



**Wektorowy model obrazu**

# Plan wykładu

- Metody zapisu informacji w grafice komputerowej:  
Grafika wektorowa i grafika rastrowa
- Grafika wektorowa – zapis obiektów
- Przekształcenia geometryczne
- Efekty i zasady ich tworzenia
- Czcionki
- Formaty grafiki 2D

# Metody zapisu informacji w grafice komputerowej

- **Grafika wektorowa** – opis obiektów w postaci formuł matematycznych
- **Grafika rastrowa** – zapis informacji w sposób „ciągły” w postaci macierzy (mapy pikseli), z których każdy przechowuje określoną wartość



Raster (PNG)



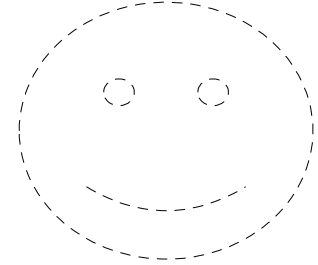
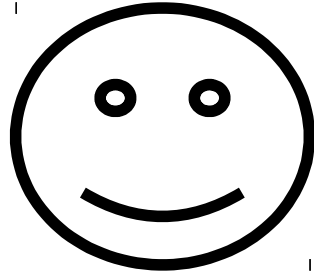
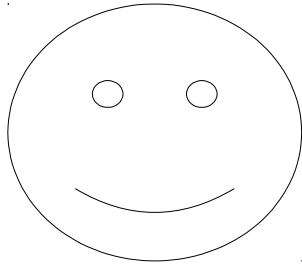
Vector (SVG)

# Grafika wektorowa

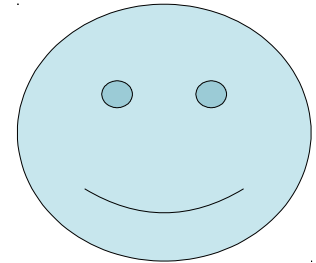
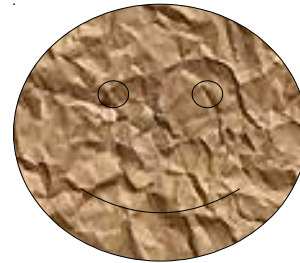
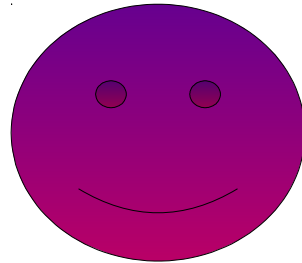
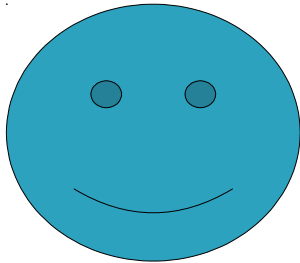
- Grafika wektorowa (obiektowa)
  - nazwa może być nieco myląca, ponieważ obrazy mogą składać się nie tylko z wektorów (odcinków), ale również z innych figur geometrycznych umiejscowionych w matematycznie zdefiniowanym układzie współrzędnych,
  - Cecha grafiki wektorowej - zapamiętywane są charakterystyczne dla danych figur dane (parametry), a dla krzywych parametrycznych współrzędne punktów kontrolnych.
  - Grafika 3D to też grafika wektorowa

# Obiekty w grafice wektorowej

- Druga nazwa grafiki wektorowej – grafika obiektowa – związana jest z faktem, iż obraz opisany jest za pomocą tzw. **obiektów**, które zbudowane są z podstawowych elementów nazywanych **prymitywami**
  - **Prymitywy** - proste figury geometryczne (odcinki, krzywe, okręgi, wielokąty) stosowane do budowy bardziej skomplikowanych obiektów
  - Każdy z prymitywów opisywany jest za pomocą parametrów
- Obiekty takie mają także określone **atrybuty**



# Atrybuty obiektów wektorowych



# Atrybuty

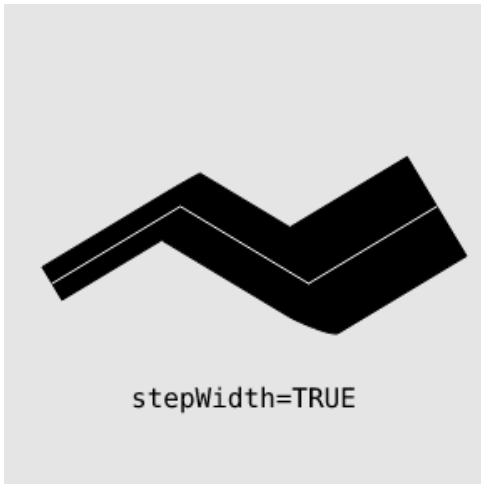
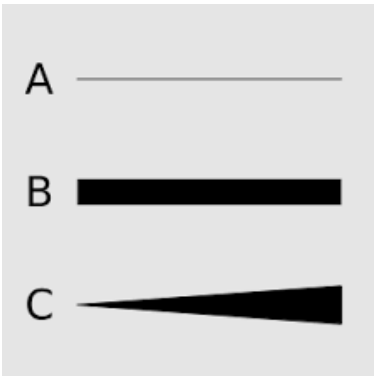
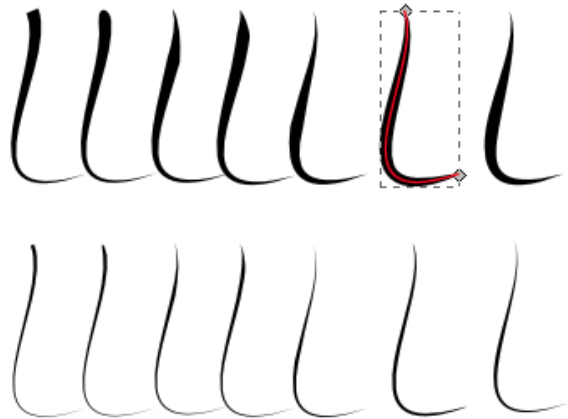
- Obrys (kontur, stroke)
  - Grubość
  - kreskowanie (dash)
  - Kolor
  - Przezroczystość
  - Obiekty
  - Zakończenia
  - Złączenia (joins)
- Wypełnienie
  - Kolor
  - Wzorek (pattern)
  - Kreskowanie (hash)
  - Gradient
  - Przezroczystość

# Szerokość obrysu

Examples:



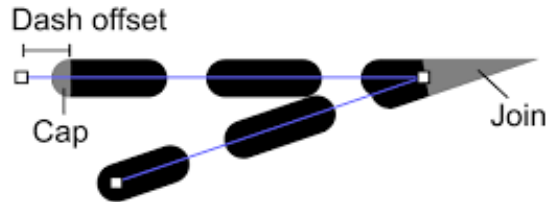
With less width:



stepWidth=TRUE



# Tworzenie wzorka konturu



{ -E- } Cable / Telephone - Stroke

Stroke Pattern Attributes

Shift: Fraction 1.000000  Single Segment

Repetitions: Unlimited 1

Stroke Pattern

0 20

Add

Delete

Length: Variable 9.000000 Width: None

Stroke Type: Dash Start: 0.000000

Invert at: None End: 0.000000

Corners: Break Dash Caps: Closed

Choose Stroke Style  
Unnamed-3 (sunflower.png)

Stroke line

Line width: 6.0 px

Solid color

Pattern

Line Style

Cap style:

Join style:

Miter limit: 10.0

Dash pattern:

Dash preset: Custom

Antialiasing

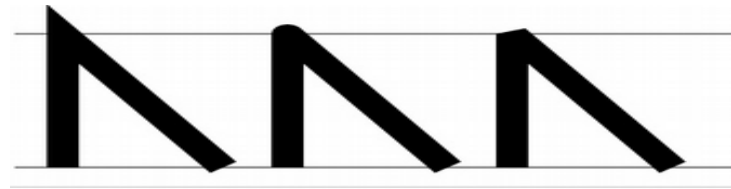
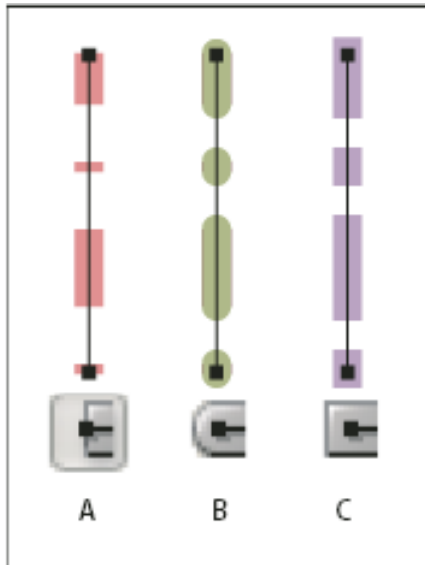
Stroke with a paint tool

Paint tool: Paintbrush

Emulate brush dynamics

Help Reset Stroke Cancel

# Złączenia linii obrysu



miter joint



round joint



bevel joint



'miter' joint



'round' joint

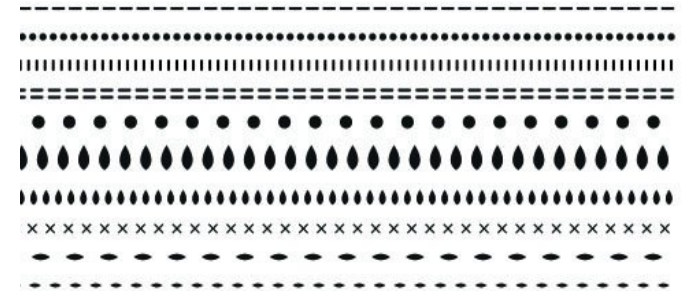
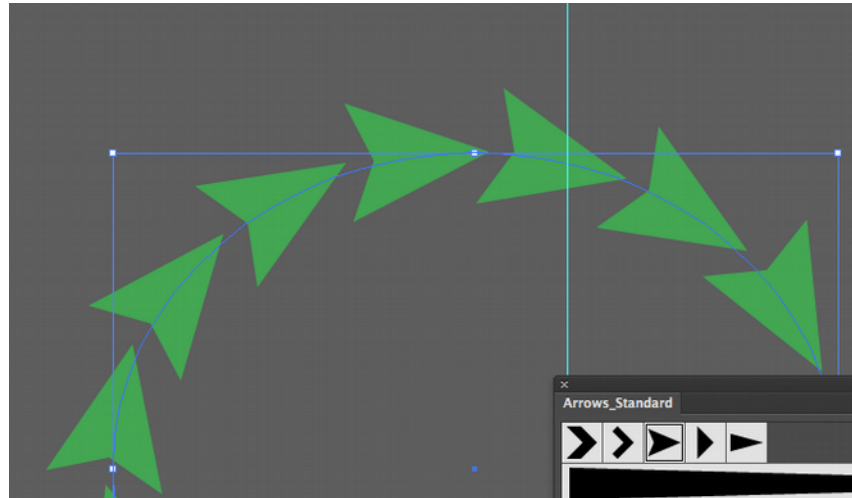
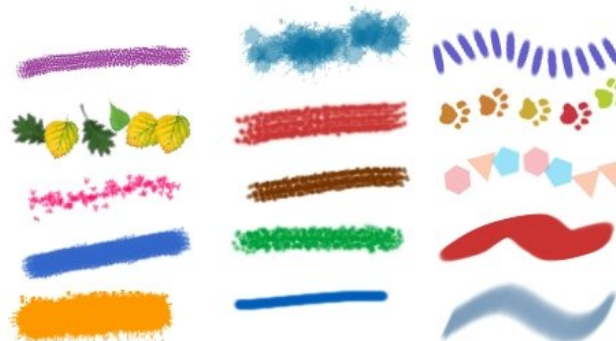


'bevel' joint

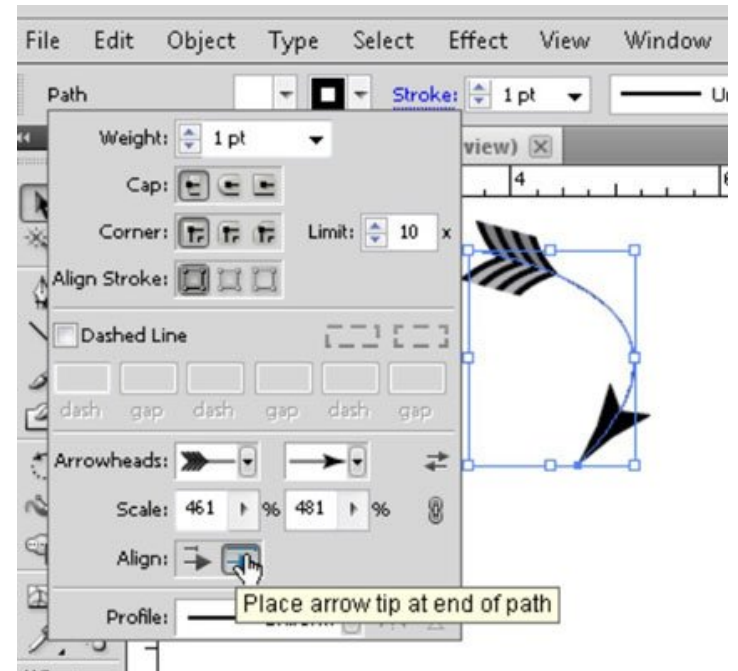
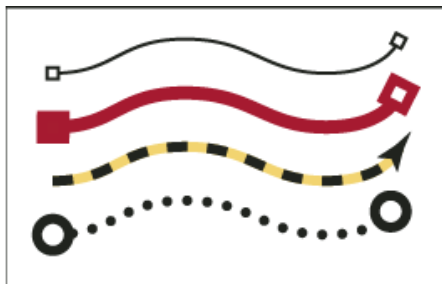
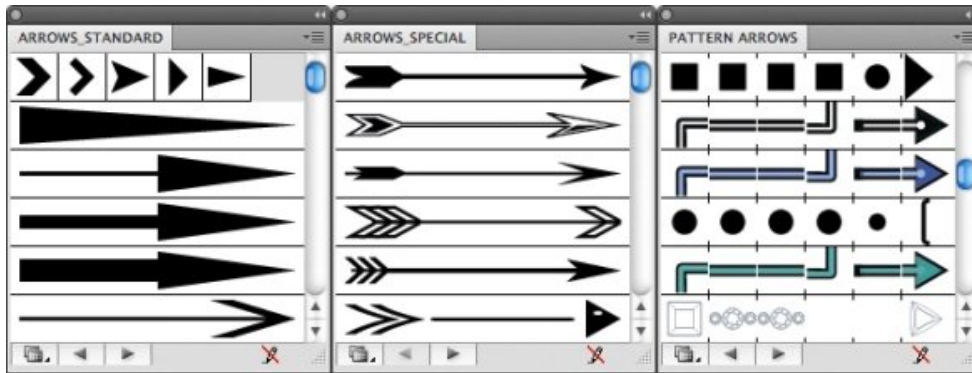


'arcs' joint

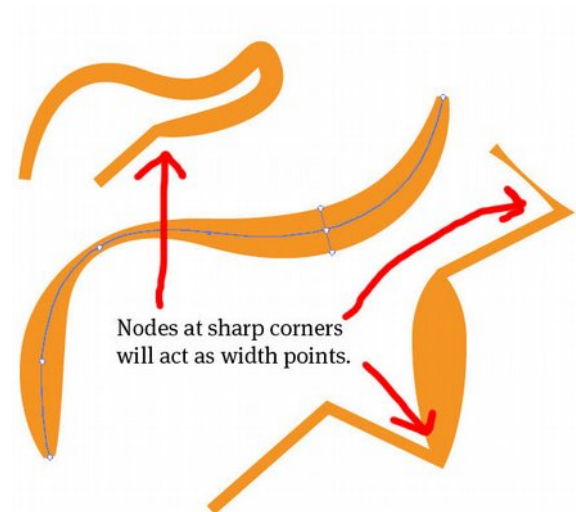
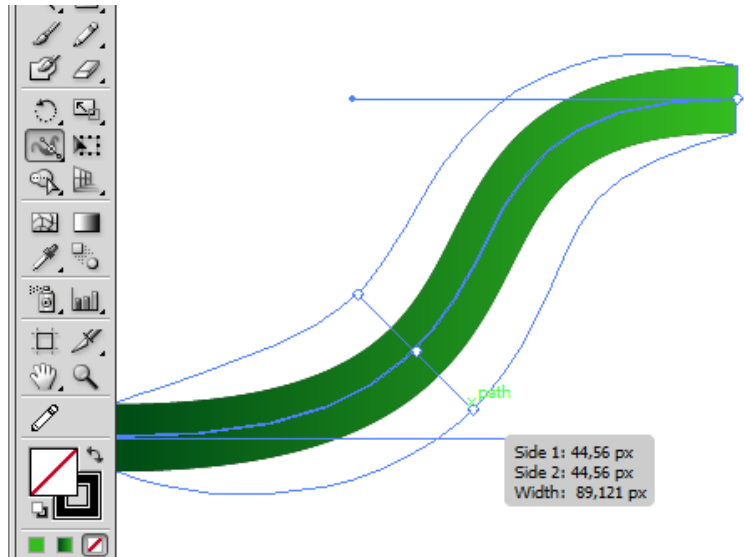
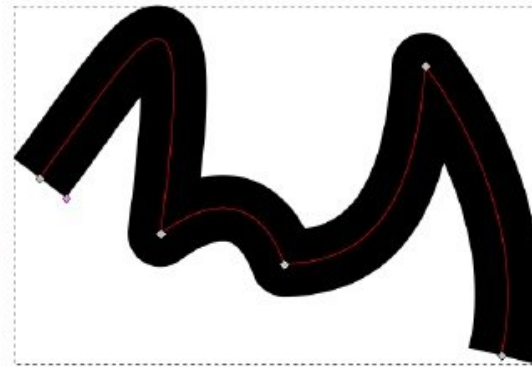
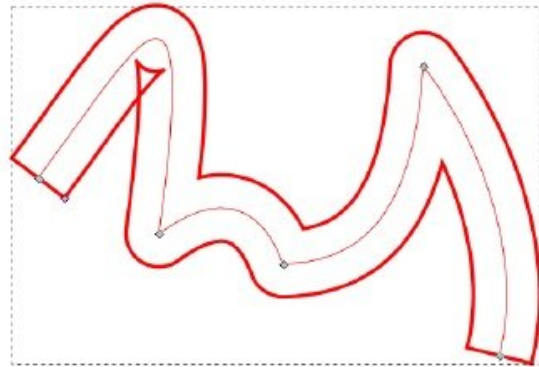
# Złożone wzorki konturu



