

GRAFIKA W PYTHONIE – CZY WĄŻ ROZUMIE WEKTORY?

Agnieszka Borowiecka
Ośrodek Edukacji Informatycznej i Zastosowań Komputerów
w Warszawie, Raszyńska 8/10
agnieszka.borowiecka@oeiizk.edu.pl

Abstract. *In vector graphics, an image is defined using mathematically described objects. Inkscape, together with the Simple Inkscape Scripting extension, allows you to use Python programming skills to creatively generate graphics, for example, for school promotions or in teaching materials.*

1. Wstęp

O znaczeniu grafiki nie trzeba nikogo przekonywać. Właściwie dobrany obraz zostanie lepiej zapamiętany, niż najbardziej poruszające słowa. Przygotowując ilustrację lub infografikę powinniśmy wykazywać się pomysłowością i przenikliwością w wychwyceniu najbardziej istotnych szczegółów. Jednak w panującym dziś zalewie informacji oraz przy milionach kilobajtów przesyłanych i przetwarzanych danych, wielkość miejsca, jaką możemy przeznaczyć na nasze prace graficzne również zaczyna mieć znaczenie. W przypadku grafiki rastrowej (bitmapowej) ilość wykorzystanego miejsca w pamięci komputera zależy od tego, jak duży i szczegółowy obraz chcemy zapamiętać. Tymczasem grafika wektorowa (obiektowa) – tworzy obraz złożony z figur lub brył geometrycznych umiejscowionych w układzie współrzędnych, a poszczególne obiekty są opisywane za pomocą parametrów (np. współrzędna punktu, długość odcinka) oraz atrybutów (grubość linii, kolor). Podstawową zaletą grafiki wektorowej jest jej skalowalność i możliwość zmiany proporcji bez utraty jakości, a także łatwy dostęp do poszczególnych elementów obrazu oraz ich modyfikowania. Mimo coraz większych możliwości sztucznej inteligencji dokonywanie podobnych zmian w fotografii wciąż jeszcze nie jest tak proste.

2. Tworzenie i edycja grafiki wektorowej

Narzędzia do grafiki wektorowej dostępne są w wielu programach, np. edytorze tekstu, programie do tworzenia prezentacji czy arkuszu kalkulacyjnym. Do dokumentu

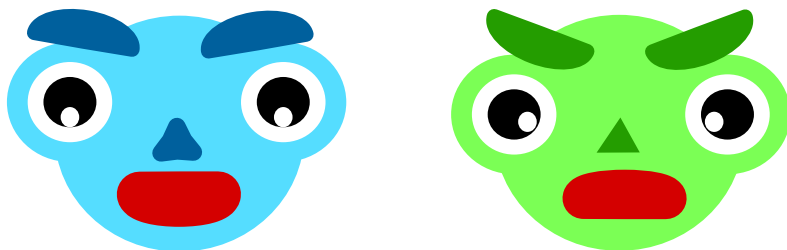
możemy dodawać podstawowe kształty (np. linie, elipsy, prostokąty, strzałki, gwiazdy) oraz ikony. Dodane obiekty można modyfikować, przy czym zakres dostępnych zmian zależy od wybranego programu.



Rysunek 1. Zmienione kolory ikony budzika w programie PowerPoint

Dostępnych jest wielu różnych narzędzi do edycji grafiki wektorowej. Ciekawą pozycją jest Inkscape, bezpłatna wieloplatformowa aplikacja stworzona w ramach projektu GNU. Warto zastanowić się nad wykorzystaniem tego programu podczas pracy w szkole, ze względu na rozbudowaną pomoc do programu, częściowo spolszczone poradniki w menu pomocy oraz szereg filmów instruktażowych dostępnych w serwisie YouTube.

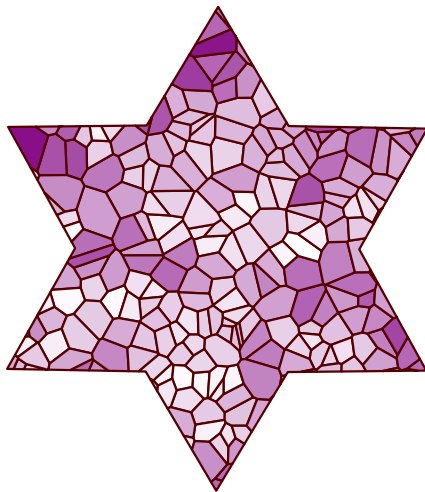
W programie Inkscape domyślny format zapisu plików to SVG, czyli Scalable Vector Graphics, format grafiki wektorowej powszechnie używany w Internecie. Możemy samodzielnie tworzyć rysunki budując je z podstawowych kształtów (prostokąty, elipsy, wielokąty i gwiazdy, linie oraz łuki), a następnie modyfikować za pomocą narzędzi do edycji węzłów czy operacji logicznych na ścieżkach. Można również wczytać gotowe pliki SVG i dostosować je do swoich potrzeb.



