

# INFORMATYKA

## – MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA

PROGRAM NAUCZANIA INFORMATYKI Z ELEMENTAMI  
PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH

Informatyka – poziom podstawowy

Środowisko programowania

Witold Kranas

$$\sum_{i=1}^n$$

Człowiek - najlepsza inwestycja



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI



WARSZAWSKA  
WYŻSZA SZKOŁA  
INFORMATYKI

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Tytuł: **Środowisko programowania**

Autor: **Witold Kranas**

Redaktor merytoryczny: **prof. dr hab. Maciej M. Sysło**

Materiał dydaktyczny opracowany w ramach projektu edukacyjnego  
**Informatyka – mój sposób na poznanie i opisanie świata.**  
**Program nauczania informatyki z elementami przedmiotów**  
**matematyczno-przyrodniczych**

[www.info-plus.wysi.edu.pl](http://www.info-plus.wysi.edu.pl)

[infoplus@wysi.edu.pl](mailto:infoplus@wysi.edu.pl)

Wydawca: Warszawska Wyższa Szkoła Informatyki  
ul. Lewartowskiego 17, 00-169 Warszawa  
[www.wysi.edu.pl](http://www.wysi.edu.pl)  
[rektorat@wysi.edu.pl](mailto:rektorat@wysi.edu.pl)

Projekt graficzny: *Marzena Kamasa*

Warszawa 2013

Copyright © Warszawska Wyższa Szkoła Informatyki 2013  
Publikacja nie jest przeznaczona do sprzedaży

*Człowiek - najlepsza inwestycja*



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI



WARSZAWSKA  
WYŻSZA SZKOŁA  
INFORMATYKI

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY





# SCENARIUSZ TEMATYCZNY

## ŚRODOWISKO PROGRAMOWANIA

→ INFORMATYKA – POZIOM PODSTAWOWY

**OPRACOWANY W RAMACH PROJEKTU:**  
**INFORMATYKA – MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA.**  
**PROGRAM NAUCZANIA INFORMATYKI**  
**Z ELEMENTAMI PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH**

### **Streszczenie**

Poznanie środowiska programowania Akademii Khana (Khan Academy Computer Science) na zasadach odwróconej lekcji. Korzystanie z tego środowiska wymaga znajomości języka angielskiego, zwłaszcza jeśli wykorzystywane będzie do nauki podstaw programowania.

Nauczyciel wprowadza uczniów do JS Processing – środowiska programistycznego Akademii Khana. Środowisko to jest wyposażone w przewodniki (tutoriale) oraz ćwiczenia przygotowane tak, aby uzasadnić stwierdzenie – programowanie jest dla każdego. Środowisko programowania uczniowie poznają w ramach pracy pozalekcyjnej. Nauczyciel proponuje kilka zadań algorytmicznych: rysowanie symetryczne, ilustrację spadku swobodnego, rysowanie fraktali: drzewa binarnego, trójkąta Sierpińskiego, płątka Kocha. Te zadania uczniowie realizują z pomocą nauczyciela na lekcjach.

### **Czas realizacji**

4 x 45 minut

### **Tematy lekcji**

- 1 Uczymy się programowania w Akademii Khana (1 x 45 minut)
- 2 Dwa zadania algorytmiczne (1 x 45 minut)
- 3 Kodujemy fraktale (2 x 45 minut)



## Podstawa programowa

Poznawanie środowiska programowania – umiejętność kodowania jest niezbędna do realizacji piątego punktu podstawy programowej: *Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego*. Tutaj uczniowie poznają kodowanie na zasadzie nauczania odwróconego – z pomocą Akademii Khana przerabiają kolejne tematy, a zasadnicze cele kształcenia realizują w trakcie lekcji.

### Etap edukacyjny IV, przedmiot informatyka (poziom podstawowy)

#### **Cele kształcenia – wymagania ogólne**

III. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, z zastosowaniem podejścia algorytmicznego.

#### **Treści nauczania – wymagania szczegółowe**

5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego.

Uczeń:

- 1) prowadzi dyskusje nad sytuacjami problemowymi;
- 2) formułuje specyfikacje dla wybranych sytuacji problemowych;
- 3) projektuje rozwiązanie: wybiera metodę rozwiązania, odpowiednio dobiera narzędzia komputerowe, tworzy projekt rozwiązania;
- 4) realizuje rozwiązanie na komputerze za pomocą oprogramowania aplikacyjnego lub języka programowania;
- 6) przeprowadza prezentację i omawia zastosowania rozwiązania.

W nawiązaniu do programu nauczania, istotne wiadomości i umiejętności to:

- Poznawanie oprogramowania przez korzystanie z systemu pomocy.
- Poznanie podstawowych możliwości programowania.
- Poznanie typowych konstrukcji algorytmicznych i programistycznych.
- Umiejętność tworzenia planu, algorytmu rozwiązania.
- Umiejętność zamiany algorytmu na program.
- Umiejętność uruchamiania i testowania programu.
- Ocenianie rozwiązania.
- Prezentowanie algorytmu i programu.
- Dyskutowanie wyników programu.



## LEKCJA NR 1

### TEMAT: Uczymy się programowania w Akademii Khana

#### Streszczenie

Wprowadzenie do nauki programowania w środowisku Akademii Khana.

Uczniowie poznali tę możliwość w trakcie lekcji *Korzystanie z Akademii Khana, filmy, ćwiczenia i programowanie*. Posiadają już konta na portalu. Pierwsza lekcja jest przygotowaniem do samodzielnej nauki programowania. Uczniowie zapoznają się też z problemami algorytmicznymi, które będą rozwiązywać na kolejnych lekcjach.