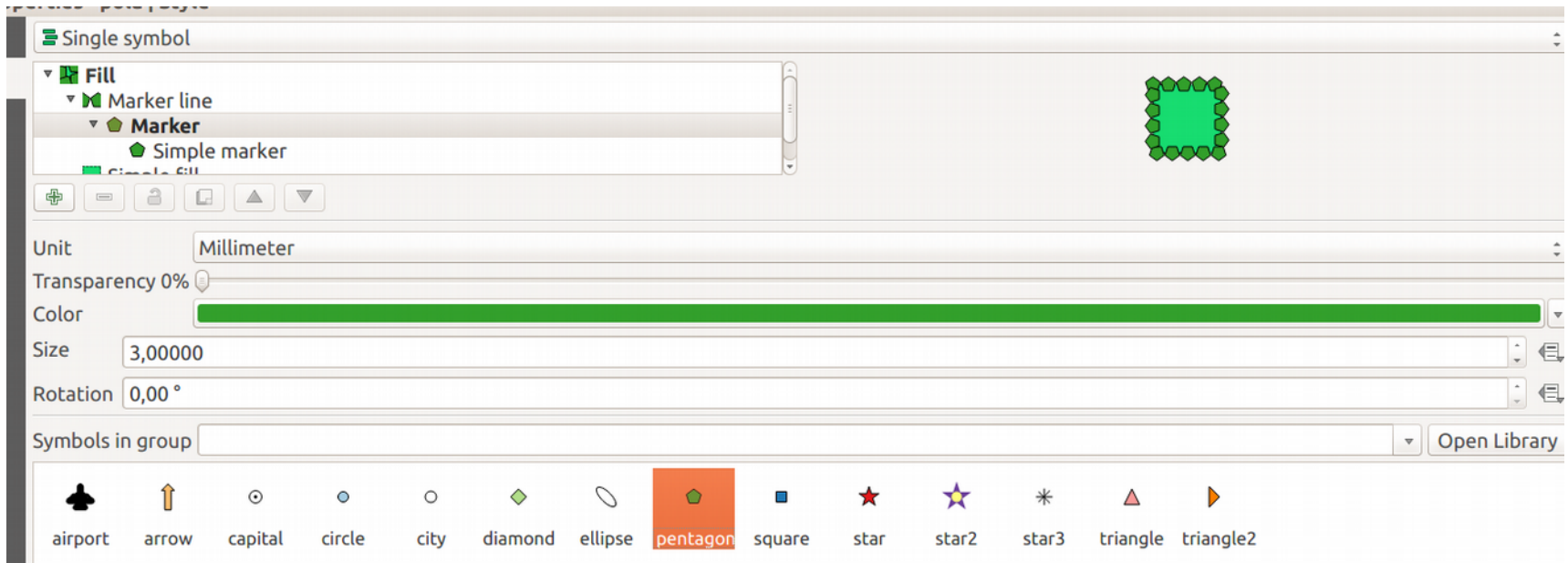
# Zakończenia obrysu



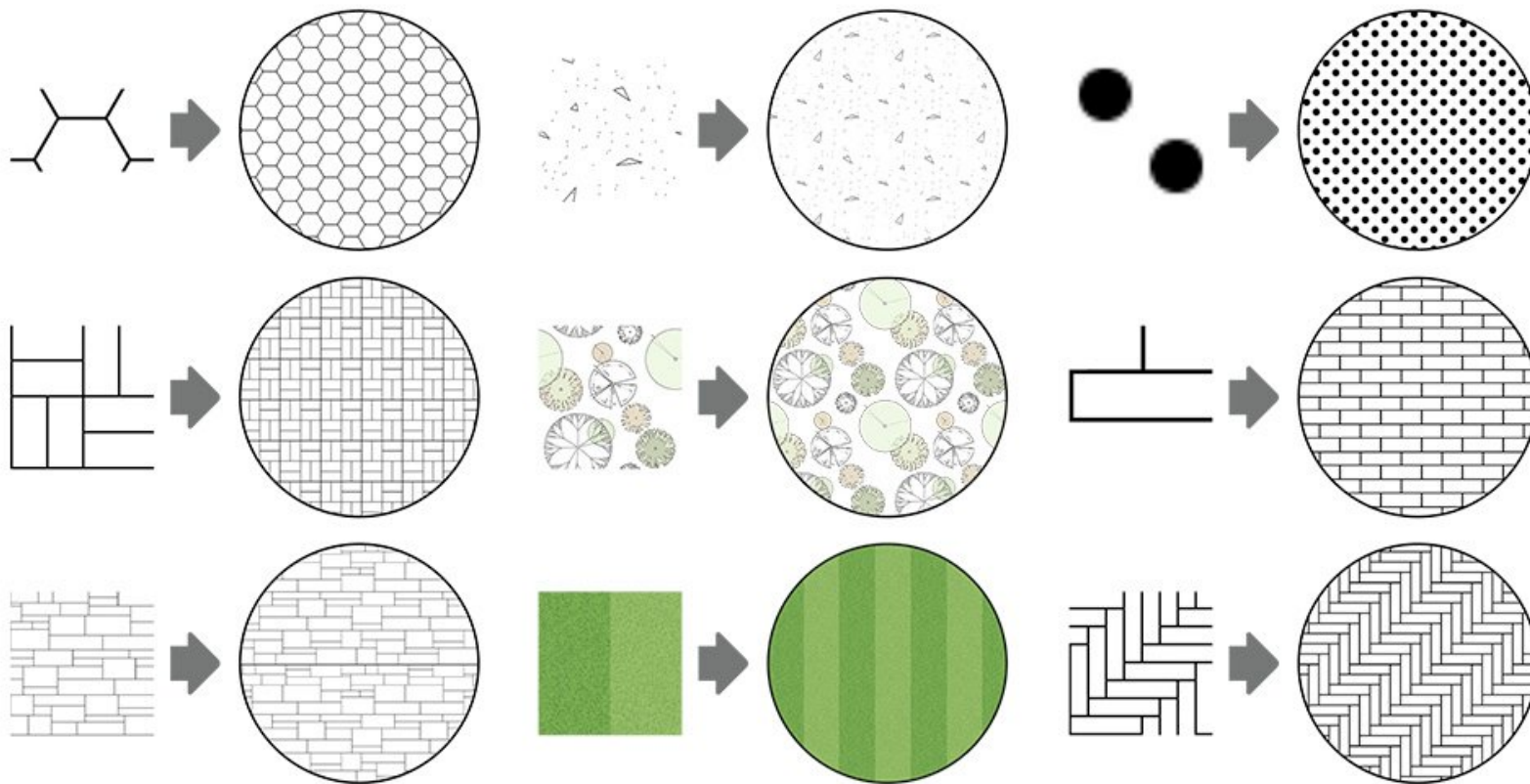
# Obrys artystyczny (kaligraficzny)



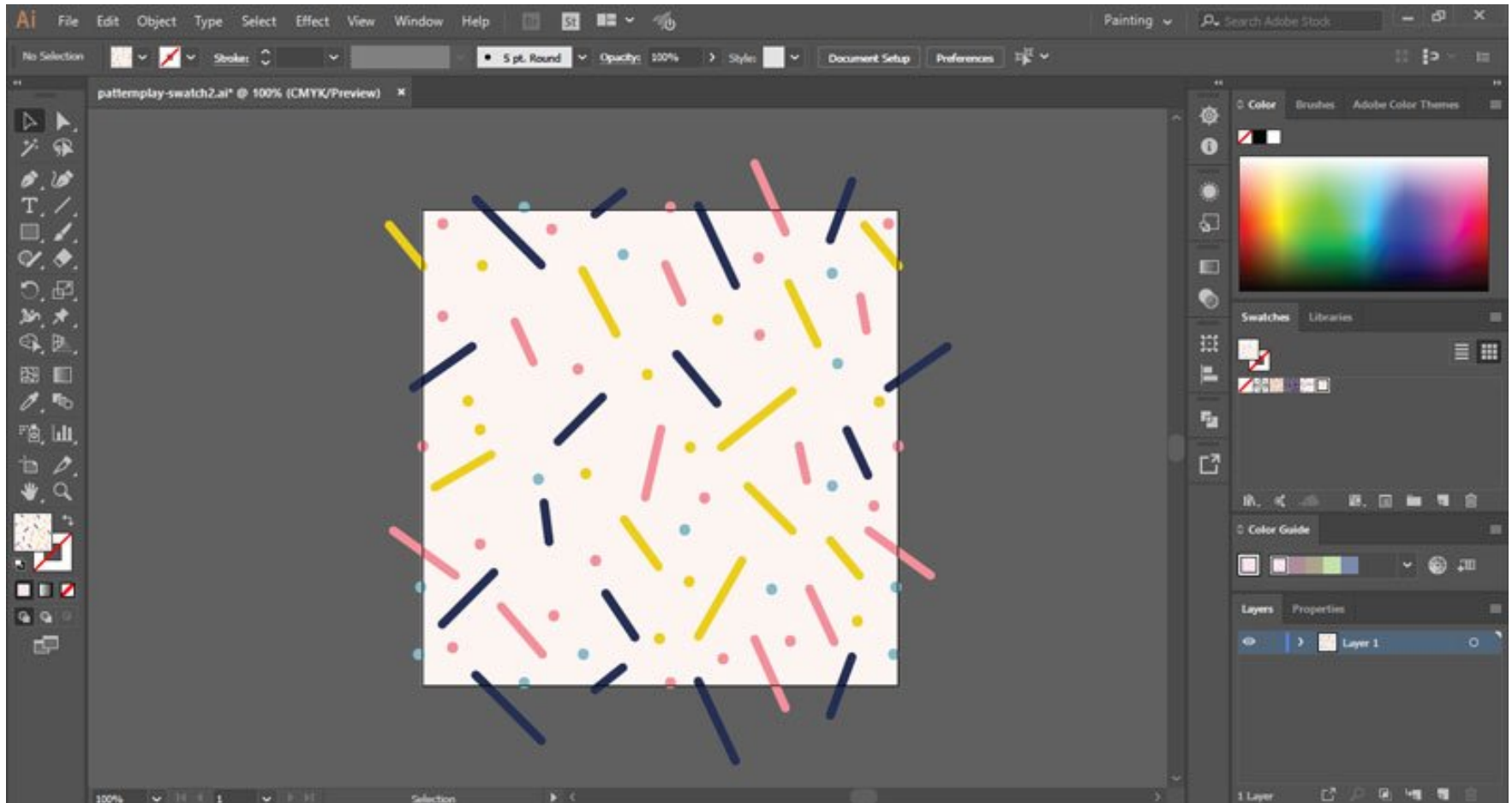
# Obrys a wizualizacja kartograficzna



# Wypełnienie wzorkiem

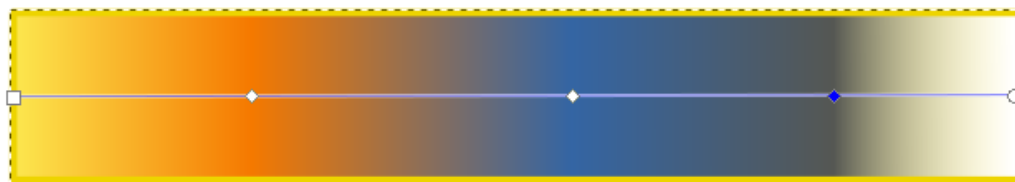
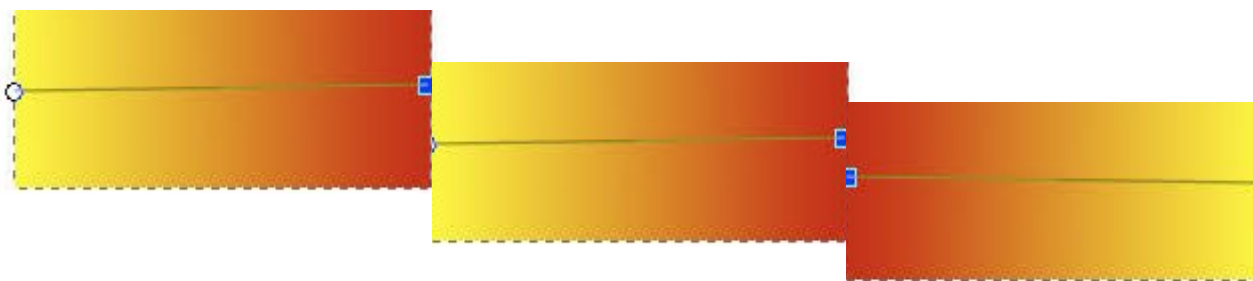


# Wzór powtarzalny

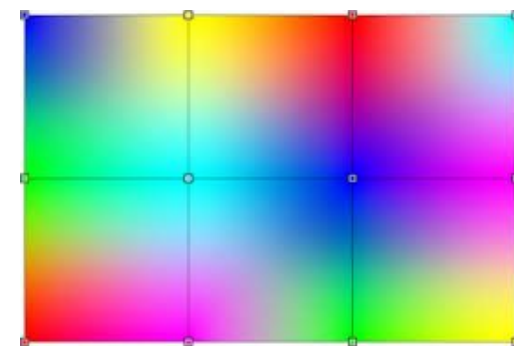




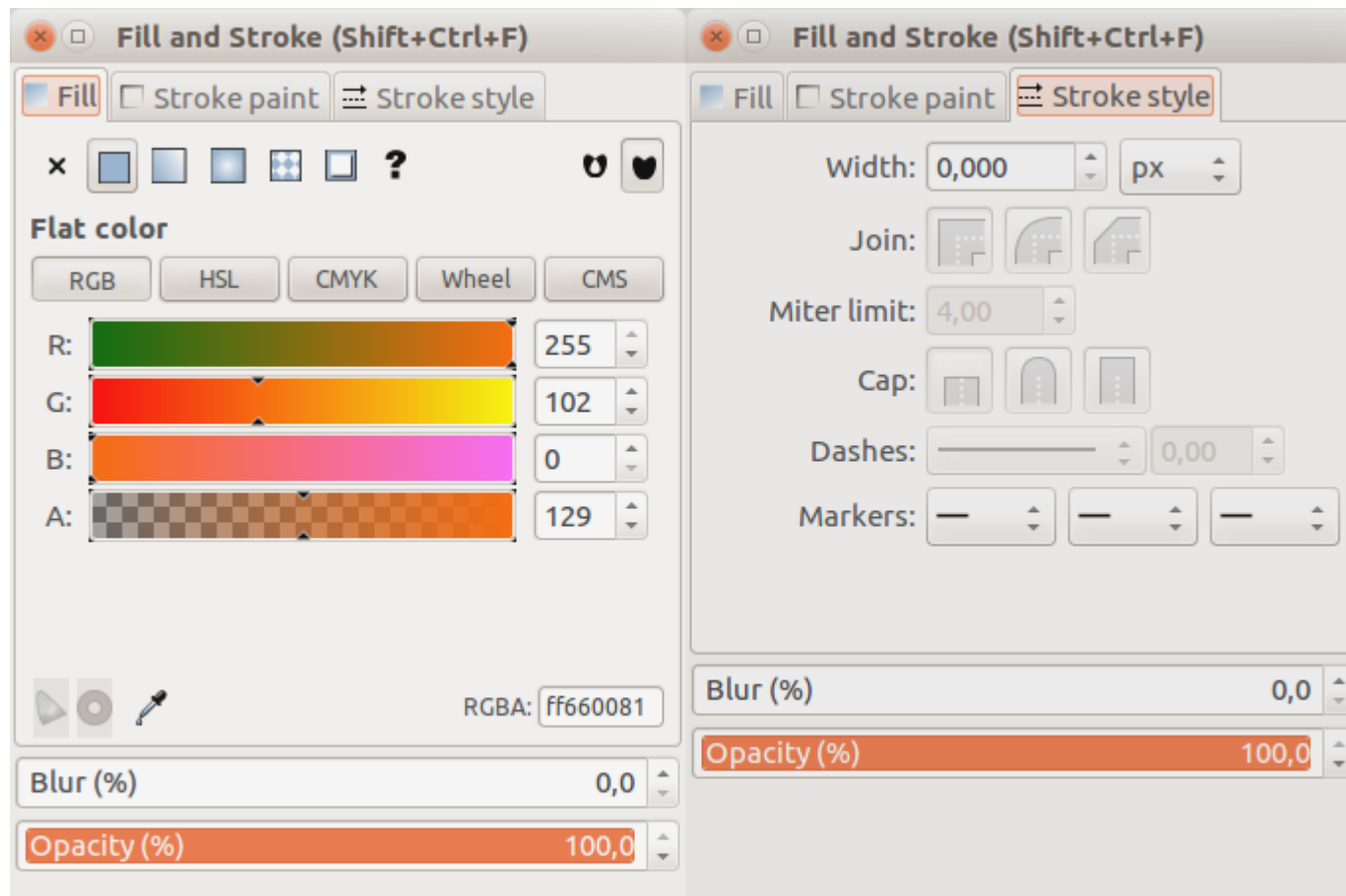
# Gradients – wypełnienia tonalne



- stop12694
- stop12710
- stop12712
- stop12714
- stop12696



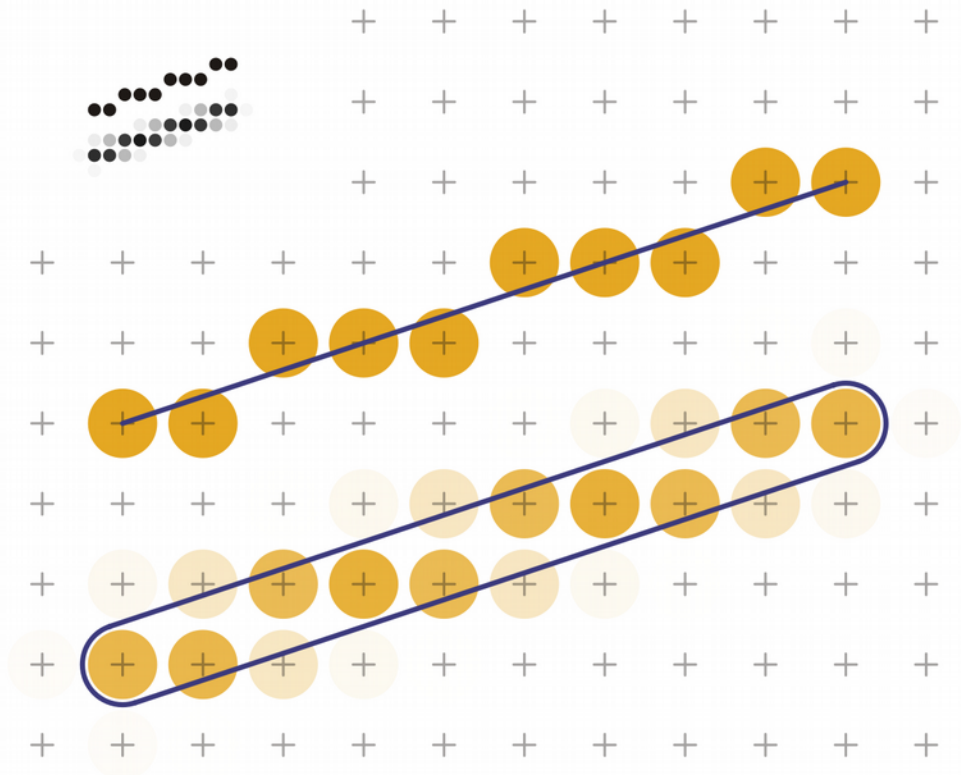
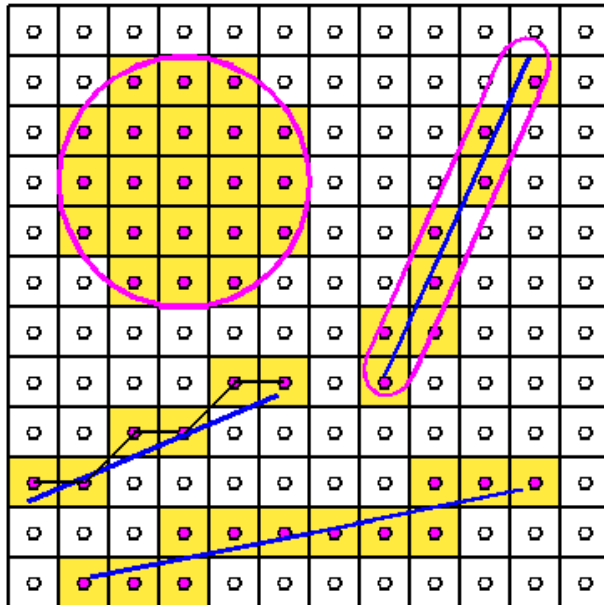
# Wypełnienie i Obrys - właściwości



# Wizualizacja wektora w siatce rastrowej

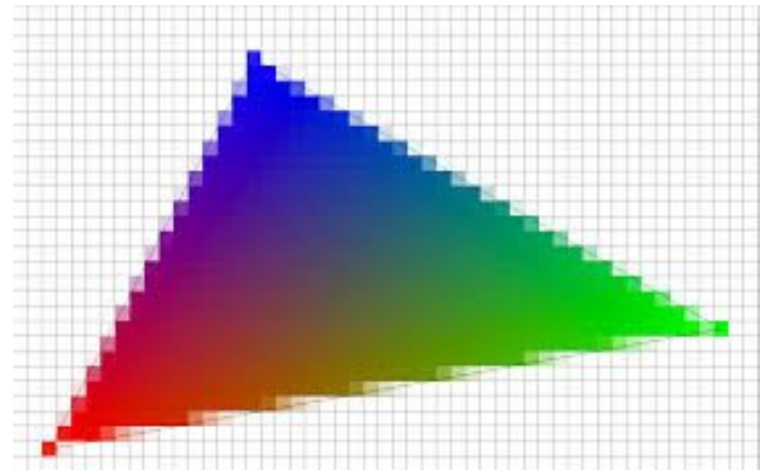
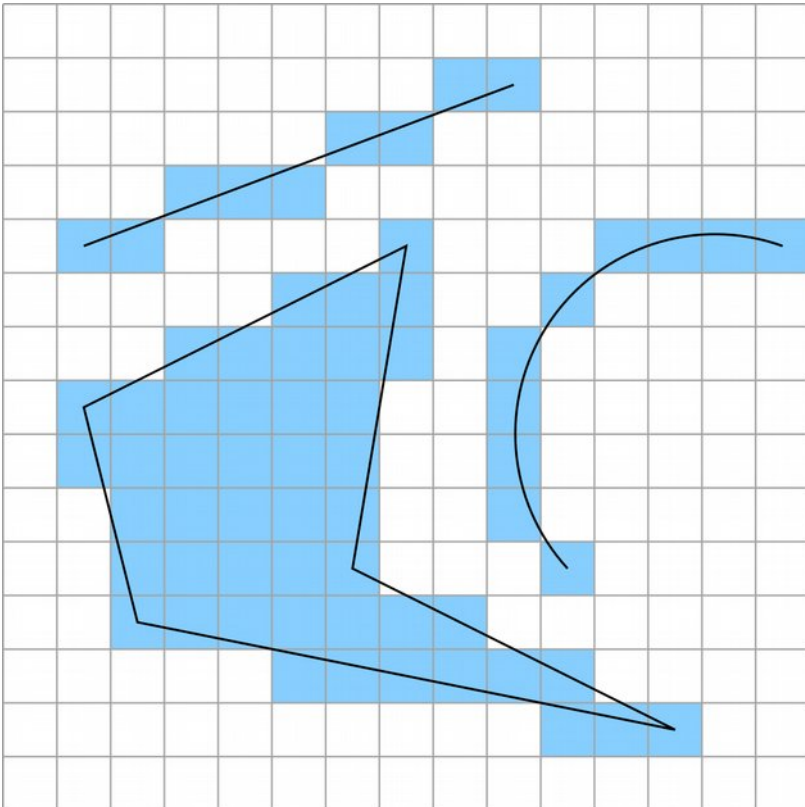
## RASTER DISPLAY

Which PIXELS should be turned on ?



# Rastryzacja

- Zamiana obiektu wektorowego do postaci bitmapy.