Rysunek 2. „Mordki” otrzymane z elips, łuków i trójkątów

Oprócz różnorodnych operacji, w tym bardziej zaawansowanych (np. wyginanie napisów wzdłuż krzywych, tworzenie deseni na ścieżce, interpolacja czy klonowanie),

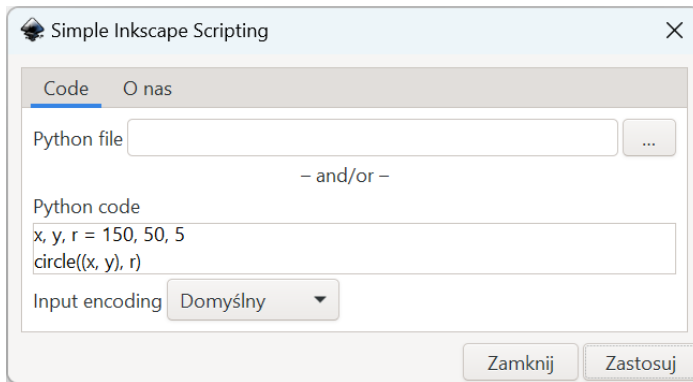
Inkscape oferuje stale powiększaną bibliotekę rozszerzeń. Jedynie od naszej wyobraźni i wytrwałości zależy, jak złożone rysunki przygotujemy.



Rysunek 3. Mozaika w kształcie gwiazdy uzyskana za pomocą rozszerzenia Voronoi

3. Simple Inkscape Scripting

Dla miłośników programowania w języku Python polecamy rozszerzenie Simple Inkscape Scripting (SIS), umożliwiające automatyzację powtarzających się zadań rysunkowych. Po pobraniu rozszerzenia ze strony [5] należy je rozpakować i umieścić w odpowiednim folderze. Aby uruchomić skrypt wybieramy **Efekty | Renderowanie | Simple Inkscape Scripting...** Linijki programu możemy wprowadzić bezpośrednio w polu tekstowym lub wybrać zapisany plik z dysku. Po naciśnięciu przycisku **Zastosuj** skrypt zostanie uruchomiony i w otwartym dokumencie pojawią się nowe kształty. Można je edytować w programie Inkscape tak samo, jak każdy inny obiekt.

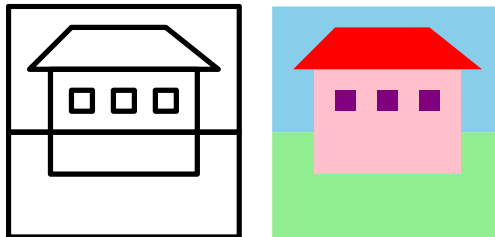


Rysunek 4. Przykładowy skrypt rysujący okrąg o środku w punkcie (150, 50) i promieniu 5

W przypadku dłuższych programów wygodniej jest je pisać korzystając z edytora Pythona (np. IDLE lub Mu), dzięki temu łatwiej uniknąć błędów składniowych.

4. Python pomaga rysować w Inkscape

Rozszerzenie SIS udostępnia bogaty zestaw funkcji Pythona, np. `rect()` do rysowania prostokąta, `line()` – odcinka łączącego dwa punkty, `polygon()` – wielokąta o podanych wierzchołkach. Jeśli nie określimy koloru figur, zostaną przyjęte domyślne ustawienia: brak wypełnienia oraz czarna obwódka o grubości 1 mm. Do każdego rysowanego obiektu możemy dodać atrybuty, np. kolor wypełnienia.



Rysunek 5. Wynik działania programu z ustawieniami domyślnymi i z dodanymi atrybutami określającymi kolory

Pracę możemy rozpocząć od prostego przykładu, który pozwoli lepiej poznać składnię poleceń oraz operowanie współrzędnymi i wymiarowanie kształtów. Możemy narysować na kartce szkic domku z dachem w formie trapezu i trzema oknami, przyjąć jednostki odpowiadające pojedynczej kratce i wyliczyć współrzędne odpowiednich

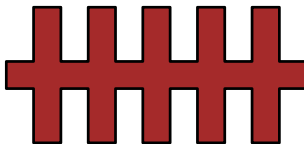
punktów. Następnie wystarczy napisać fragment kodu, wczytać go w oknie Simple Inkscape Scripting i zlecić wykonanie za pomocą przycisku **Zastosuj**.

```
style(stroke_width = 0)

rect((0, 0), (44, 24), fill = 'skyblue')
rect((0, 24), (44, 44), fill = 'lightgreen')
rect((8, 12), (36, 32), fill = 'pink')
polygon([(4, 12), (40, 12), (30, 4), (12, 4)], fill = '#ff0000')
for i in range(0, 5, 2):
    rect((12 + 4 * i, 16), (12 + 4 * (i + 1), 20), fill = 'purple')
```

