do 1.



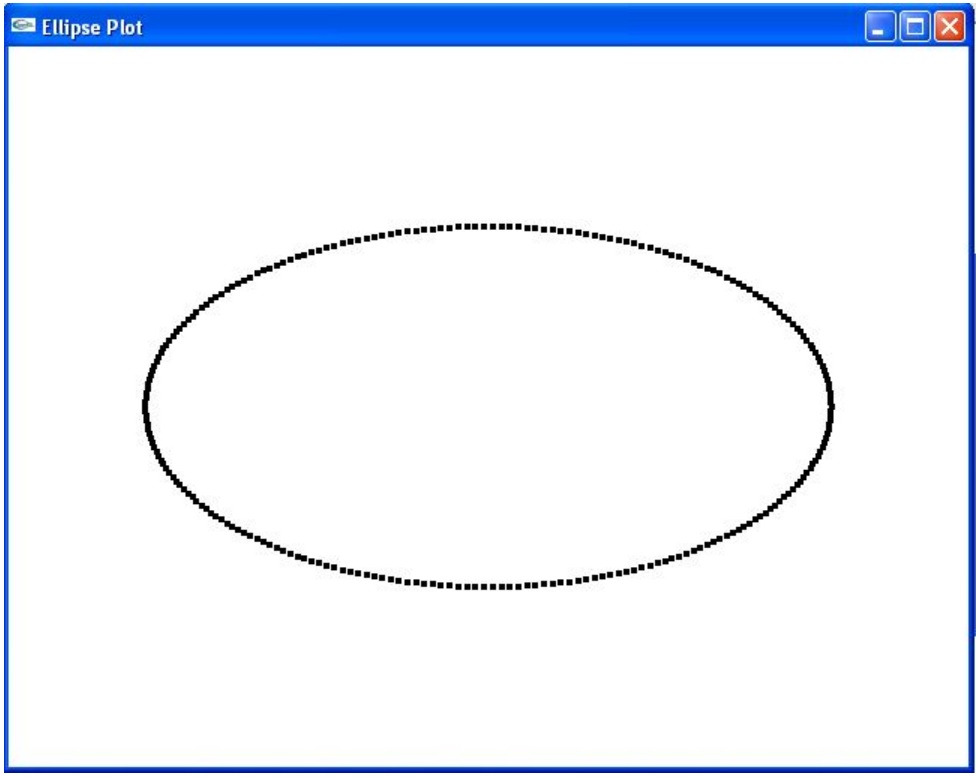
Krzywe Parametryczne

Np.:

Elipsa.



Krzywe Parametryczne



Współrzędne biegunowe

Każdy punkt na krzywej jest reprezentowany przez odległość (r) i kąt (Θ).

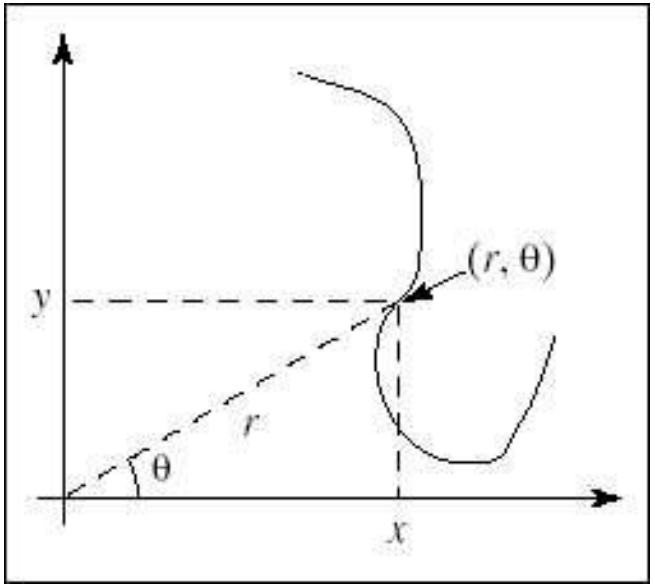
r i Θ są funkcjami t

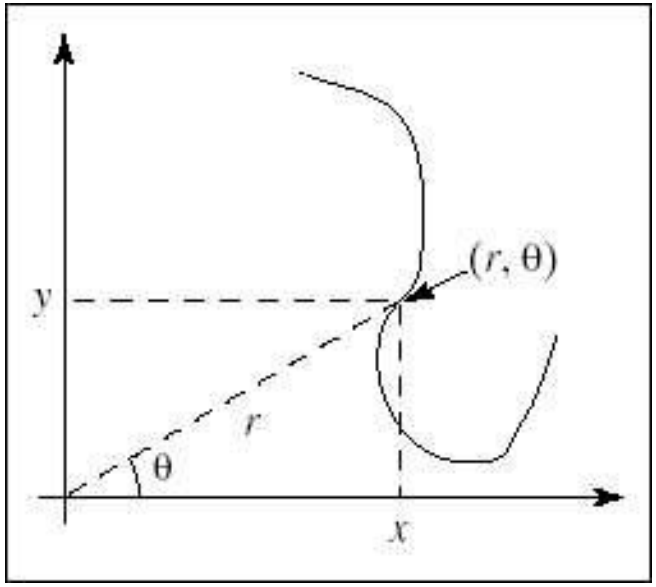
Mamy dla punktu określonego

przez (r(t), Θ(t))

x(t) = r(t)cos(Θ(t))

y(t) = r(t)sin(Θ(t))



Krzywe Parametryczne

- $x = f(\Theta)\cos(\Theta)$
- $y = f(\Theta)\sin(\Theta)$