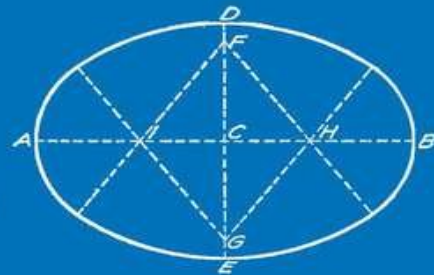
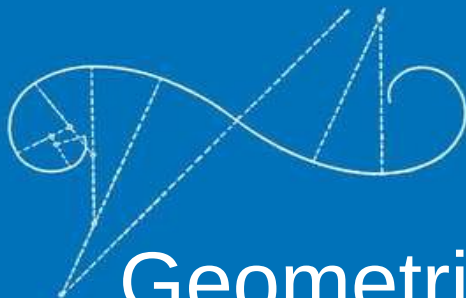
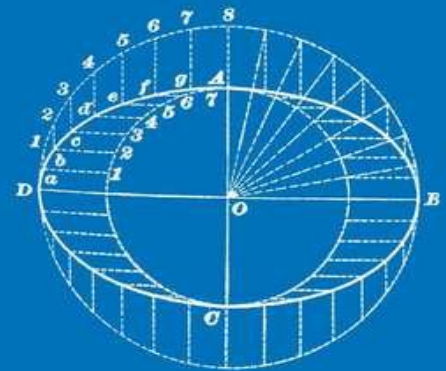
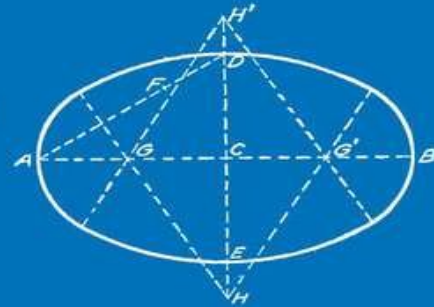
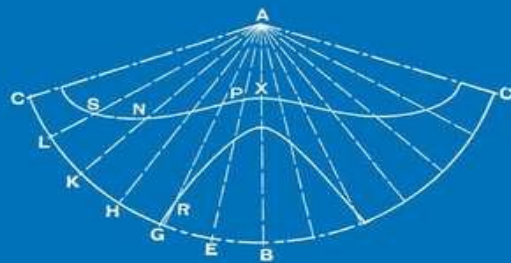
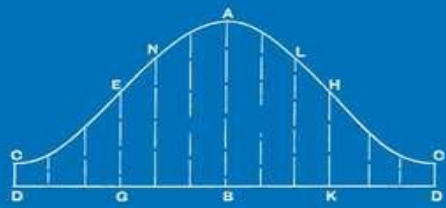
sample

sample

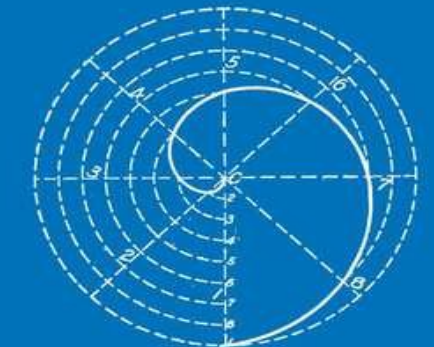
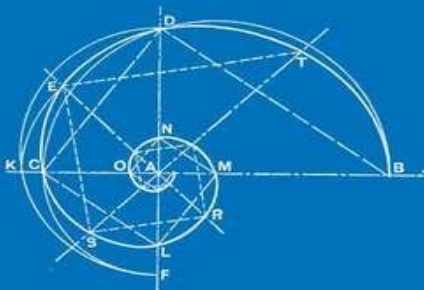
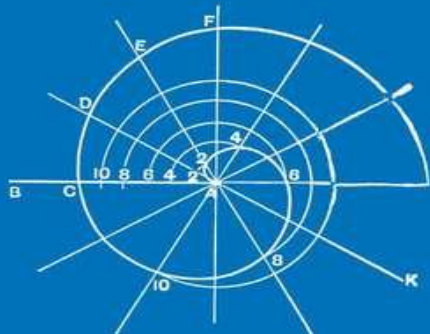
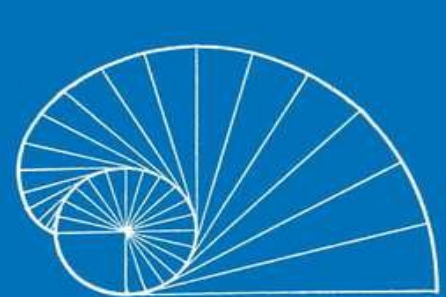
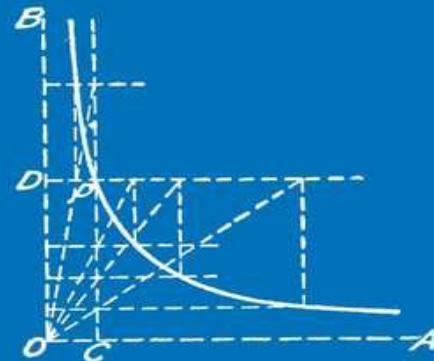
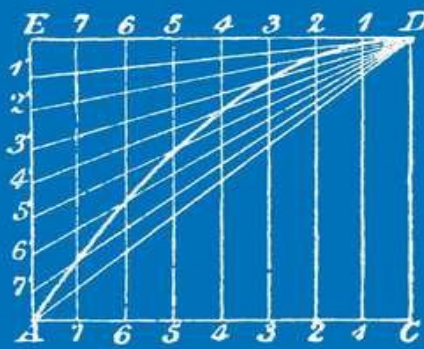
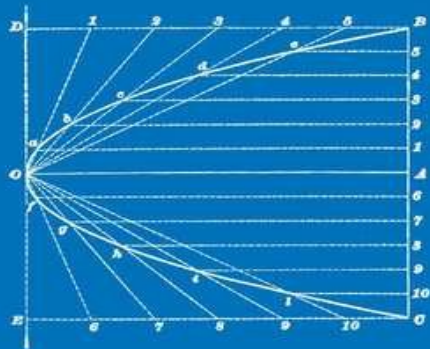
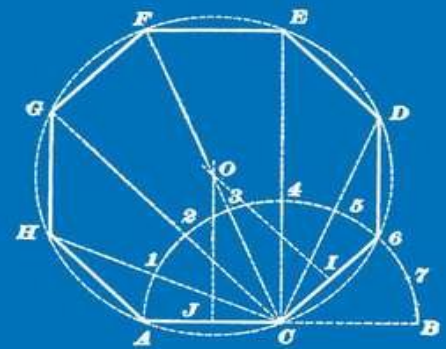
# Tryb szkieletowy (wireframe) i rendering

Rendering – zamiana matematycznego modelu obrazu składającego się z geometrii (szkic), wypełnień, obrysów itp. (w grafice 3D znacznie więcej) na 2D obraz rastrowy





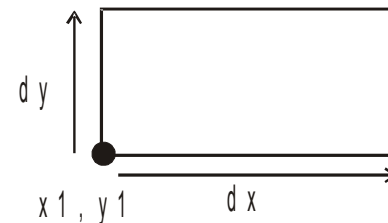
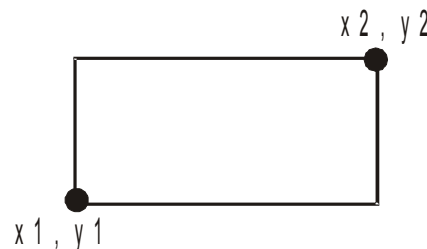
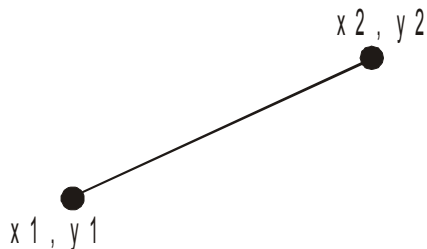
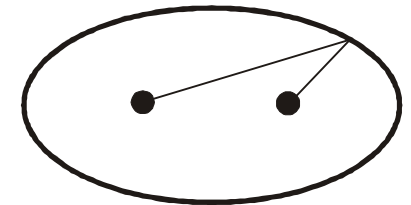
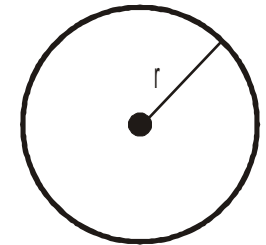
# Geometria obiektów



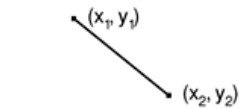
# Obiekty parametryczne

opisane równaniem matematycznym oraz parametrami:  
(Brak jednolitego standardu w programach 2D)

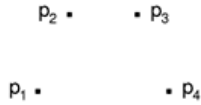
- **Odcinek**
  - Współrzędne jego końców
- **okrąg**
  - współrzędne środka i długość promienia
- **prostokąt**
  - współrzędne dwóch przeciwstawnych rogów
  - współrzędne jednego rogu oraz wymiary boków,
- **elipsa**
  - współrzędne ognisk + prowadnice



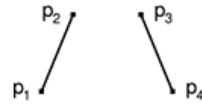
# Obiekty parametryczne



drawLine()



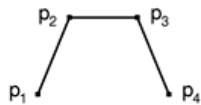
drawPoints()



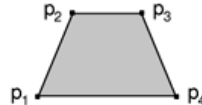
drawLineSegments()



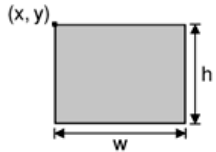
drawCubicBezier()



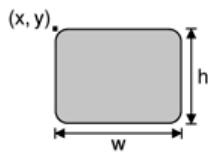
drawPolyline()



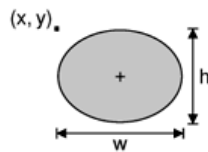
drawPolygon()



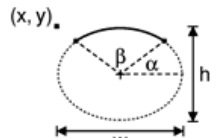
drawRect()



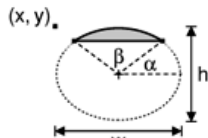
drawRoundRect()



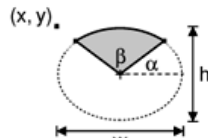
drawEllipse()



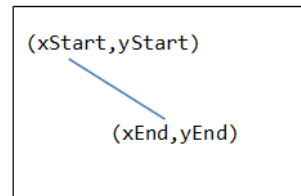
drawArc()



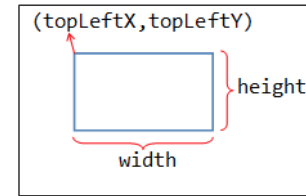
drawChord()



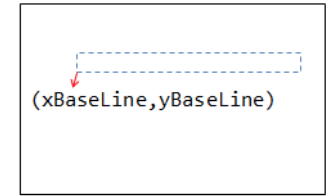
drawPie()



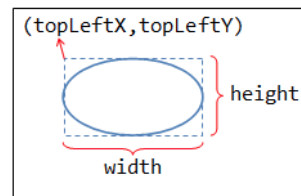
drawLine()



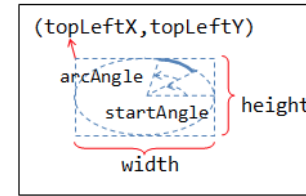
drawRect()



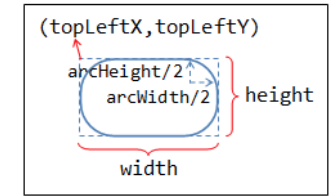
drawString()



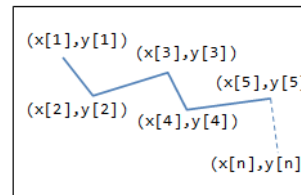
drawOval()



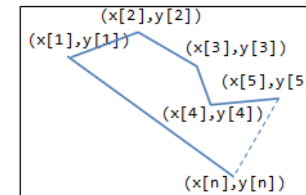
drawArc()



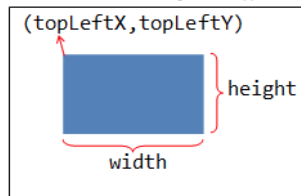
drawRoundRect()



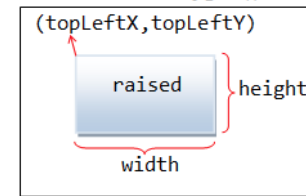
drawPolyline()



drawPolygon()



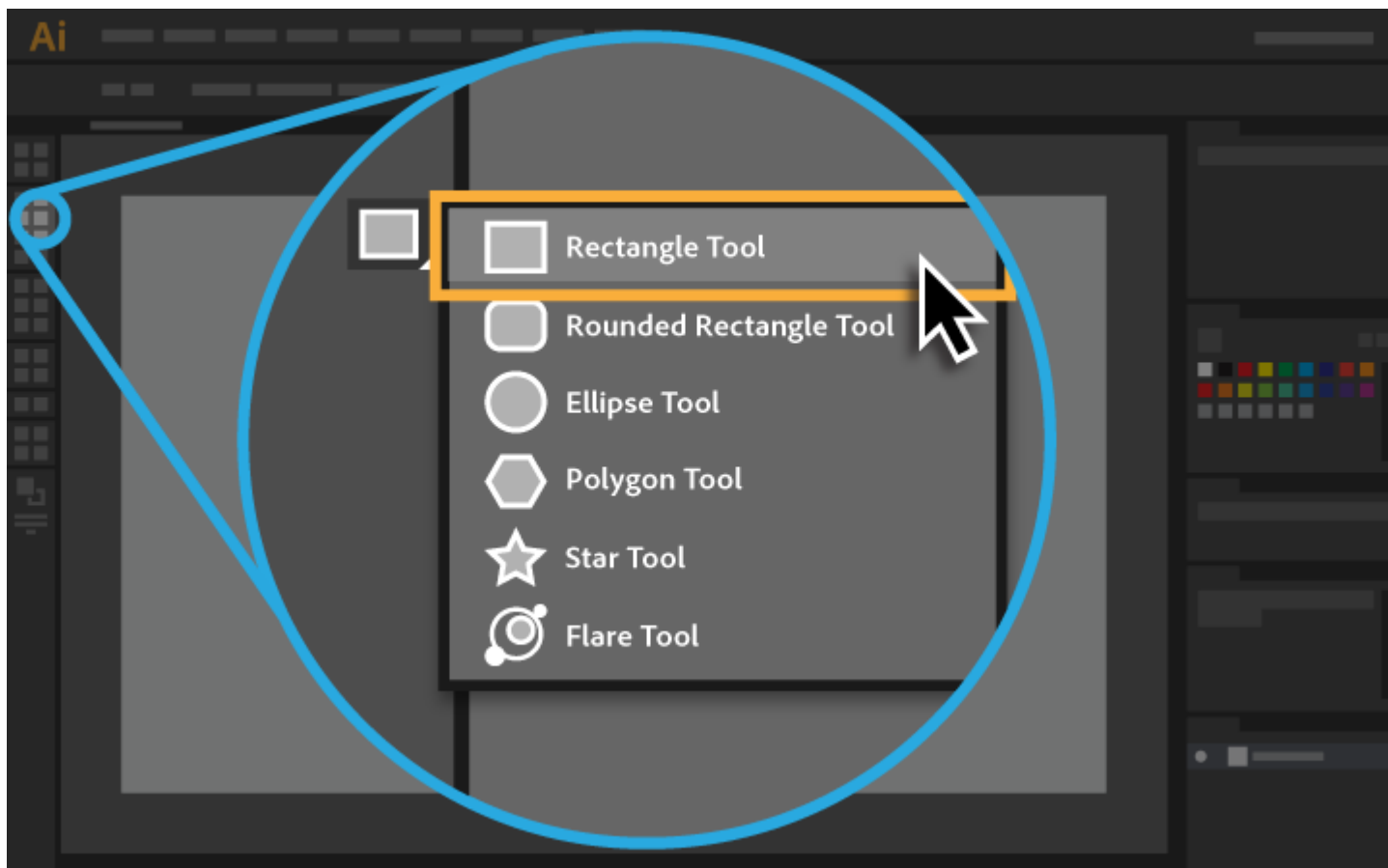
fillRect()



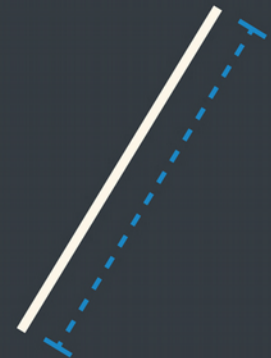
fill3DRect()



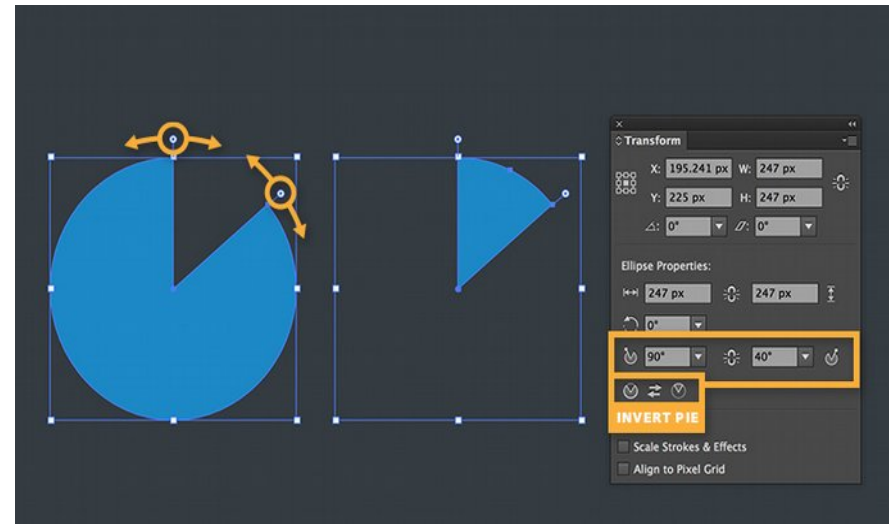
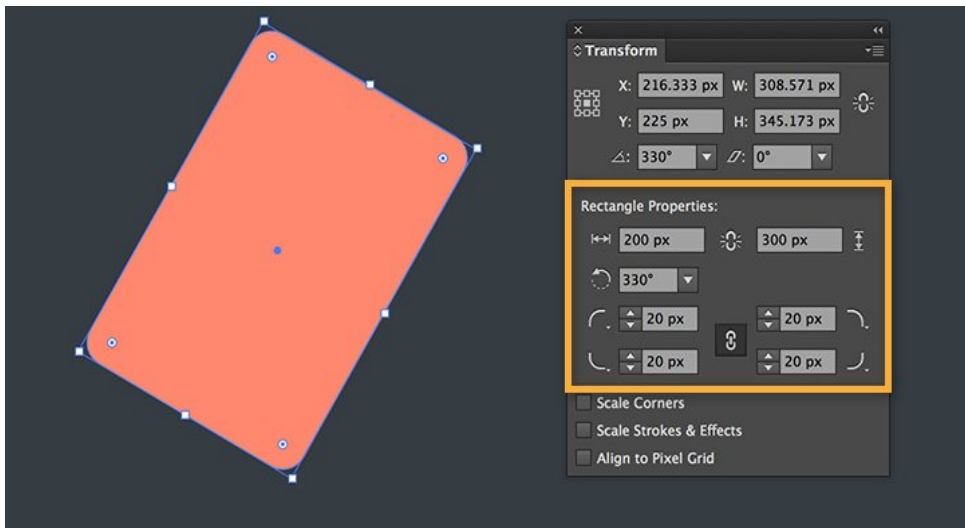
# Brak jednego standardu w programach 2D i 3D



# Obiekty podstawowe pozwalają tworzyć obiekty pochodne

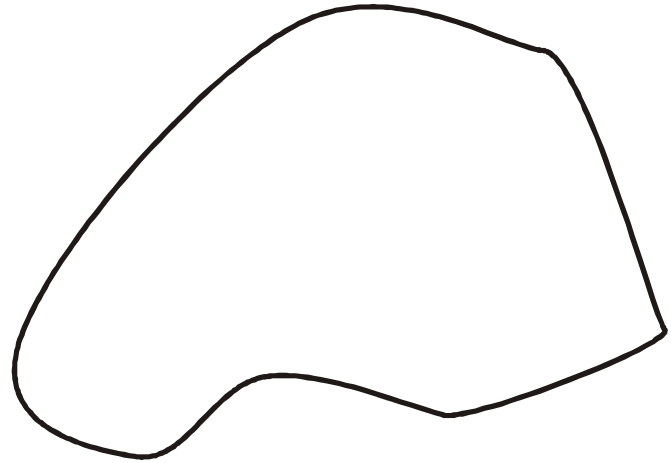
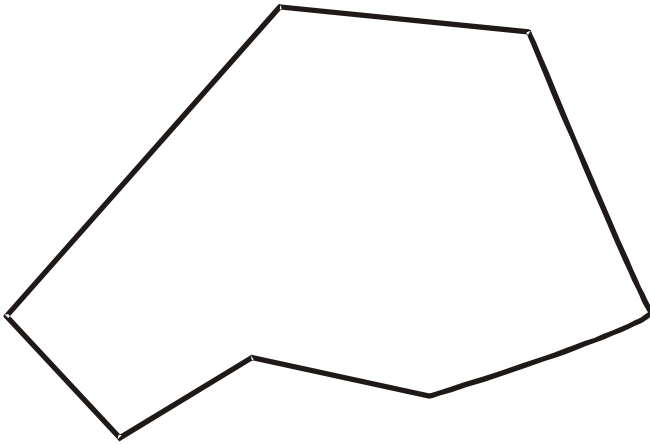


# Dodatkowe parametry obiektów



# Obiekty nieparametryczne

- Są wyrażone albo za pomocą linii łamanej, albo krzywej



# Krzywe

- **Krzywe wielomianowe**

wykorzystuje się krzywe niskich stopni, opisywane niewielką liczbą punktów kontrolnych. Najpowszechniej stosowane są krzywe drugiego stopnia (trzy punkty kontrolne) lub trzeciego (cztery punkty kontrolne);

- **Krzywe wymierne**

można za ich pomocą reprezentować wszystkie krzywe stożkowe, w szczególności okręgi, elipsy i ich wycinki, co ma fundamentalne znaczenie w projektowaniu wspomaganym komputerowo.