Rysunek 6. Program rysujący domek z czerwonym dachem i trzema oknami na niebiesko-zielonym tle

Rozszerzenie SIS udostępnia m.in. operacje grupowania, duplikowania i klonowania obiektów, a także narzędzia do transformacji – przesuwania, obracania, skalowania i pochylania. Korzystając ze zmiennych, instrukcji warunkowych, pętli oraz funkcji, możemy łatwo tworzyć bardziej złożone rysunki. Dzięki operacjom na ścieżkach możliwe jest także tworzenie nietypowych kształtów, na przykład krzywej wyglądającej jak płotek złożony z pięciu sztachet i łączącej je belki.



Rysunek 7 Krzywa w kształcie płotka

```
deski = []

def plot(x, y, sz, wys, ile):
    style(fill = 'brown', stroke_width = 1, stroke = '#000000')
    for i in range(ile):
        p = rect((x + sz * (2 * i + 1), y), (x + sz * (2 * i + 2), y + wys))
        deski.append(p)
    p = rect((0, (wys - sz) / 2), ((2 * ile + 1) * sz, (wys + sz) / 2))
    deski.append(p)
    plotek = apply_path_operation('union', [d.to_path() for d in deski])

plot(0, 0, 10, 50, 5)
```

Rysunek 8. Program rysujący płot złożony z 5 sztachet o podanej wielkości

Stopniowo możemy przejść do tworzenia coraz bardziej skomplikowanych wzorów, korzystając z elementów losowości lub rekurencji. Przy wyliczaniu współrzędnych musimy wiedzieć, w jaki sposób tworzone są kolejne etapy rysunku, np. przy rysowaniu

trójkąta Sierpińskiego postępujemy w następujący sposób: rysujemy trójkąt, dzielimy go na 4 mniejsze trójkąty i usuwamy środkowy z nich.

```
from math import sqrt

def rysuj_dywan(n, x, y, a):
    if n == 0:
        polygon([(x, y), (x + a, y), (x + a / 2, y - a * sqrt(3) / 2)])
    else:
        rysuj_dywan(n - 1, x, y, a / 2)
        rysuj_dywan(n - 1, x + a / 2, y, a / 2)
        rysuj_dywan(n - 1, x + a / 4, y - a * sqrt(3) / 4, a / 2)

for obj in all_shapes():
    obj.remove()

style(stroke = 'blue', stroke_width = 0.2)
a = 100
n = 3
rysuj_dywan(n, 0, 0, a)

canvas.resize_to_content()
```

Rysunek 9. Program rysujący trójkątny dywan Sierpińskiego stopnia 3

5. Podsumowanie

Grafika wektorowa może być wykorzystywana m.in. na stronach internetowych, do promocji szkoły czy w materiałach dydaktycznych. Warto z uczniami przyrzeć się możliwościom programu Inkscape. Dodanie rozszerzenia SIS pozwoli wpleść naukę programowania oraz myślenie matematyczne w nasze zajęcia. W wielu sytuacjach umiejętność programowania w języku Python może przyspieszyć lub ułatwić tworzenie rysunków, szczególnie wtedy, gdy składają się one z dużej liczby precyzyjnie rozmieszczonych, powtarzających się albo podobnych kształtów. Należy jednak pamiętać, że nie wszystkie możliwości programu Inkscape zostały uwzględnione w rozszerzeniu Simple Inkscape Scripting, a wiele czynności może się okazać łatwiejsze do wykonania przy użyciu myszki, niż za pomocą złożonych obliczeń matematycznych.

Literatura

1. Borowiecka A., *Python pomaga rysować w Inkscape*, kwartalnik W cyfrowej szkole 1/2025, OEIiZK, Warszawa 2025.
2. Borowiecka A., *Fraktale w Inkscape*, kwartalnik W cyfrowej szkole 3/2025, OEIiZK, Warszawa 2025, w przygotowaniu.
3. Dokumentacja Simple Inkscape Scripting, <https://tiny.pl/gf90232n>, ostatni dostęp 8.06.2025 roku.

4. *How to Create a Mosaic Using Voronoi Pattern in Inkscape*, Design Bundles.NET, <https://tiny.pl/kyydz7yn>, ostatni dostęp 8.06.2025 roku.
5. Rozszerzenie Simple Inkscape Scripting, <https://tiny.pl/r282ckz8>, ostatni dostęp 8.06.2025 roku.
6. Strona Inkscape, <https://inkscape.org>, ostatni dostęp 8.06.2025 roku.