- **Krzywe sklejane**

Stosując krzywe wielomianowe i wymierne trudno za pomocą jednej krzywej przedstawiać skomplikowane kształty. Z tego względu powszechnie stosuje się **krzywe B-sklejane (b-spline)**

# Wielomianowe krzywe parametryczne

- Definiują punkty na krzywej 2D za pomocą trzech wielomianów z parametrem  $t$ , odpowiednio dla  $x$ ,  $y$ .
- Współczynniki wielomianów są tak dobierane aby krzywa przebiegała wzdłuż pożądanej ścieżki.
- Można wykorzystać różne stopnie wielomianów
  - najczęściej stosowane są **wielomiany trzeciego stopnia** (występują w nich trzecie potęgi parametru). W odniesieniu do takich krzywych często używane jest określenie krzywa trzeciego stopnia.
  - wielomiany niższego stopnia są zbyt mało elastyczne, jeśli chodzi o sterowanie kształtem krzywej.
  - wielomiany wyższego stopnia wprowadzają niepożądane oscylacje, a ponadto wymagają większej liczby obliczeń.

# Wielomianowe krzywe parametryczne

- Każdy segment  $Q$  całej krzywej jest określony trzema funkcjami  $x, y, z$  które są wielomianami trzeciego stopnia parametru  $t$ .
- Wielomiany trzeciego stopnia, które określają segment krzywej  $Q(t)=[x(t) \ y(t) \ z(t)]^T$  mają postać:

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x$$

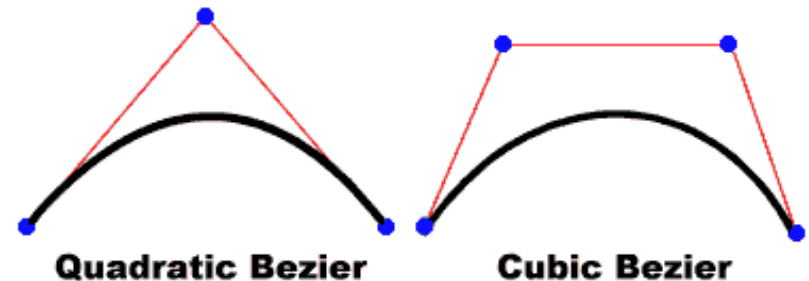
$$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y$$

$$z(t) = a_z t^3 + b_z t^2 + c_z t + d_z$$

Aby operować na skończonych segmentach krzywej parametr  $t$  jest ograniczony do przedziału  $[0,1]$

# Krzywe Béziera

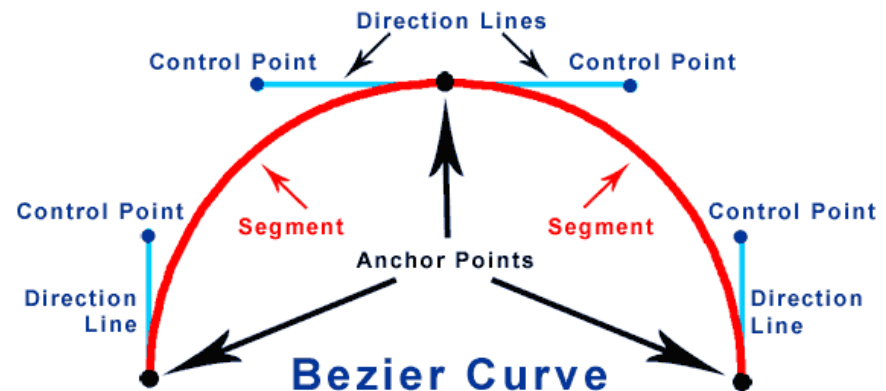
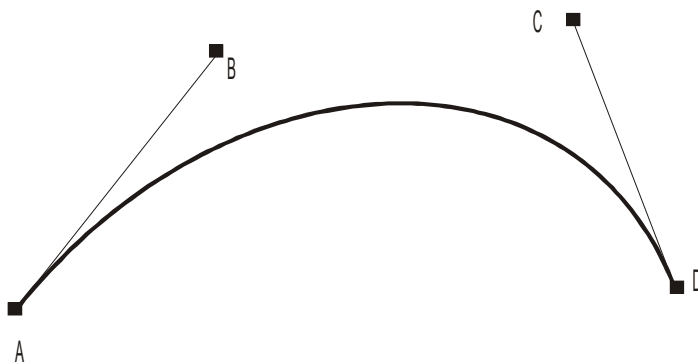
- Parametryczna krzywa powszechnie stosowana w programach do projektowania inżynierskiego CAD (MicroStation), projektowania grafiki komputerowej (Corel Draw, Adobe Illustrator, Inkscape), do reprezentowania kształtów znaków w czcionkach komputerowych (TrueType, METAFONT, Type1) i systemach przetwarzania grafiki (PostScript, MetaPost) oraz w grafice wektorowej (np. format SVG).
- Została opracowana przez Pierre'a Beziera z myślą o wykorzystaniu przy projektowaniu samochodów w firmie Renault.



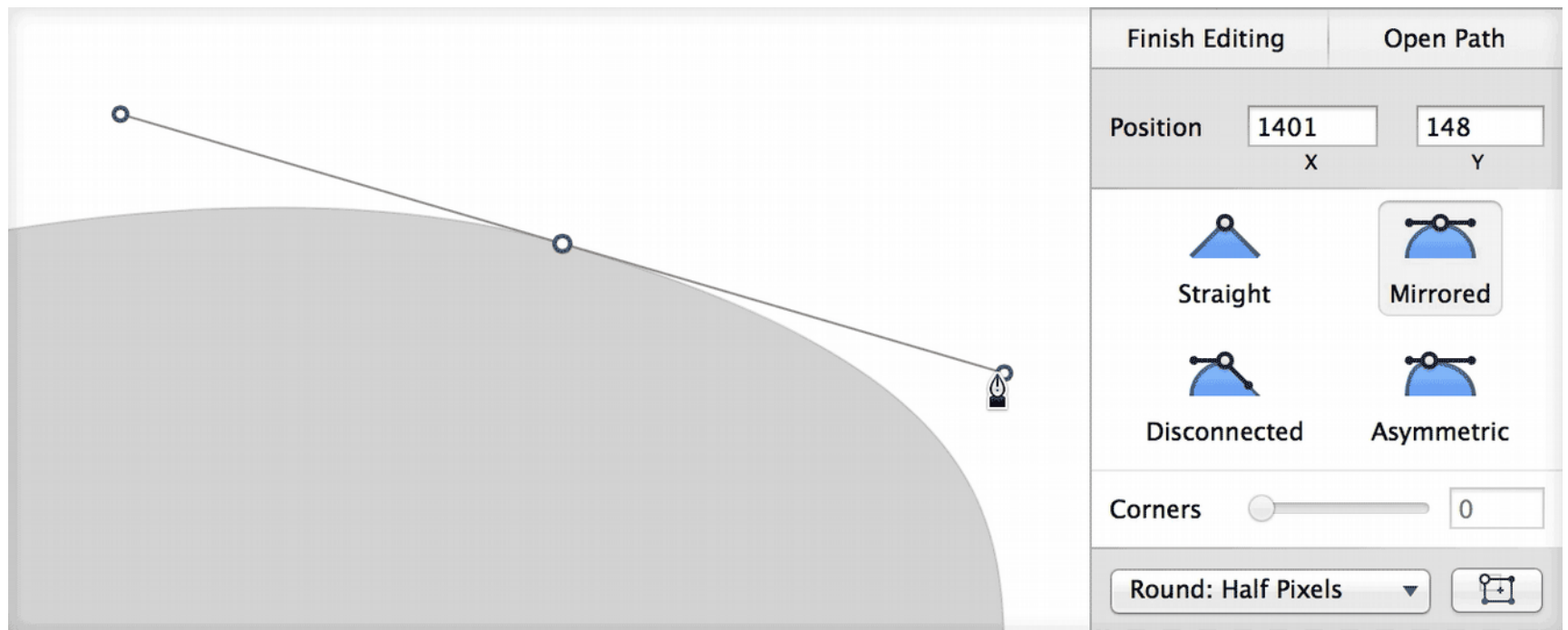


# Krzywa Béziera 3-go stopnia






- Najczęściej używane są krzywe trzeciego stopnia leżące na płaszczyźnie.
- Definiując krzywą trzeciego stopnia określamy 4 punkty A, B, C i D, których położenie wyznacza przebieg krzywej. Krzywa ma swój początek w punkcie A i skierowana jest w stronę punktu B. Następnie zmierza w stronę punktu D dochodząc do niego od strony punktu C. Odcinek AB jest styczny do krzywej w punkcie A, natomiast odcinek CD jest styczny w punkcie D



# Złączenia punktów



The image shows a software interface for connecting points. On the left, a grey curved shape is shown with three points connected by lines. The top-left point is a simple circle. The middle point is a circle with a horizontal line passing through it. The bottom-right point is a circle with a pen nib icon. On the right, a control panel is visible with two tabs: "Finish Editing" and "Open Path". Below the tabs, there are two input fields for "Position" with values "1401" (labeled "X") and "148" (labeled "Y"). There are four icons representing different connection styles: "Straight" (a triangle), "Mirrored" (a circle with a horizontal line), "Disconnected" (a circle with a diagonal line), and "Asymmetric" (a circle with a horizontal line). Below these icons is a "Corners" slider set to "0". At the bottom, there is a dropdown menu set to "Round: Half Pixels" and a small icon of a square with a plus sign.

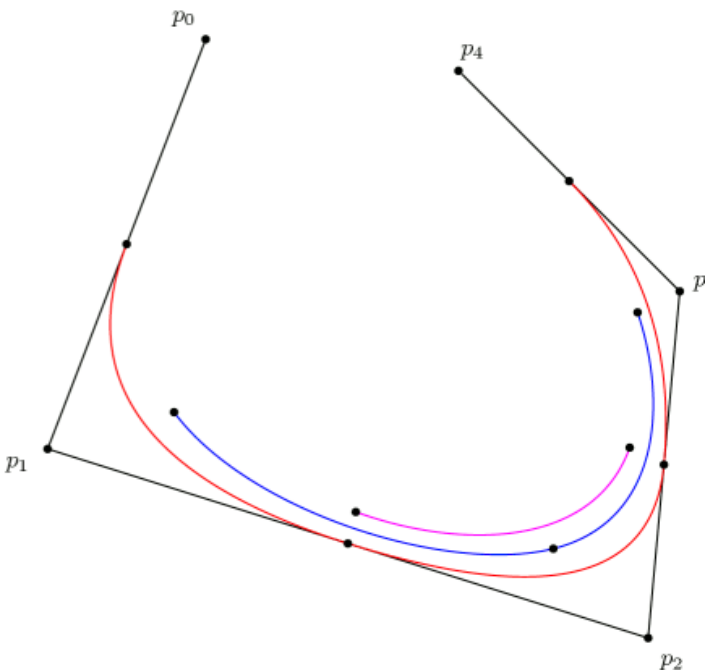
Finish Editing		Open Path	
Position	1401 X	148 Y	
	Straight		Mirrored
	Disconnected		Asymmetric
Corners	<input type="range" value="0"/>		0
Round: Half Pixels			

# Krzywe sklewane

- oferują lokalną kontrolę kształtu – na skutek przemieszczenia jednego punktu kontrolnego zmianie ulegnie tylko jego bliskie otoczenie.
- Krzywe B-sklewane są to takie krzywe, które składają się z fragmentów krzywych bądź to wielomianowych, bądź wymiernych (względnie niskiego stopnia), natomiast matematyczne równania opisujące taką krzywą gwarantują, iż w punktach połączenia różnych fragmentów krzywa będzie gładka.
- Krzywe B-sklewane, podobnie jak inne krzywe parametryczna używane w grafice komputerowej, są wyznaczone przez ciąg punktów kontrolnych
- Rozróżniamy krzywe:
  - Jednorodnie odległości między punktami są stałe
  - Niejednorodnie (NURBS) – odległości między punktami są różne

# Przykład krzywej skleianej n-stopnia

- Jeśli  $n = 1$  wówczas "sklejne" są odcinki, identyczne z łamaną kontrolną krzywej. Dla  $n > 1$  krzywa B-sklejana jest przybliżana kilkoma kawałkami krzywych wielomianowych odpowiednich stopni, połączonych z ciągłością  $C^n$ .

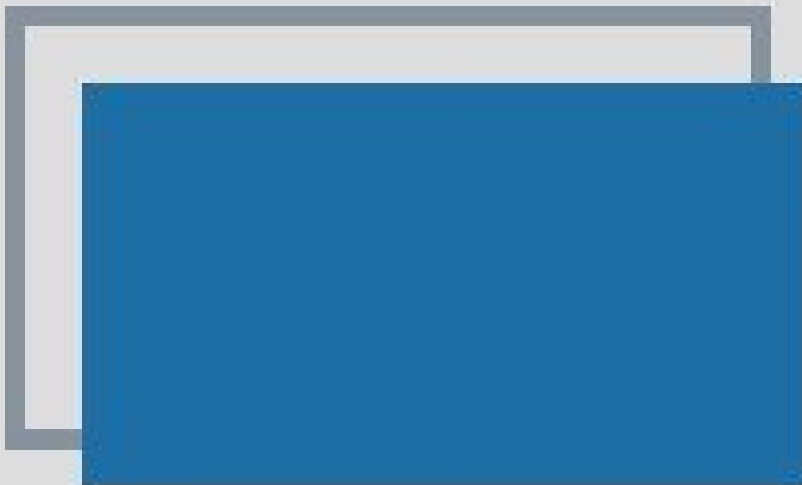


$$p(t) = \sum_{i=0}^{m-n-1} p_i N_i^n(t) \quad \text{dla } t \in [u_n, u_{m-n}),$$

- $m + 1$  — liczba węzłów,
- $n$  — stopień krzywej,
- $p_i$  — punkty kontrolne,
- $N_i^n(t)$  — unormowana funkcja B-sklejana stopnia  $n$ .

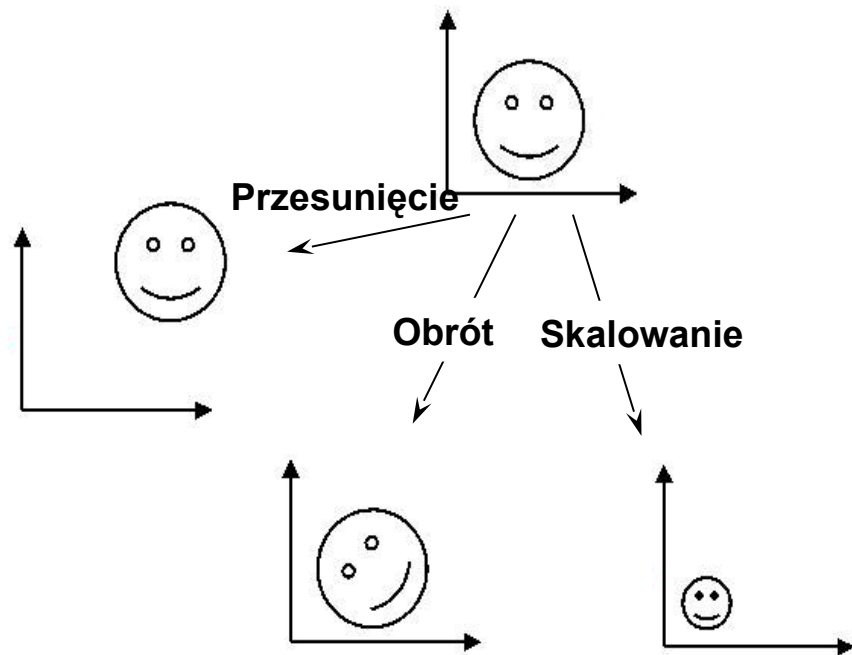


# Przekształcenia geometryczne 2D



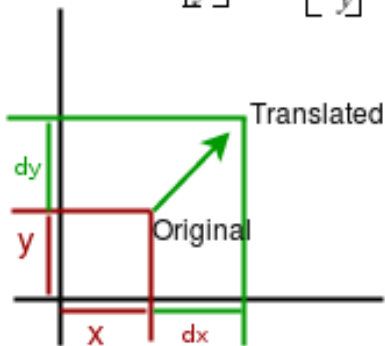
# Przekształcenia geometryczne 2D

- Przesunięcie (translacja)
- Skalowanie
- Obrót
- Przekształcenia złożone



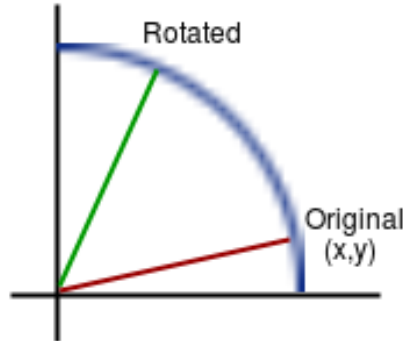
# Podstawy matematyczne przekształceń

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} d_x \\ d_y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$



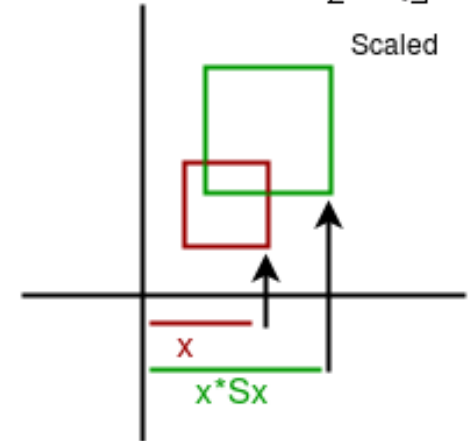
TRANSLATION

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$



ROTATION

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$



SCALING

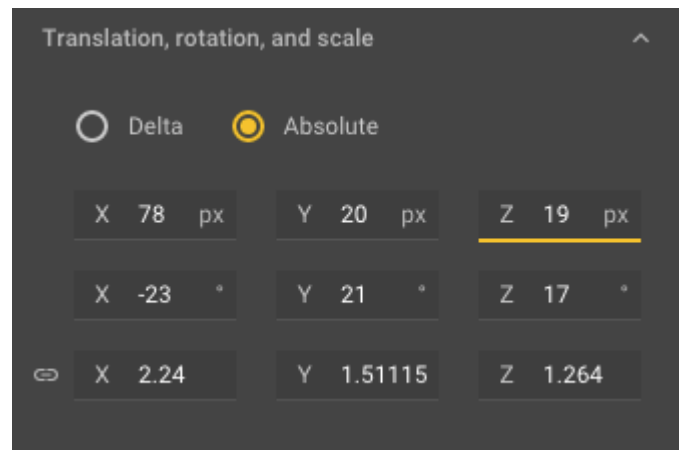
# Transformacje

- Transformacje dotyczą wyłącznie węzłów obiektu. Po transformacji obiekt jest przerysowany
- **Skalowanie i rotacja** odbywa się względem początku układu współrzędnych . Gdy obiekt zmniejsza się jest bliżej początku układu współrzędnych, gdy zwiększa – odwrotnie
- Kąty dodatnie obrotu są mierzone w kierunku przeciwnym do ruchu wskazówek zegara; tj. od x do y.
- **Skalowanie** jednorodne występuje gdy  $s_x=s_y$ ; wtedy proporcje obiektu nie ulegają zmianie. Gdy  $s_x \neq s_y$ , proporcje obiektu ulegają zmianie
- **Rotacja i skalowanie** obiektu w miejscu oznacza przesunięcie na początek układu, transformację a następnie przesunięcie na oryginalne położenie



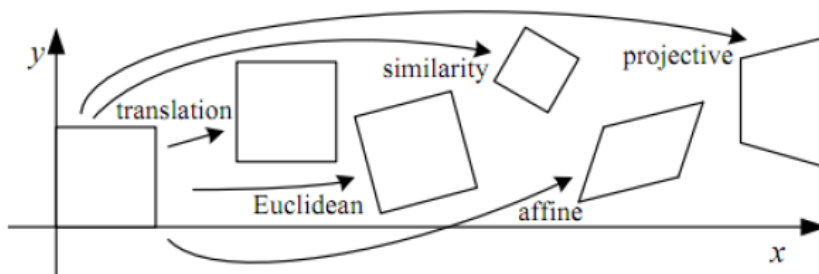
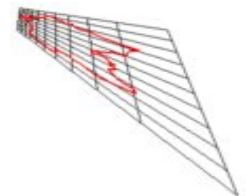
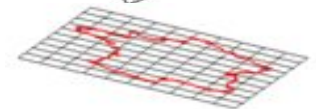
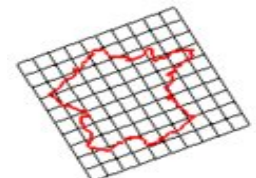
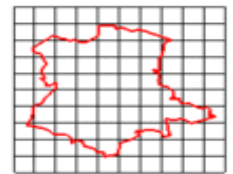
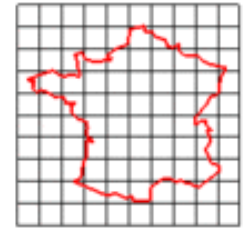
# Transformacje względne i bezwzględne (delta & absolute)

- Transformacje bezwzględne: podaje się nowe wymiary/położenie obiektu
- Transformacje względne: podaje się współczynnik transformacji (np. przesunięcia)



# Hierarchia transformacji

- **Translacja** – transformacja przemieszczająca każdy obiekt geometrii o tę samą odległość
- **Euklidesowe** (sztywne, rigid) – zachowuje odległość euklidesową dla każdej pary punktów. Obejmuje: translację, rotację odbicie oraz ich złożenia
- **Podobieństwa** – zachowuje kształt, odległości nie są wymagane. Obejmuje transformacje euklidesowe oraz skalowanie i ich złożenia
- **Afiniczne** – zachowuje punkty, równoległość linii oraz płaszczyzn (3D). Zachowanie kształtu nie jest wymagane. Obejmuje transformacje podobieństwa, ścinanie oraz ich złożenia
- **Projekcja** – dowolna transformacja reprezentowana przez przemieszczenie 4 punktów do nowego położenia. Obejmuje wszystkie poprzednie transformacje

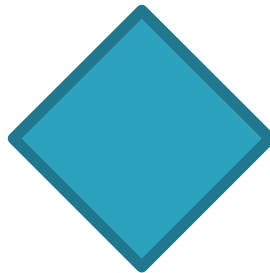


# Przekształcenia a rachunek macierzowy

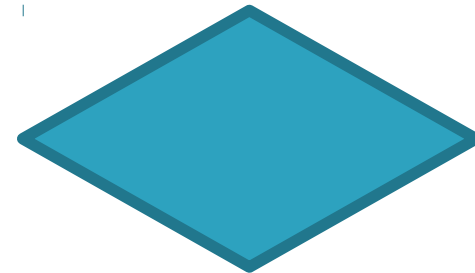
- Iloczyn dowolnej sekwencji macierzy obrotu, przesunięcia i skalowania
  - Mają właściwość zachowania równoległości linii
  - Nie odnosi się to do zachowania długości oraz kątów



Sześciąt jednostkowy



Obrót o  $-45^\circ$

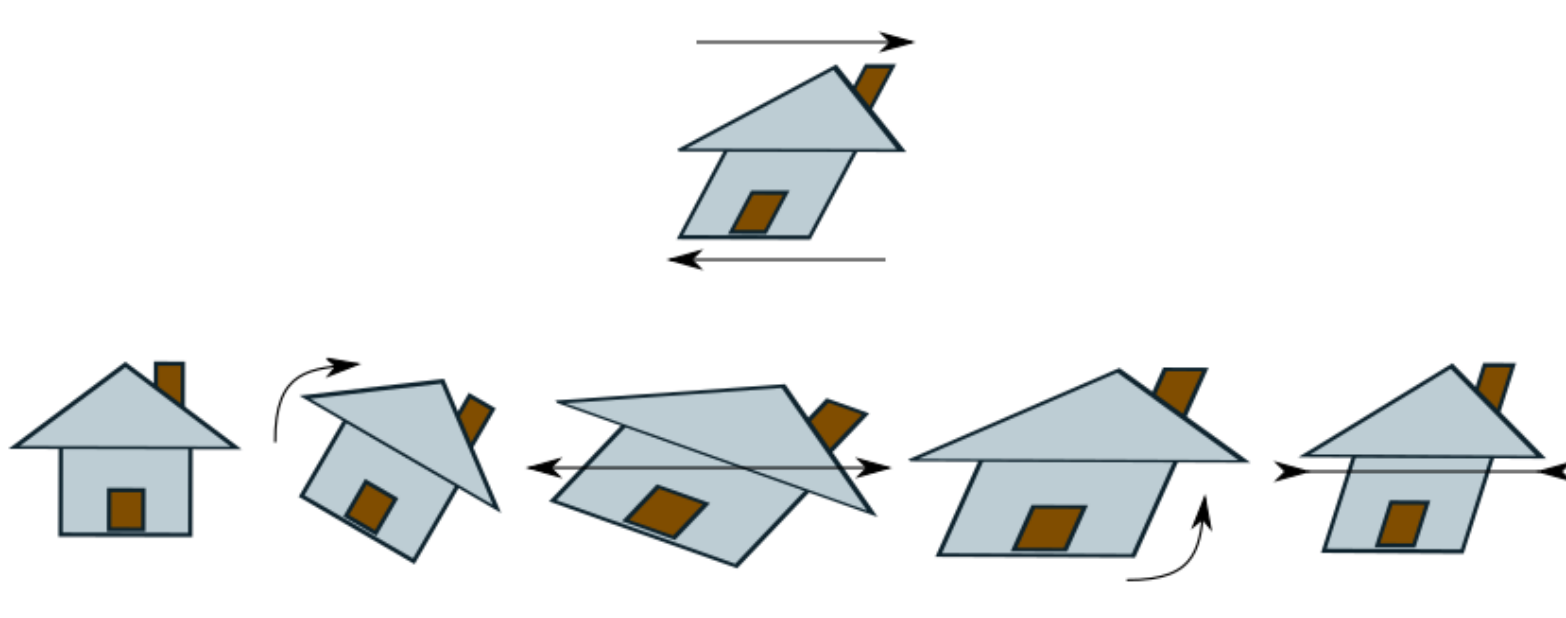


Skalowanie w kierunku osi x  
Brak skalowania w kierunku osi y

Jednostkowy kwadrat jest obracany o  $-45^\circ$  i jest skalowany niejednorodnie. Wynikiem jest afiniczne przekształcenie jednostkowego sześciąt w którym linie równoległe pozostają równoległe, natomiast kąty i długości nie zostają zachowane.

# Ścinanie jako przekształcenie afiniczne

- Łączenie podstawowych macierzy **R**, **S**, **T** w celu uzyskania pożądanego wyniku
- Ścinanie – (shearing) jest przykładem takich operacji



# Hierarchia transformacji

A square transforms to:



Projective  
8dof

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$



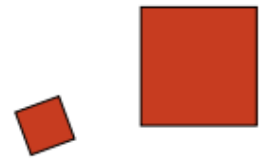
Affine  
6dof

$$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$



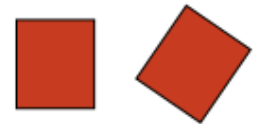
Similarity  
4dof

$$\begin{bmatrix} sr_{11} & sr_{12} & t_x \\ sr_{21} & sr_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

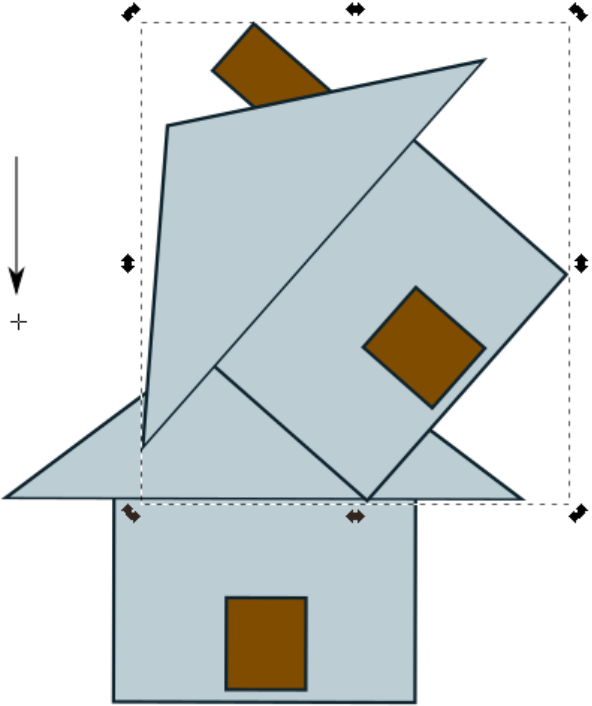
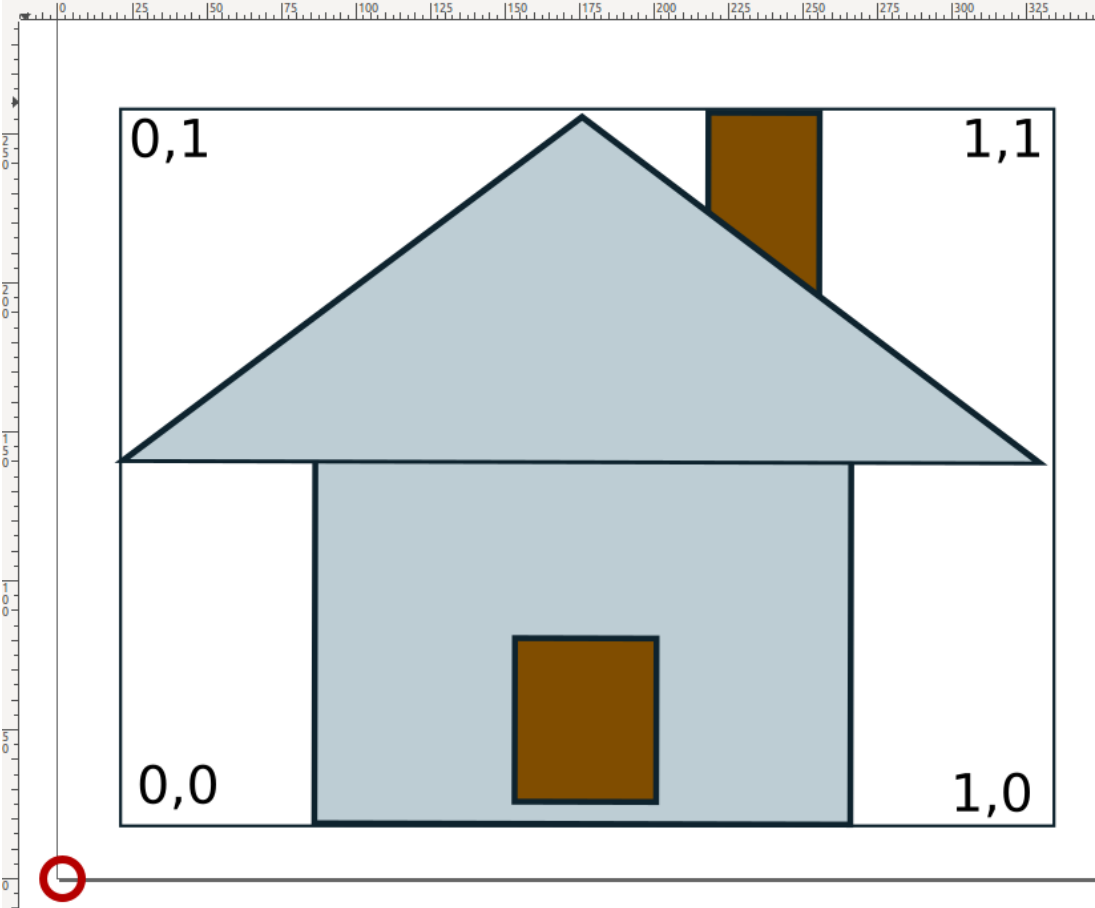


Euclidean  
3dof

$$\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$



# Globalny i lokalny układ współrzędnych



# Zaawansowane operacje wektorowe

- Maskowanie
- Klonowanie wielokrotne
- Perspektywa i obwiednia
- Ścieżki tekstu

# Maskowanie

- Procedura filtrowania danych. W przypadku grafiki wektorowej wskazywany jest obiekt określający, które części obiektu maskowanego pozostają widoczne

**Clip Test**

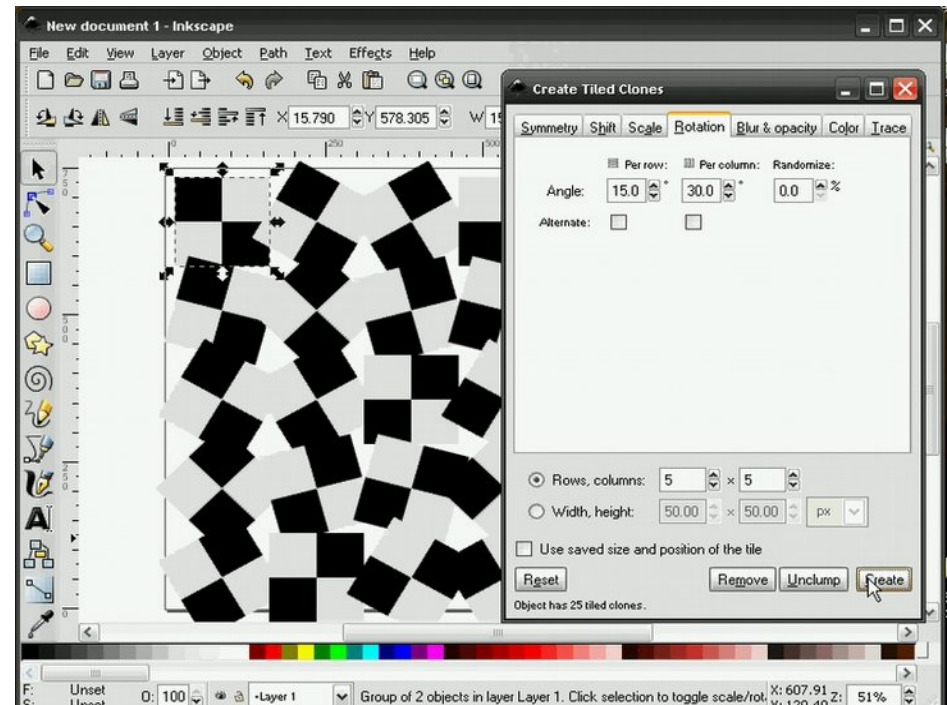
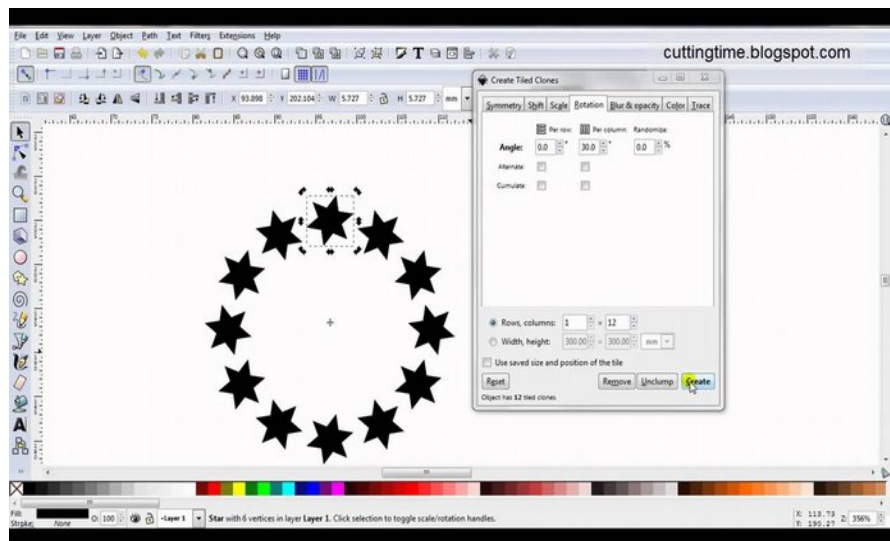
**Clip Test**





# Klonowanie - składanie efektów

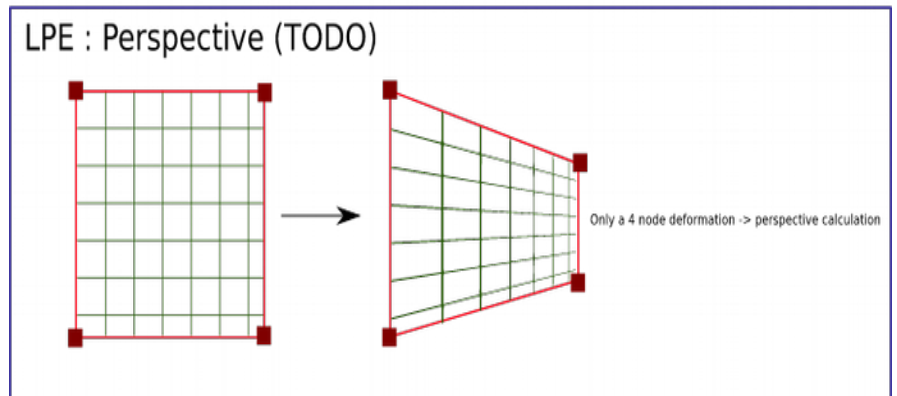
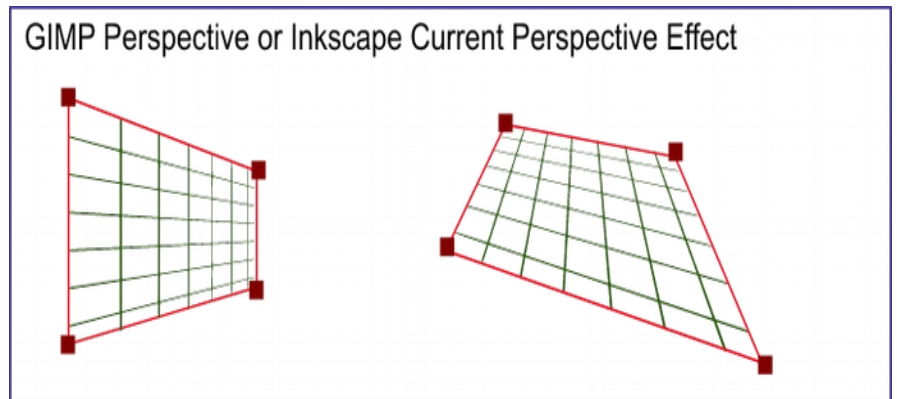
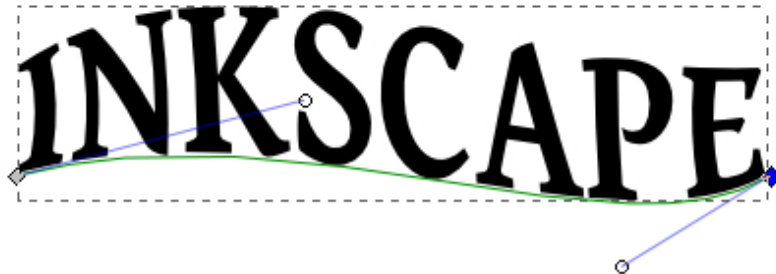
- Operacje wielokrotnego przyrostowego transformowania obiektu. Operacje tworzą klon, a nie kopię obiektu. Klon to referencja do obiektu oryginalnego + lista zastosowanych transformacji



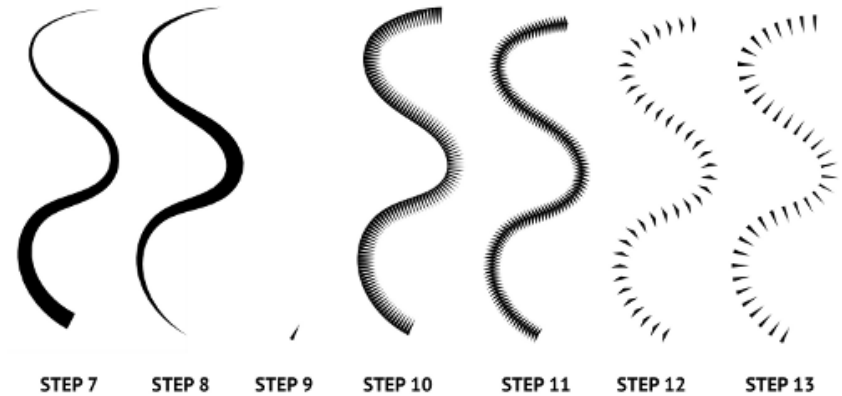
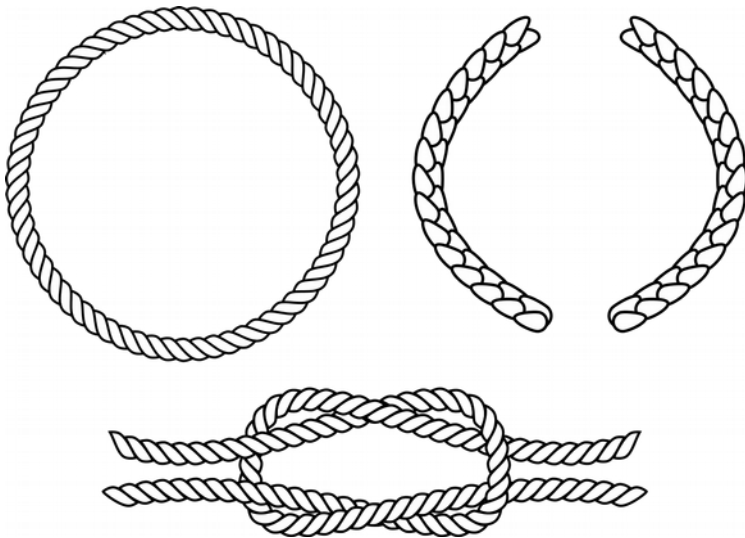
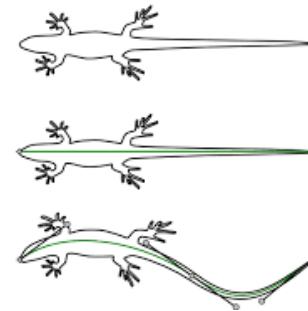
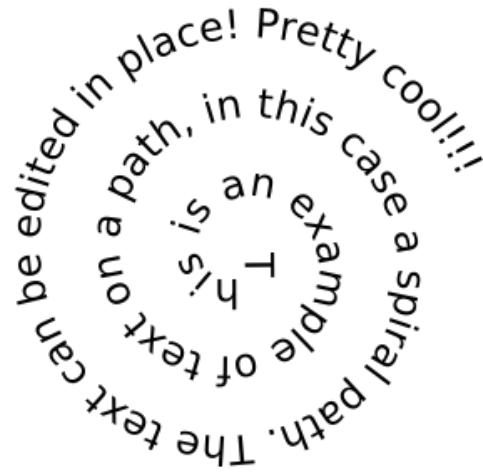
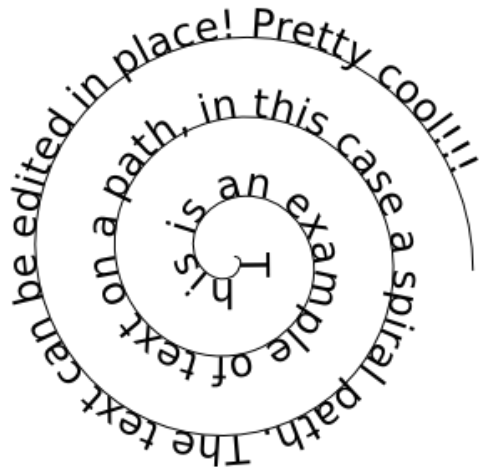
# Perspektywa i obwiednia – efekty ścieżki



**INKSCAPE**



# Różne efekty ścieżki



STEP 7

STEP 8

STEP 9

STEP 10

STEP 11

STEP 12

STEP 13



Typografia

# Typografia

- Typografia to termin związany z użyciem znaków pisarskich w druku, prezentacją ich na ekranie monitora komputerowego itp. Typografia obejmuje:
- Stosowanie czcionek w druku, druk wypukły;
- ogół zagadnień dotyczących projektowania drukowanych liter i innych znaków pisarskich (zarówno czcionek jak i fontów) oraz wzajemnych relacji pomiędzy tymi znakami i grupami znaków;
- układ graficzny tekstu na drukowanej stronie, sztukę użytkową zajmującą się estetyką szaty graficznej tekstu danej publikacji/strony WWW

# Czcionka i font

- Czcionka/font – oryginalnie odnośni się do nośnika pojedynczych znaków pisma drukarskiego
- W postaci cyfrowej są to znaki zaprojektowane w formie obiektów wektorowych lub bitmapowych.
- **Krój pisma** to obraz kompletu znaków pisma o jednolitych i spójnych cechach graficznych: stylu, rytmie, proporcji, kształcie szeryfów itp. np. “Lato”. Często występuje w wielu odmianach, czasami nawet znacznie różniących się od kroju podstawowego, jednak zawsze zachowujących w sposób konsekwentny główne założenia graficzne danej rodziny krojów.

# Rodzina krojów pisma:

Lato

---

rodzina krojów pisma

thin light **normal bold ultrabold**  
italic: *thin light normal bold ultrabold*

---

10 odmian w rodzinie Lato

# Klasyfikacja krojów

- Kroje szeryfowe
- Kroje bezszeryfowe

SZERYFOWA  
Times New Roman

ABC



SZERYFY

BEZSZERYFOWA  
Arial

ABC



# Kroje szeryfowe

Times New Roman

EB Garamond

Georgia

Cambria

Didot

Merriweather

# Kroje bezszeryfowe

Arial

Helvetica Neue

Lato

Lucida Sans

Roboto

Proxima Nova

# Czcionki monotypiczne i konsolowe

- Czcionki, gdzie każdy znak zajmuje taką samą szerokość. Mogą być zarówno szeryfowe (maszyna do pisania) i bezszeryfowe (konsolowe)

aiw

iwa

**aiw**

**iwa**

# Kerning i tracking; interlinia

- Tracking – gęstość składu – regulowanie odległości międzyliterowych
- Kerning – regulowanie odległości pomiędzy **konkretnymi parami** znaków w danym kroju pisma

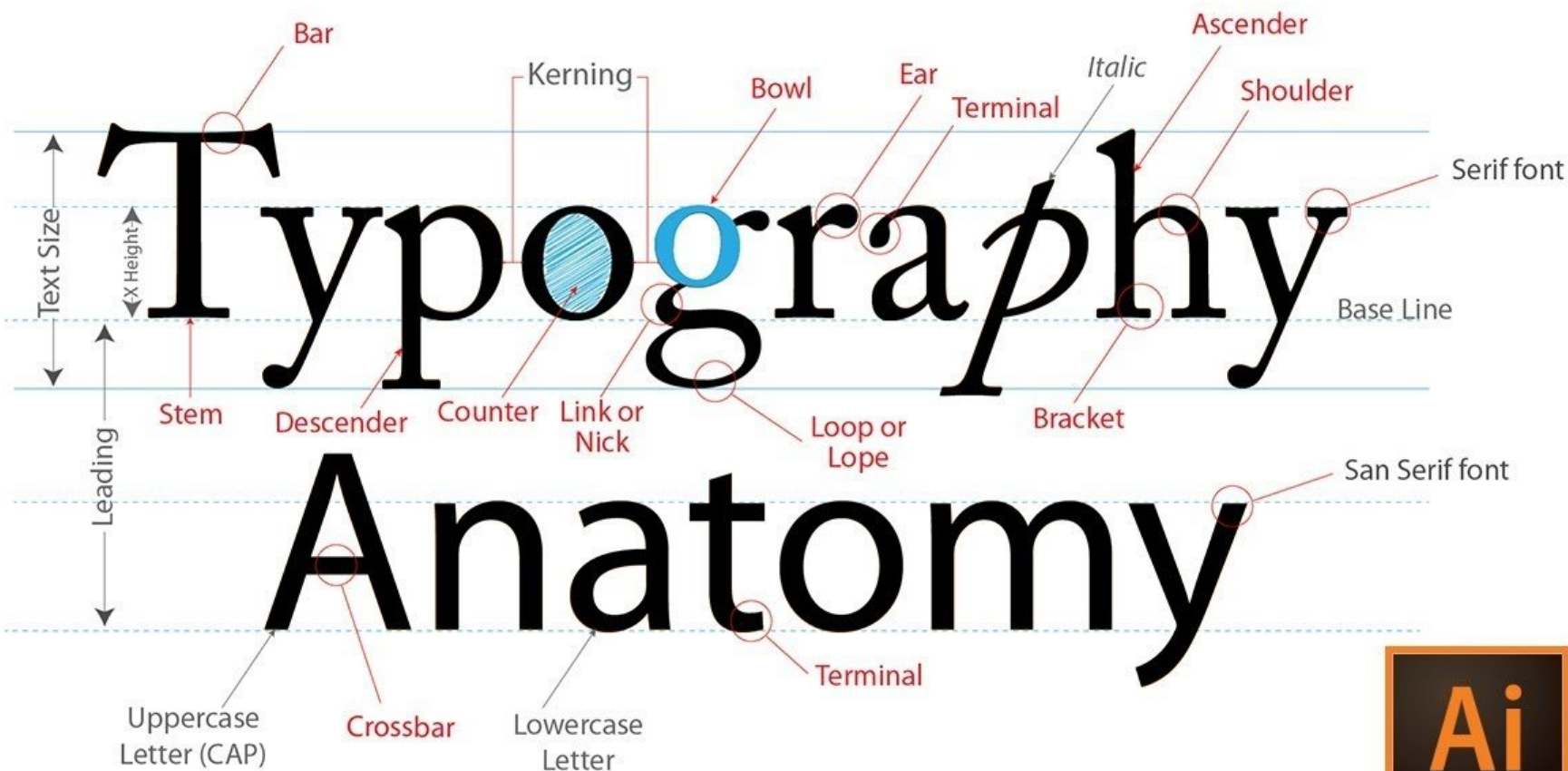


# Interlinia

- Interlinia - odległość między sąsiednimi wierszami tekstu liczona jako odstęp pomiędzy dolną linią pisma w wierszu górnym i dolną linią pisma w wierszu dolnym

99designs is a large,  
global community  
that brings together  
lots of different people.

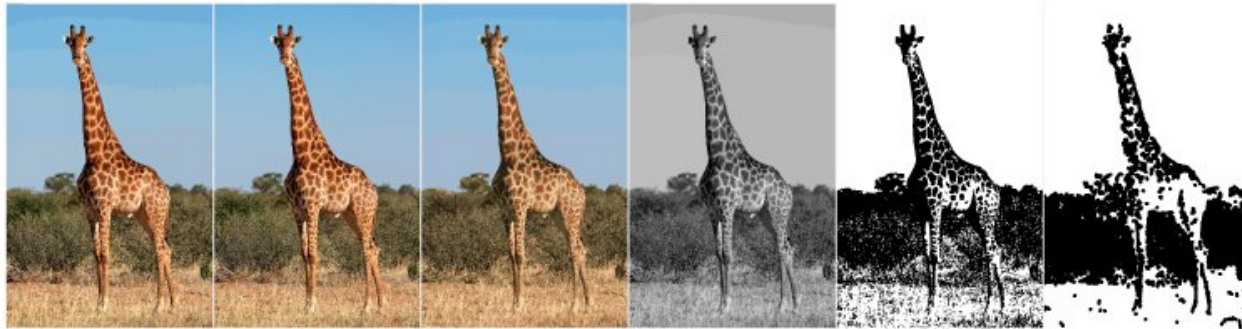




# Wektoryzacja

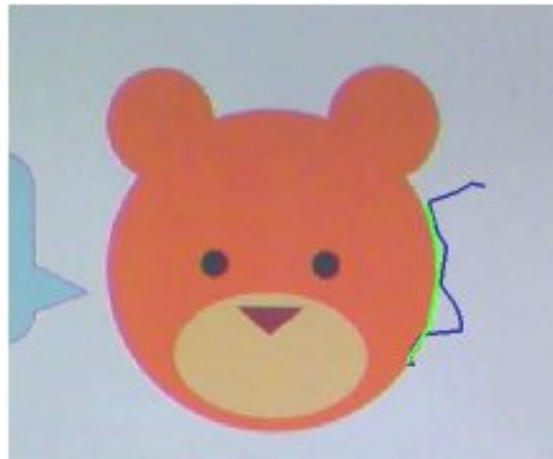
- Inaczej: trasowanie bitmap.
- Bitmapa nie posiada żadnej struktury – jest tylko macierzą wartości. Proces wektoryzacji to zamiana macierzy do reprezentacji obiektowej
- Ręczna – trasowanie rysunku, dokładna ale wolna
- Automatyczna – zastosowanie algorytmów znajdujących obiekty na obrazie. Jakość zależna od jakości algorytmów
- Półautomatyczna – automatyzacja ograniczona do znajdowania krawędzi, zachowuje kontrolę nad całym procesem
- OCR – optyczne rozpoznawanie tekstu – połączone z algorytmami uczenia maszynowego

# Proces automatyczny





# Proces półautomatyczny – wykrywanie krawędzi



# GIS

## Types of snapping

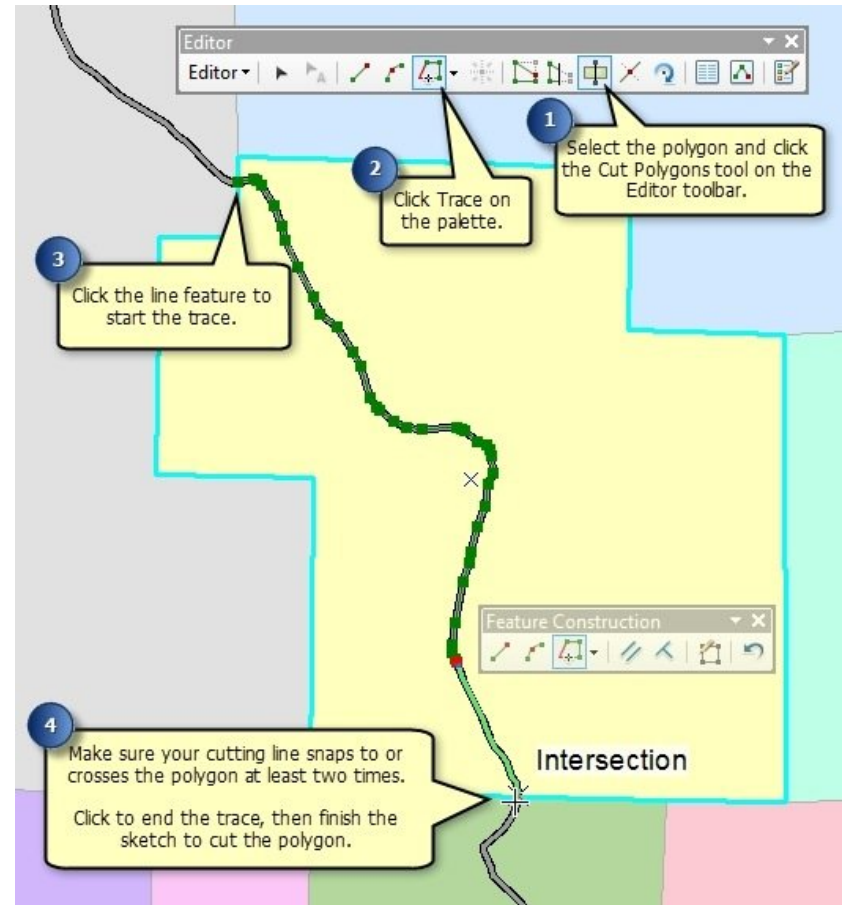
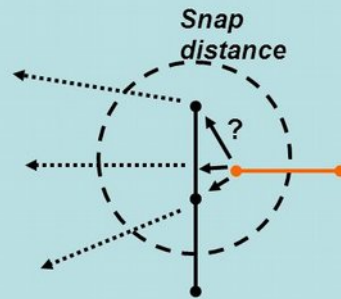
End snapping



Edge snapping



Vertex snapping



# Formaty grafiki 2D

# Windows Graphic Metafile

- Format graficzny opracowany jako przenośny pomiędzy aplikacjami (w praktyce MS OFFICE), zawierający grafikę wektorową i bitmapy. Działa na zasadzie wywoływania funkcji rysujących z GDI (Graphic Device Interface). Ze względu na silną integrację z własnościowymi technologiami MS ograniczona używalność poza środowiskiem OFFICE.

# POSTSCRIPT

- Język komputerowy opisu strony używany do opisu grafiki wektorowej, proceduralny. Obecnie obowiązuje wersja 2 i 3. Skomplikowany i powolny w użyciu. W formie bezpośredniej używany przez profesjonalne drukarki postsriptowe. Obsługuje własny system czcionek (postscript, Type1).  
Rozszerzenia .ps
- Format **.eps** jest jednym z standardów wymiany danych między aplikacjami graficznymi. Jest to format postscript z dodatkowymi ograniczeniami, aby mógł być używany do przechowywania grafiki w taki sposób aby mogła być używana w innych plikach postscript. Najważniejsza różnica to definicja bounding box.

```
%!PS
/Courier          % name the desired font
20 selectfont    % choose the size in points and establish
                 % the font as the current one
72 500 moveto    % position the current point at
                 % coordinates 72, 500 (the origin is at the
                 % lower-left corner of the page)
(Hello world!) show % stroke the text in parentheses
showpage        % print all on the page
```

# Scalable Vector Graphics

- Otwarty format zalecany przez konsorcjum W3C, obsługujący interaktywność i animacje. Zdefiniowany jako tekstowy plik XML, obsługiwany przez współczesne przeglądarki. Rozszerzenie .svg. Obsługiwany przez wiele programów graficznych. Dedykowany jest Inkscape. Ze względu na tekstowy opis zawiera liczne powtórzenia tekstu (markups) z tego powodu w praktyce używa się formatu .svgz, który jest plikiem svg spakowanym programem gzip. Obecny standard 1.1, istnieją uproszczone standardy SVG Basic i SVG Tiny.



```
<svg xmlns="http://www.w3.org/2000/svg" version="1.1">  
  <rect width="300" height="100" style="fill:rgb(0,0,255);stroke-width:1;stroke:rgb(0,0,0)" />  
</svg>
```

# Portable Document Format **PDF**

- Format przeznaczony do **publikacji** dokumentów, niezależny od urządzenia, systemu, oprogramowania. Zawiera pełny opis układu (layout) spłaszczonego dokumentu, zawiera tekst, czcionki, grafiki, i inne informacje niezbędne do prawidłowego wyświetlenia. Od 2008 otwarty standard wg normy ISO 32000-1:2008. Opis strony w PDF jest oparty na postscript, ale uproszczony o elementy kontrolujące przepływ danych na drukarkę.
- Posiada rozszerzenia związane z edycją i elektronicznym obiegiem dokumentów

# Formaty własnościowe

Formaty własnościowe, opatentowane, z tego powodu są problemy z przenoszeniem plików pomiędzy aplikacjami

- Cdr – Corel Draw
- Ai – Adobe Illustrator
- SWF – Shockwave flash (zanika)
- DWG - autocad





Grafika prezentacyjna



# Grafika prezentacyjna

- **Uproszczona** grafika wektorowa, przeznaczona głównie do **wizualizacji** danych i/lub procesów, struktur itp.
  - Wykresy – **złożone (compound)** obiekty parametryczne
  - Diagramy i modele
  - Mapy, kartogramy, kartodiagramy
  - Wizualizacje naukowe
  - Inne: grafy, chmury tagów
- Do grafiki prezentacyjnej istnieje wiele narzędzi: od programów biurowych, poprzez narzędzia statystyczne po specjalizowane języki analizy danych

# Narzędzia grafiki prezentacyjnej

The collage shows various tools used for data visualization and analysis:

- Spreadsheet:** A Microsoft Excel window showing a table with columns for years (2010, 2011, 2012) and rows for countries (Denmark, Germany, Sweden). A bar chart titled "Development over time" is overlaid on the data.
- Statistical Software:** A STATISTICA window displaying a table of data with columns for various variables and a histogram showing the distribution of one variable.
- Mathematical Software:** A window titled "MATLAB CODE" showing a script for simulating a trajectory. The code includes comments in Polish and Python-like syntax for plotting.
- Bar Chart:** A separate bar chart titled "Scores by group and gender" showing scores for five groups (G1-G5) for Men (red bars) and Women (yellow bars). Error bars are present for each bar.

At the bottom, a large arrow points from left to right, indicating the direction of increasing complexity and effort.

Wzrost możliwości i nakładu pracy

```
import numpy as np
import matplotlib.pyplot as plt

N = 5
men_means = (20, 35, 30, 35, 27)
men_std = (2, 3, 4, 1, 2)

ind = np.arange(N) # the x location
width = 0.35 # the width of th

fig, ax = plt.subplots()
rects1 = ax.bar(ind, men_means, width, color='r', yerr=men_std)

women_means = (25, 32, 34, 20, 25)
women_std = (3, 5, 2, 3, 3)
rects2 = ax.bar(ind + width, women_means, width, color='y', yerr=women_std)

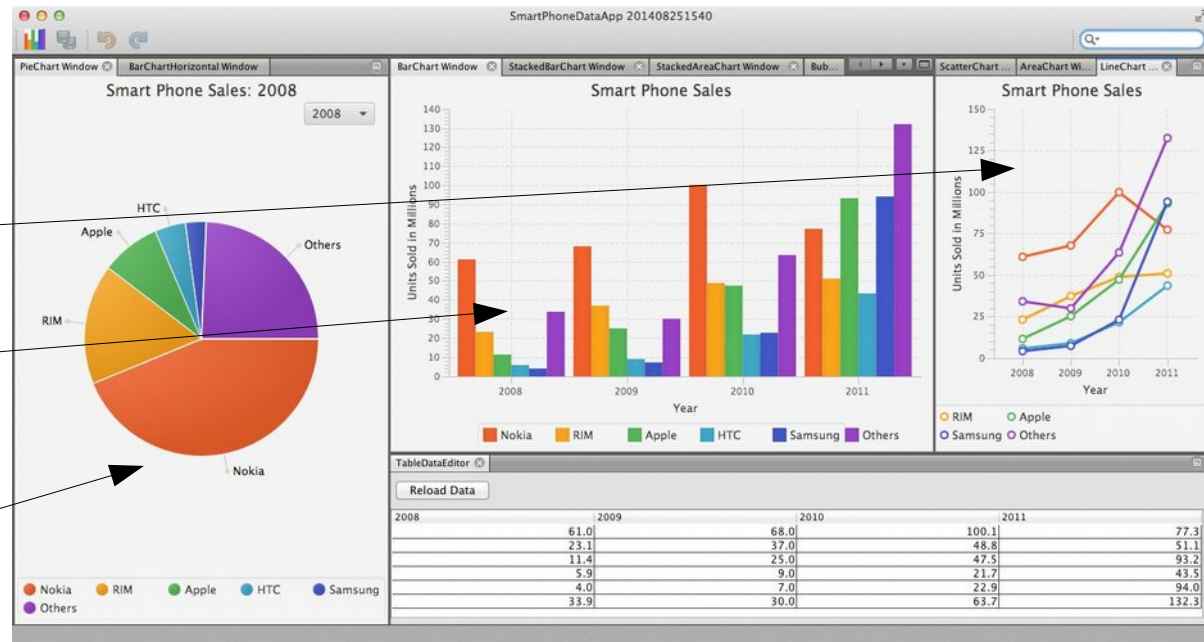
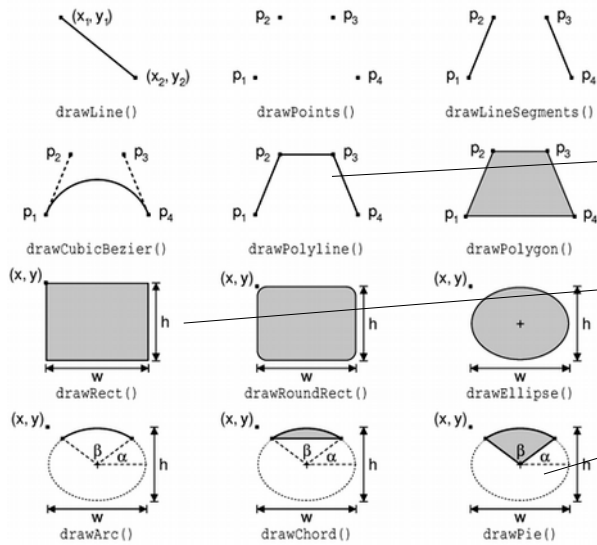
# add some text for labels, title and axes ticks
ax.set_ylabel('Scores')
ax.set_title('Scores by group and gender')
ax.set_xticks(ind + width / 2)
ax.set_xticklabels(('G1', 'G2', 'G3', 'G4', 'G5'))

ax.legend((rects1[0], rects2[0]), ('Men', 'Women'))
```

# Obiekty nisko- i wysokopoziomowe

- Obiekty niskopoziomowe – prymitywy: elementy, z których zbudowane są obiekty wysokopoziomowe. Najczęściej, punkty, linie, poligony, prostokąty, okręgi, krzywe spline, tekst
- Obiekty wysokopoziomowe – gotowe narzędzia prezentacyjne (wykresy, diagramy itp.) zbudowane z prymitywów i sterowane danymi, które mają prezentować

# Obiekty nisko- i wysokopoziomowe

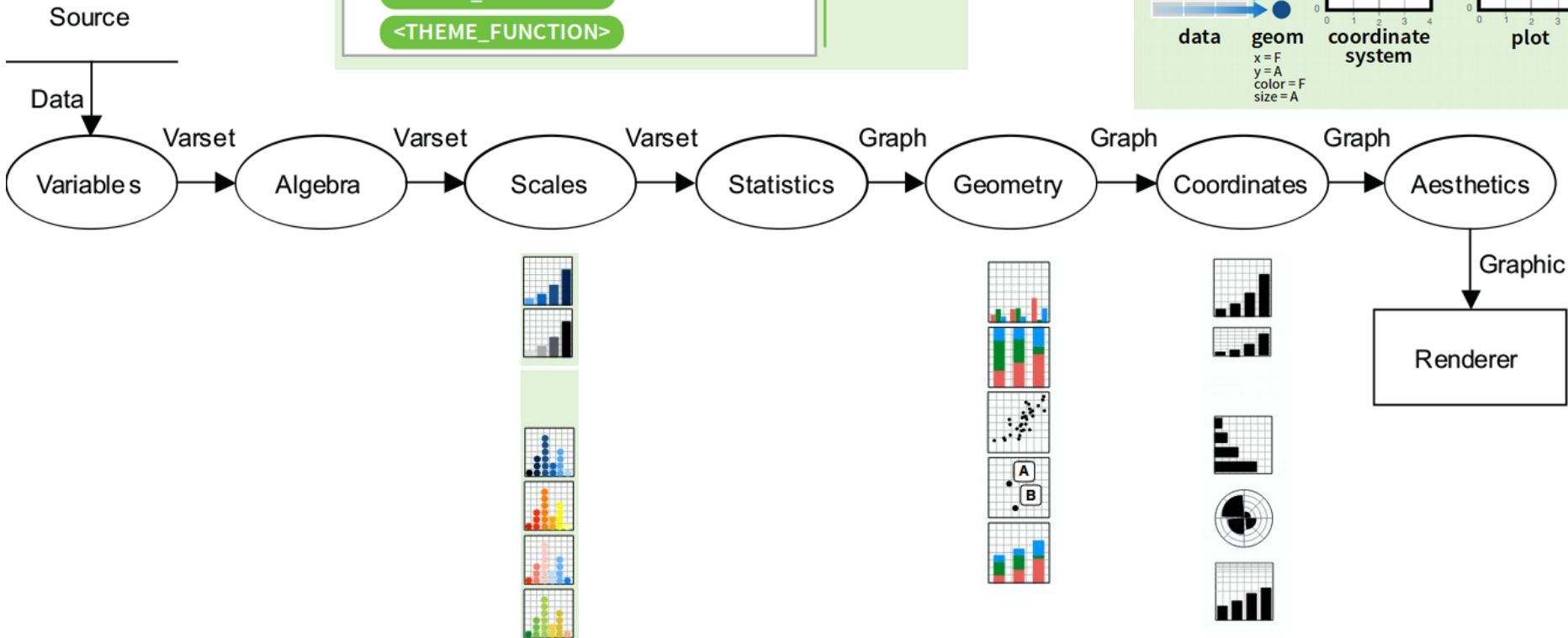
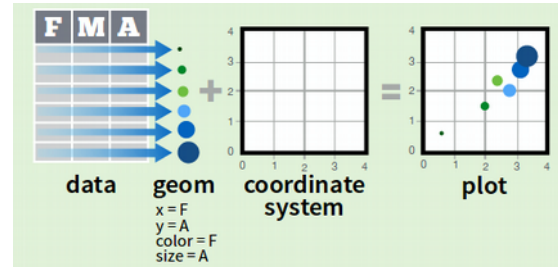


# Gramatyka grafiki prezentacyjnej

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION> (  
    mapping = aes(<MAPPINGS>),  
    stat = <STAT> ,  
    position = <POSITION>  
  ) +  
  <COORDINATE_FUNCTION> +  
  <FACET_FUNCTION> +  
  <SCALE_FUNCTION> +  
  <THEME_FUNCTION>
```

Required

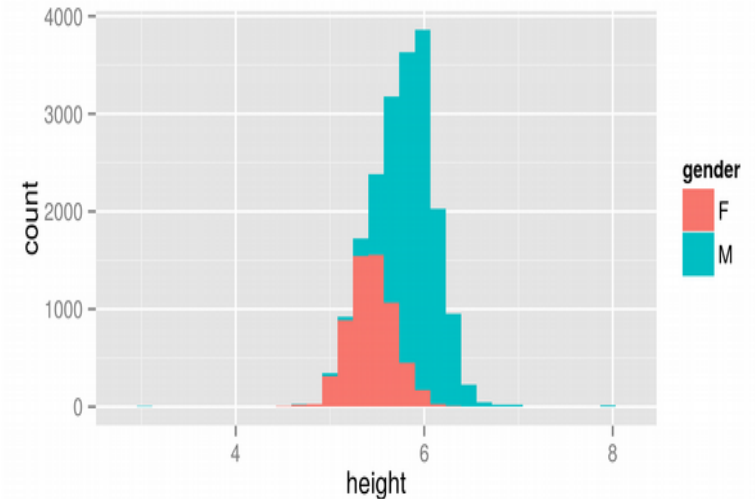
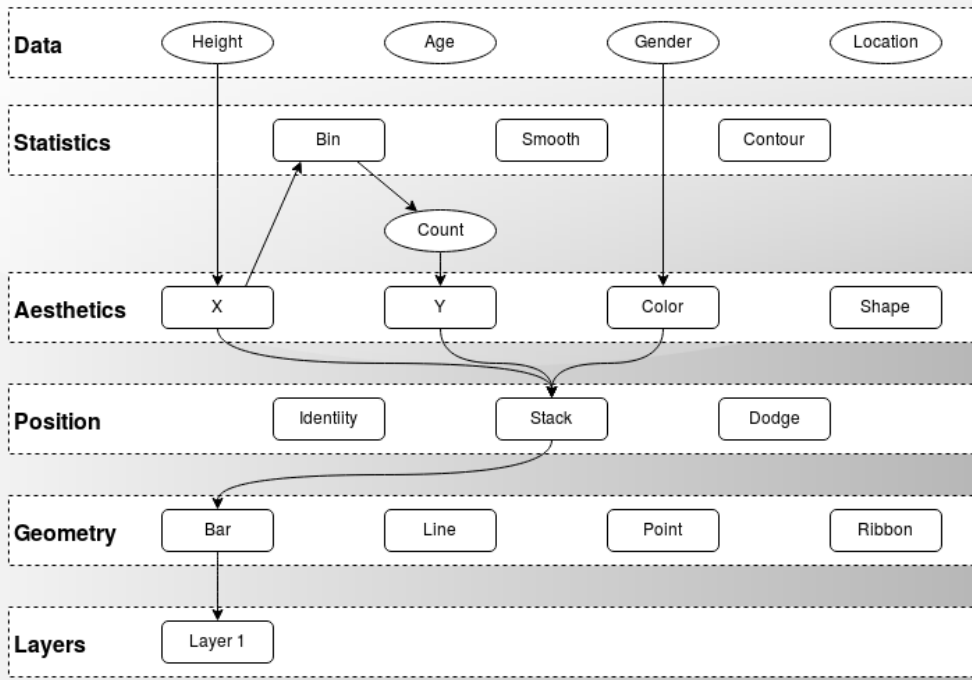
Not required, sensible defaults supplied



# Przykład – bar graph

```
ggplot(data=profile, aes(x=height, fill=gender)) +  
  geom_bar(aes(y=..count..), position="stack", stat="bin")
```

Grammar of Graphics: Bar Graph



# Wykresy: typy wizualizacji danych (ggplot)

## One Variable

### Continuous

`c <- ggplot(mpg, aes(hwy)); c2 <- ggplot(mpg)`



**c + geom\_area(stat = "bin")**  
x, y, alpha, color, fill, linetype, size



**c + geom\_density(kernel = "gaussian")**  
x, y, alpha, color, fill, group, linetype, size, weight



**c + geom\_dotplot()**  
x, y, alpha, color, fill



**c + geom\_freqpoly()**  
x, y, alpha, color, group, linetype, size



**c + geom\_histogram(binwidth = 5)**  
x, y, alpha, color, fill, linetype, size, weight



**c2 + geom\_qq(aes(sample = hwy))**  
x, y, alpha, color, fill, linetype, size, weight

### Discrete

`d <- ggplot(mpg, aes(fl))`



**d + geom\_bar()**  
x, alpha, color, fill, linetype, size, weight

## Two Variables

### Continuous X, Continuous Y

`e <- ggplot(mpg, aes(cty, hwy))`



**e + geom\_label(aes(label = cty), nudge\_x = 1, nudge\_y = 1, check\_overlap = TRUE)**  
x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust



**e + geom\_jitter(height = 2, width = 2)**  
x, y, alpha, color, fill, shape, size



**e + geom\_point()**  
x, y, alpha, color, fill, shape, size, stroke



**e + geom\_quantile()**  
x, y, alpha, color, group, linetype, size, weight



**e + geom\_rug(sides = "bl")**  
x, y, alpha, color, linetype, size



**e + geom\_smooth(method = lm)**  
x, y, alpha, color, fill, group, linetype, size, weight



**e + geom\_text(aes(label = cty), nudge\_x = 1, nudge\_y = 1, check\_overlap = TRUE)**  
x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust

### Continuous Bivariate Distribution

`h <- ggplot(diamonds, aes(carat, price))`



**h + geom\_bin2d(binwidth = c(0.25, 500))**  
x, y, alpha, color, fill, linetype, size, weight



**h + geom\_density2d()**  
x, y, alpha, colour, group, linetype, size



**h + geom\_hex()**  
x, y, alpha, colour, fill, size

### Continuous Function

`i <- ggplot(economics, aes(date, unemployment))`



**i + geom\_area()**  
x, y, alpha, color, fill, linetype, size



**i + geom\_line()**  
x, y, alpha, color, group, linetype, size



**i + geom\_step(direction = "hv")**  
x, y, alpha, color, group, linetype, size

### Discrete X, Continuous Y

`f <- ggplot(mpg, aes(class, hwy))`



**f + geom\_col()**  
x, y, alpha, color, fill, group, linetype, size



**f + geom\_boxplot()**  
x, y, lower, middle, upper, ymax, ymin, alpha, color, fill, group, linetype, shape, size, weight



**f + geom\_dotplot(binaxis = "y", stackdir = "center")**  
x, y, alpha, color, fill, group



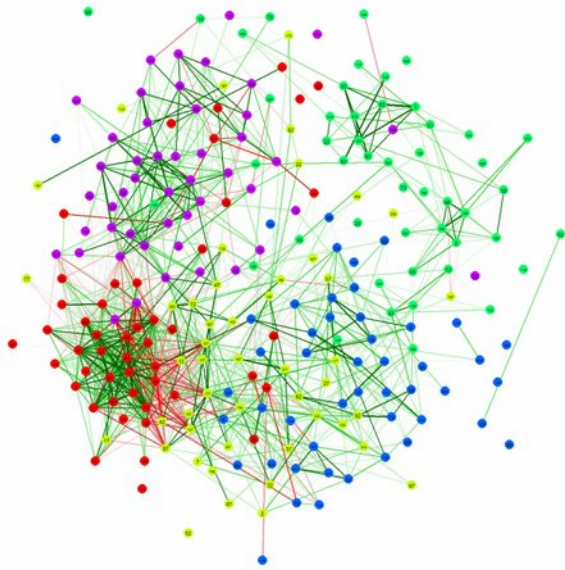
**f + geom\_violin(scale = "area")**  
x, y, alpha, color, fill, group, linetype, size, weight



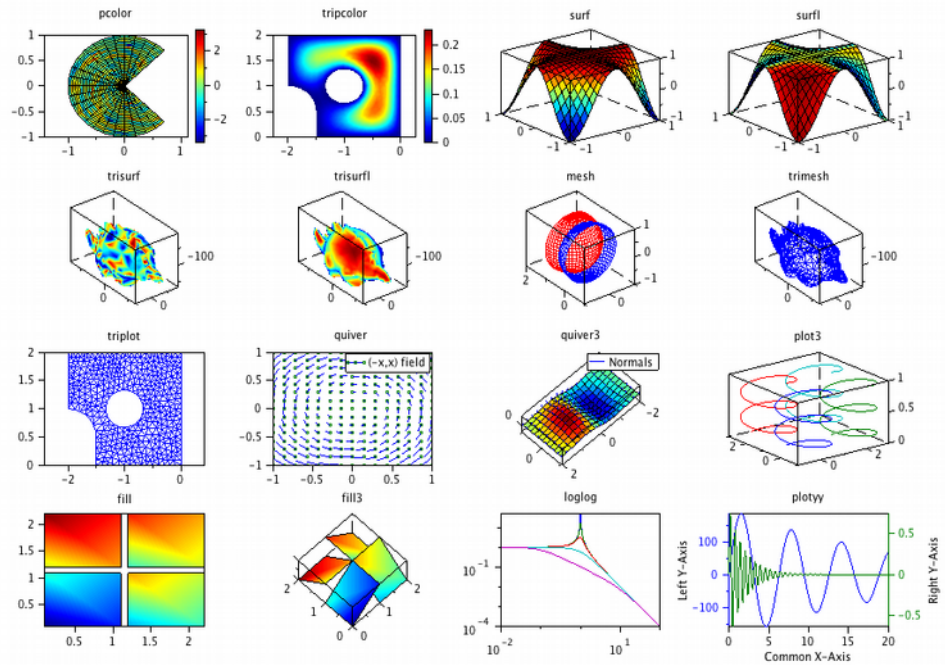


# Zaawansowane wykresy

Big 5 correlations



- Neuroticism
- Extraversion
- Openness
- Agreeableness
- Conscientiousness



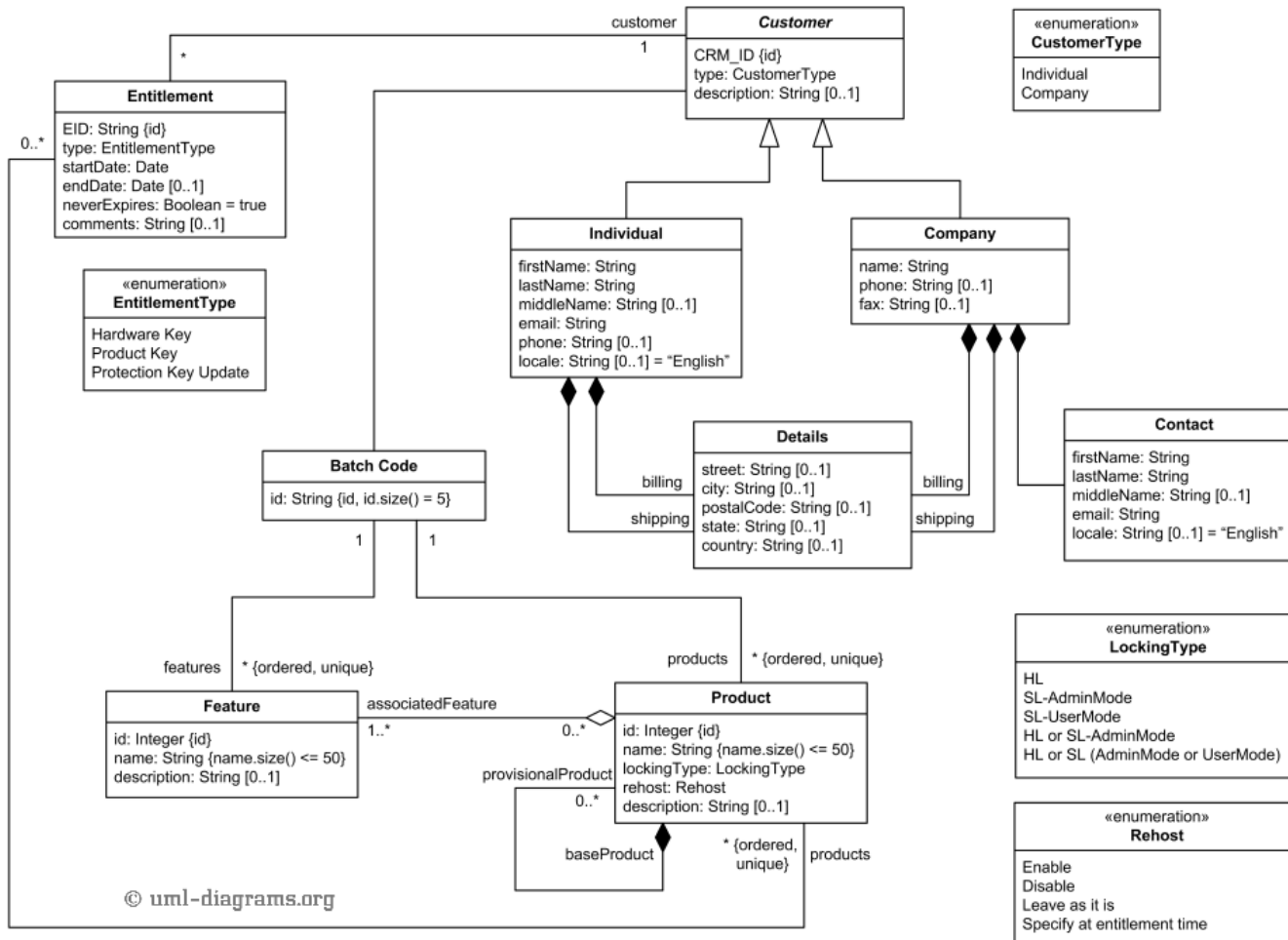
# Diagramy i modele

- Wizualizacja graficzna modeli lub procesów. Są standaryzowane języki wizualne pozwalające na wizualizację złożonych zjawisk w formie prostego diagramu
  - Diagramy przepływu
  - Diagramy organizacyjne
  - Diagramy UML (bazy, programy)
  - Diagramy idei (mind maps)
  - Diagramy sieci

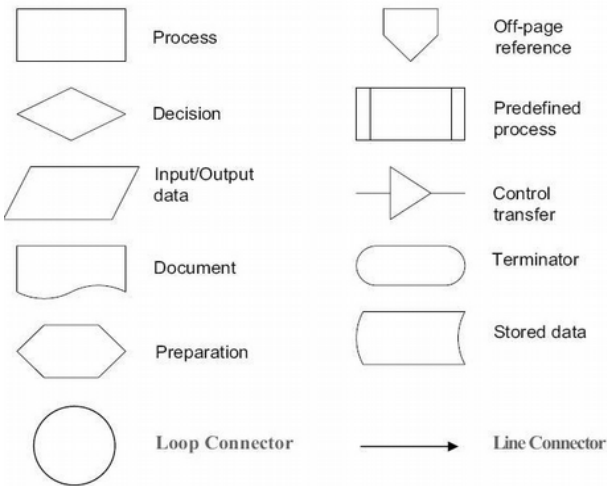
# UML - zunifikowany język modelowania

- Język pół-formalny, graficzny system wizualnego modelowania. Specyfikowania oraz dokumentowania poprzez opisywanie fragmentów rzeczywistości. Obejmuje diagramy struktur (klas, obiektów, komponentów) oraz zachowań (czynności, użycia, stanów interakcji)
- Wykorzystywany do modelowania złożonych struktur (przedsiębiorstwa, grupy itp)
- W informatyce używany do projektowania baz danych i programów komputerowych

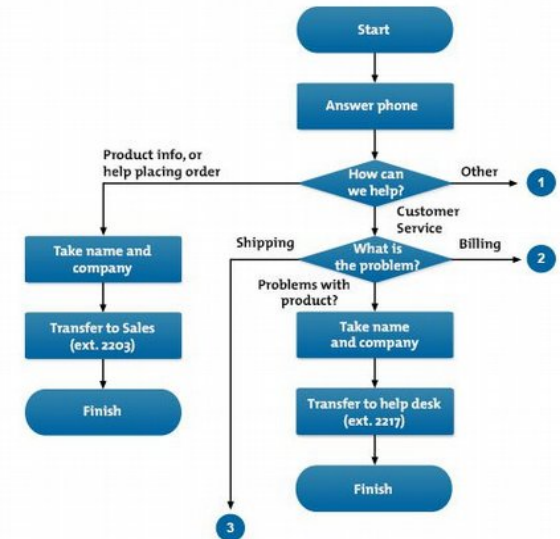
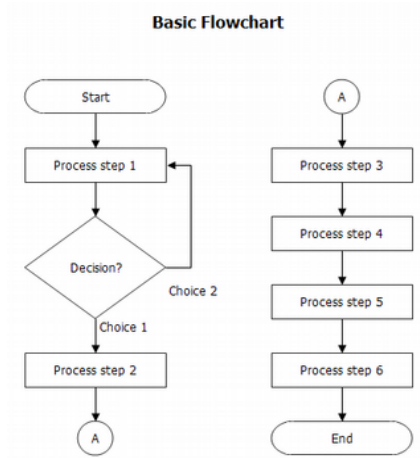
# Diagramy UML



# Diagramy przepływu (algorytmy)

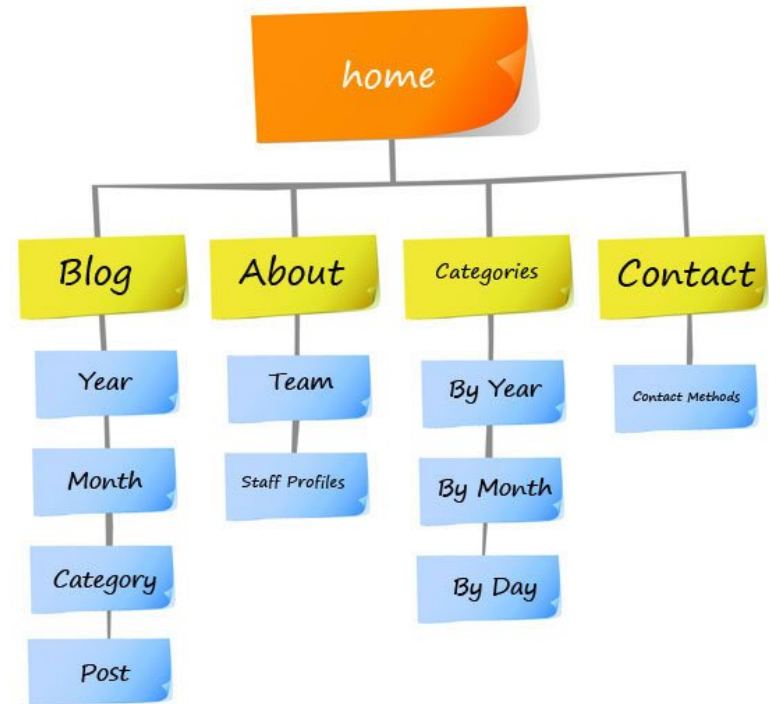
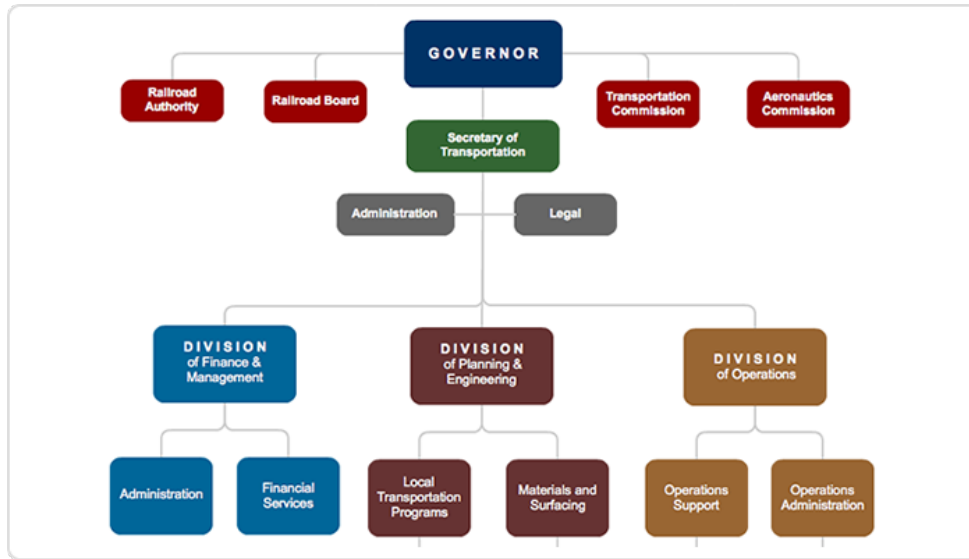


Elementy języka



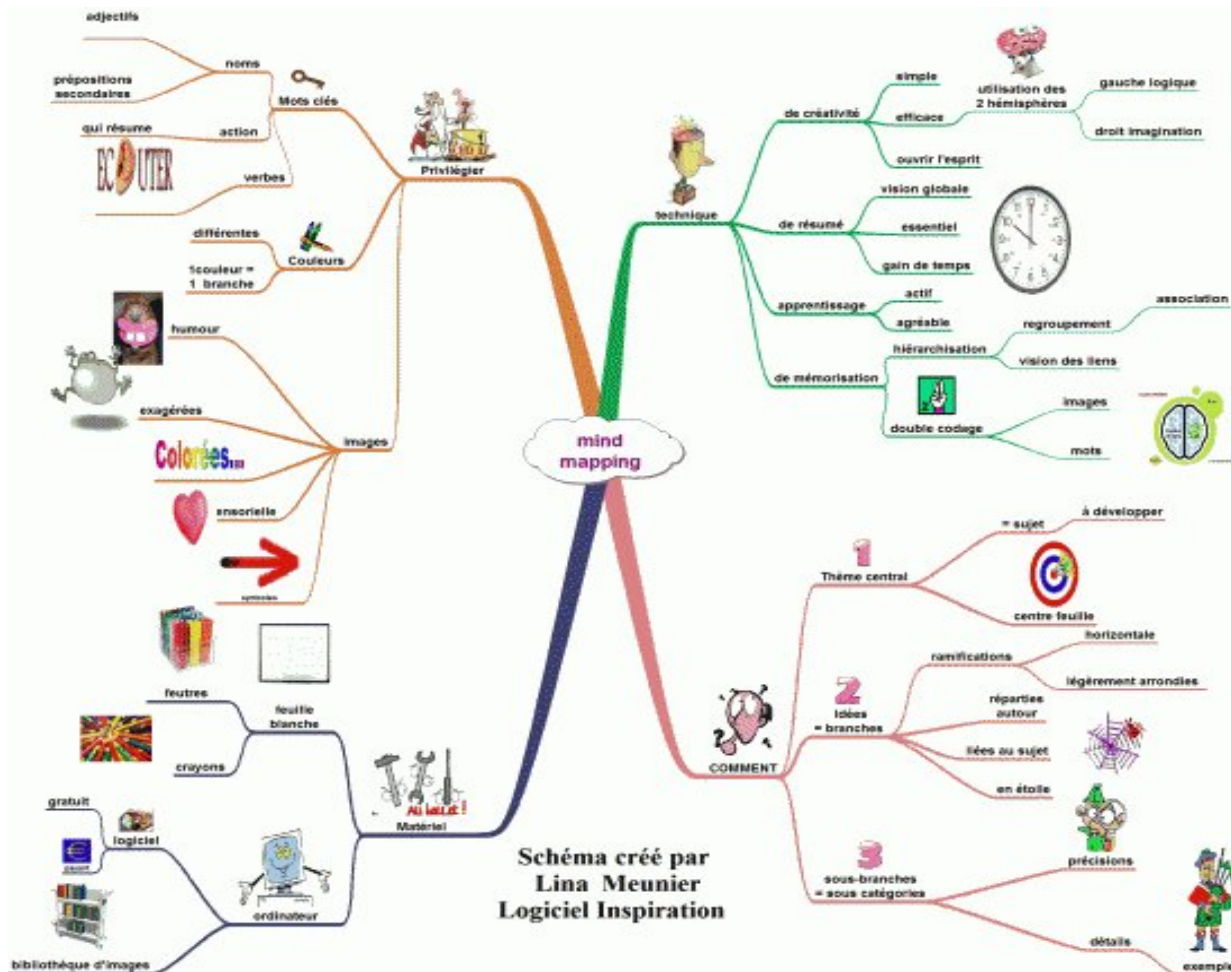
Przykład użycia

# Prezentacja struktur hierarchicznych: diagramy organizacyjne i site maps



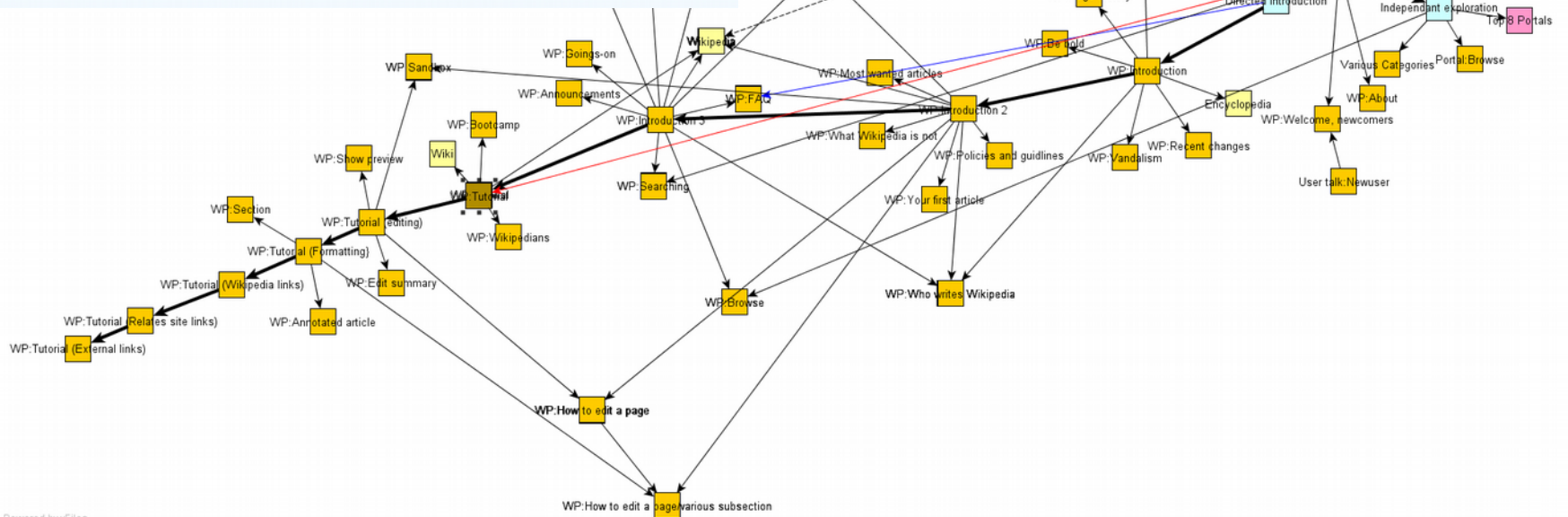
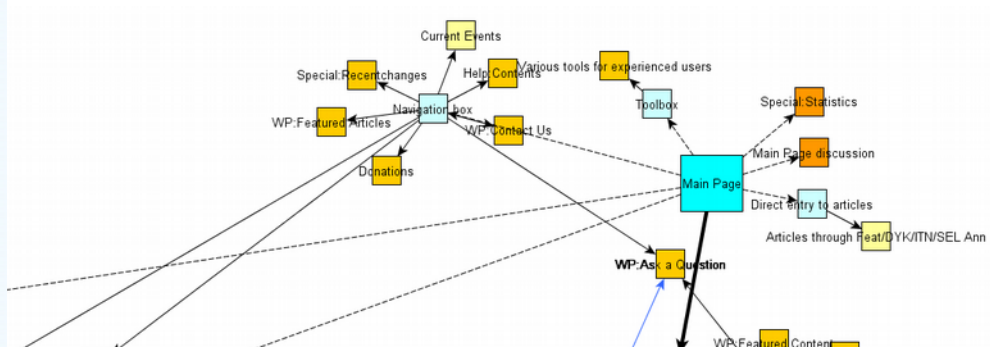
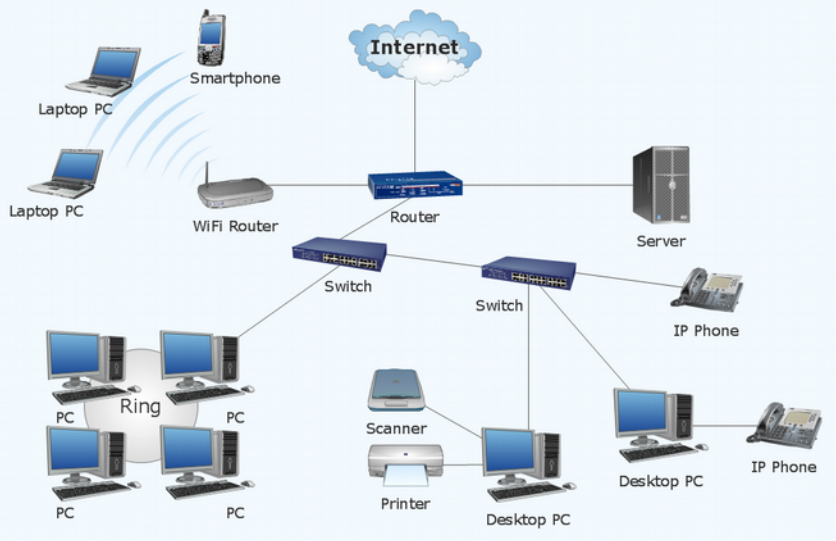
# Wizualizacja idei

## Diagramy map umysłu (mind maps)





# Diagramy sieci i grafów



# Wizualizacja kartograficzna

- Prezentacja danych geo-przestrzennych w formie mapy – w praktyce bardzo rozbudowane systemy wykresów, które łączą techniki systemów informacji geograficznej (źródło danych) i grafiki prezentacyjnej (wizualizacja w przestrzeni geograficznej)
  - Kartogramy
  - Kartodiagramy
  - Mapy (schematy) topologiczne
  - Prezentacje złożone

# Języki grafiki prezentacyjnej

- R – biblioteki graphics, lattice, ggplot2
- Matlab
- Mathematica
- Octave (open source matlab)
- Scilab
- Przykładowe biblioteki języków ogólnego zastosowania
  - Python – biblioteka matplotlib
  - C/C++ – biblioteka MathGL
  - JavaFX charts



# Przykłady: R + ggplot + RStudio IDE

~/Blogs/Introduction to R - RStudio

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function

Introduction to R

ggplot.R \* query\_gamestart.txt \* data \*

Source on Save Run Source

```
89 data %>%
90   filter(gender != 'UNKNOWN') %>%
91   ggplot() +
92     aes(x = days_seen_group, fill = gender) +
93     geom_bar(position = 'dodge')
94
95 # What about age groups
96 data %>%
97   filter(age_group != 'UNKNOWN') %>%
98   ggplot() +
99     aes(x = days_seen_group, fill = age_group) +
100     geom_bar()
101
102 # Can look at proportions
103 data %>%
104   filter(age_group != 'UNKNOWN') %>%
105   ggplot() +
106     aes(x = days_seen_group, fill = age_group) +
107     geom_bar(position = 'fill')
108
109 #####
110 # Heatplot of what level people get to #
111 #####
112
113 # Prepare the data for plotting
114 plot_data <-
115 <
116
```

103:1 (Untitled) R Script

History

Console ~/Blogs/Introduction to R/

```
> data %>%
+   filter(age_group != 'UNKNOWN') %>%
+   ggplot() +
+     aes(x = days_seen_group, fill = age_group) +
+     geom_bar(position = 'fill')
> |
```

Environment Files Plots Packages Help Viewer

Zoom Export

days_seen_group	13-17	18-24	25-29	30-34	35-39	40-49	50-64
(-1,0]	0.10	0.15	0.15	0.15	0.15	0.15	0.10
(0,5]	0.10	0.15	0.15	0.15	0.15	0.15	0.10
(5,10]	0.10	0.15	0.15	0.15	0.15	0.15	0.10
(10,100]	0.10	0.15	0.15	0.15	0.15	0.15	0.10
(100,500]	0.10	0.15	0.15	0.15	0.15	0.15	0.10

# Przykłady: Python + matplotlib + Spyder IDE

The image shows the Spyder Python IDE interface. The main editor window displays a Python script titled `plot_bayesian_ridge.py`. The script includes a docstring, imports for `numpy`, `matplotlib.pyplot`, `scipy`, and `sklearn.linear_model`. It generates simulated data with Gaussian weights and compares the Bayesian Ridge estimate, the ground truth, and the OLS estimate. The plot shows the values of the weights for 100 features, with the Bayesian Ridge estimate (blue solid line) being significantly smoother than the OLS estimate (red dashed line).

The Variable explorer window shows the following variables:

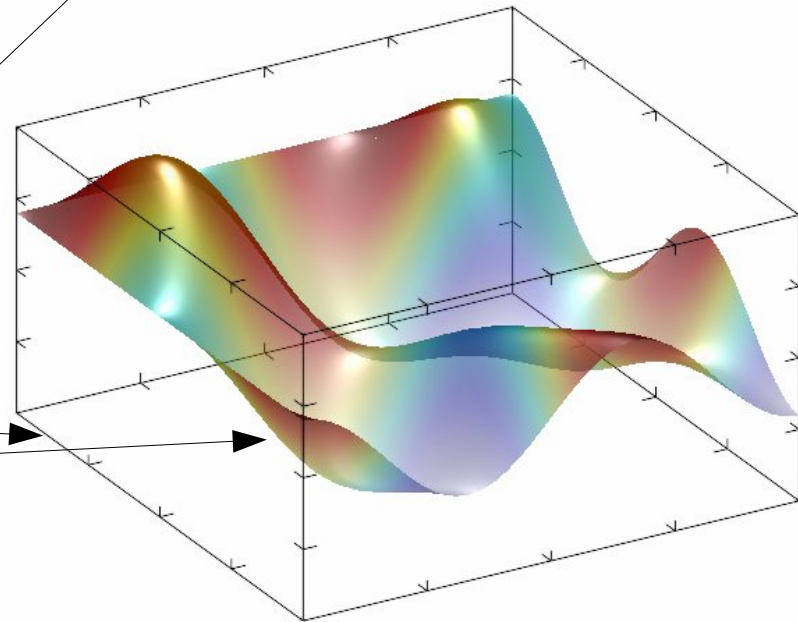
Name	Type	Size	Value
X	float64	(100, 100)	array([[ 1.7640... 0.1269...]])
alpha_	float	1	50.0
digits	datasets.base.Bunch	5	{'target': ar... 1., 0.]}}
i	int32	1	42
iris	datasets.base.Bunch	5	{'target': ar... 5.1, 1.8]}}

The IPython console window displays the plot titled "Weights of the model". The plot shows the values of the weights for 100 features. The y-axis is labeled "Values of the weights" and ranges from -0.8 to 0.8. The x-axis ranges from 0 to 100. The plot compares three estimates: Bayesian Ridge estimate (blue solid line), Ground truth (green solid line), and OLS estimate (red dashed line). The Bayesian Ridge estimate is significantly smoother than the OLS estimate, which exhibits high variance and oscillations.

# Przykłady mathGL

## SurfCA plot

```
int sample(mglGraph *gr)
{
    mglData a,b;
    mglS_prepare2d(&a,&b);
    gr->Title("SurfCA plot");
    gr->Rotate(50,60);
    gr->Alpha(true);
    gr->Light(true);
    gr->Box();
    gr->SurfCA(a,b,a);
    return 0;
}
```

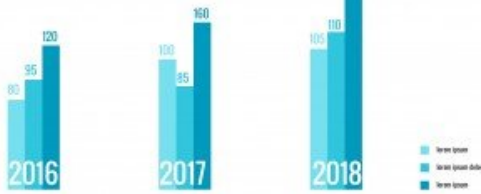






# Infografika

## INFOGRAPHIC ELEMENTS



**LOREM IPSUM DOLOR**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut magna nibh. Praesent sit amet nulla in ipsum tincidunt pulviner. Cras quisque dapibus mollis, ultricies semper interdum. Nullam euismod dolor. Duis mollis, est non commodo luctus, nisi erat porttitor ligula. Integer tincidunt odio in lectus. Sed ac ultricies diam.



28%



18%

25%

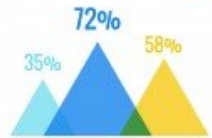


25%



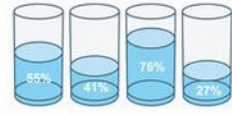
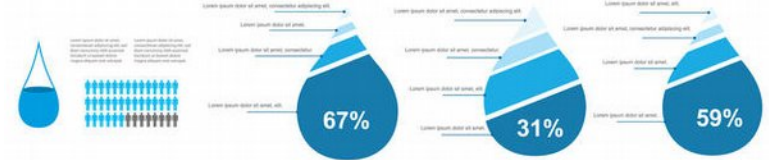
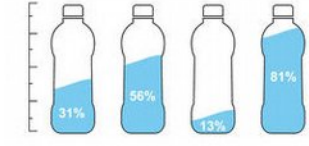
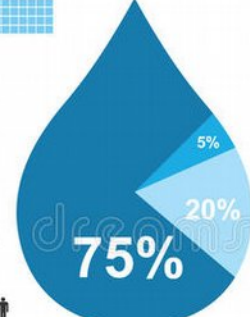
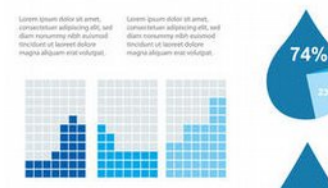
## Lorem Ipsum Dolor

01. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut magna nibh. Praesent sit amet nulla in ipsum tincidunt pulviner. Cras quisque dapibus mollis, ultricies semper interdum. Nullam euismod dolor. Duis mollis, est non commodo luctus, nisi erat porttitor ligula. Integer tincidunt odio in lectus. Sed ac ultricies diam.
02. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut magna nibh. Praesent sit amet nulla in ipsum tincidunt pulviner. Cras quisque dapibus mollis, ultricies semper interdum. Nullam euismod dolor. Duis mollis, est non commodo luctus, nisi erat porttitor ligula. Integer tincidunt odio in lectus. Sed ac ultricies diam.
03. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut magna nibh. Praesent sit amet nulla in ipsum tincidunt pulviner. Cras quisque dapibus mollis, ultricies semper interdum. Nullam euismod dolor. Duis mollis, est non commodo luctus, nisi erat porttitor ligula. Integer tincidunt odio in lectus. Sed ac ultricies diam.



**LOREM IPSUM DOLOR**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut magna nibh. Praesent sit amet nulla in ipsum tincidunt pulviner. Cras quisque dapibus mollis, ultricies semper interdum. Nullam euismod dolor. Duis mollis, est non commodo luctus, nisi erat porttitor ligula. Integer tincidunt odio in lectus. Sed ac ultricies diam.

## WATER INFOGRAPHICS



87%  
 36%  
 24%



74%  
 28%