#### Cel

Poznanie środowiska do nauki programowania.

Zaplanowanie samodzielnej nauki.

Wprowadzenie problemów algorytmicznych do rozwiązania.

#### Słowa kluczowe

Khan Academy Computer Science, JavaScript

#### Co przygotować

- Komputer z dostępem do Internetu
- Słuchawki

#### Materiał, z którym nauczyciel powinien zapoznać się przed lekcją:

- film prezentujący możliwości środowiska „Sneak Peek: Making An Awesome Program” (Zapowiedź: Tworzenie niezwykłego programu, 7,5 min)  
[www.khanacademy.org/cs/programming/intro-to-programming/p/sneak-peek-making-an-awesome-program](http://www.khanacademy.org/cs/programming/intro-to-programming/p/sneak-peek-making-an-awesome-program)
- film „A Tour of Programming on Khan Academy” (Dookoła programowania w Akademii Khana, 6 min)  
[www.khanacademy.org/cs/programming/intro-to-programming/v/programming-tour](http://www.khanacademy.org/cs/programming/intro-to-programming/v/programming-tour)
- dział Drawing basics (Podstawy rysowania)  
[www.khanacademy.org/cs/programming/drawing-basics/p/intro-to-drawing](http://www.khanacademy.org/cs/programming/drawing-basics/p/intro-to-drawing)

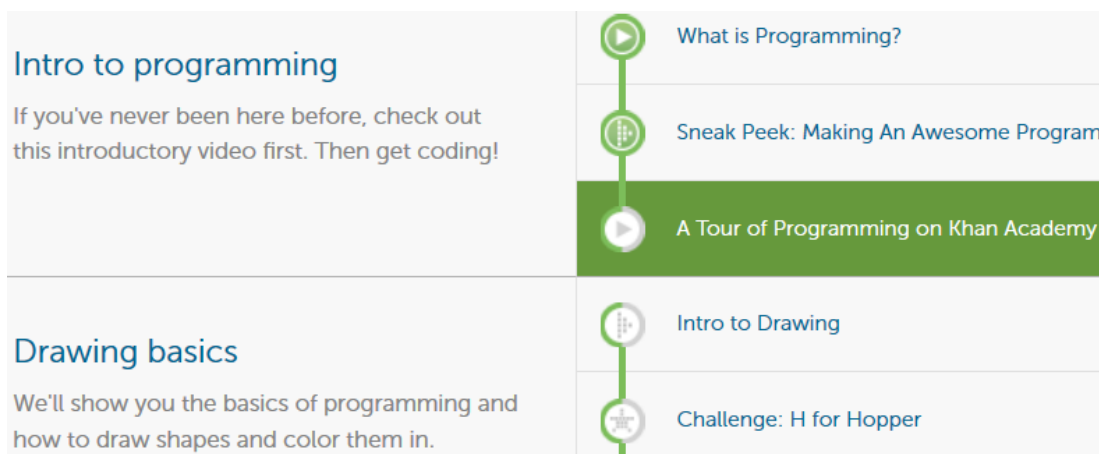
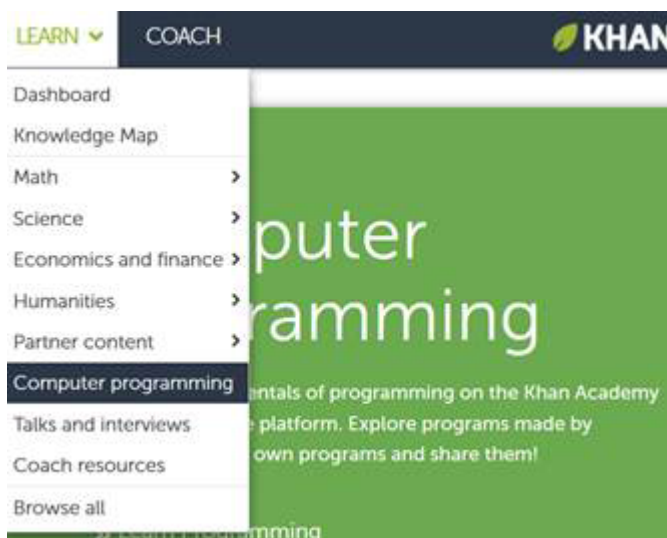


## Przebieg zajęć

### Wprowadzenie (15 minut)

Nauczyciel przypomina uczniom lekcję *Korzystanie z Akademii Khana, filmy, ćwiczenia i programowanie*.

Uczniowie logują się na portalu Akademii Khana i wybierają LEARN/Computer programming, a następnie Learn programming. Nauczyciel przedstawia sposób pracy w tym dziale Akademii Khana i możliwości środowiska. Może posłużyć się filmem „A Tour of Programming on Khan Academy” (Dookoła programowania w Akademii Khana, 6 min, polskie napisy).



Nauczyciel przypomina uczniom, że oglądali film o tworzeniu programu do malowania – „Sneak Peek: Making An Awesome Program” (Zapowiedź: Tworzenie niezwykłego programu). Pyta, co zapamiętali. Wybrani uczniowie odpowiadają. Nauczyciel wyjaśnia, że program jest pisany w języku JavaScript, a do grafiki wykorzystywane jest środowisko ProcessingJS. JavaScript jest popularnym językiem wykorzystywanym przy tworzeniu stron WWW.

### Problemy algorytmiczne (15 minut)

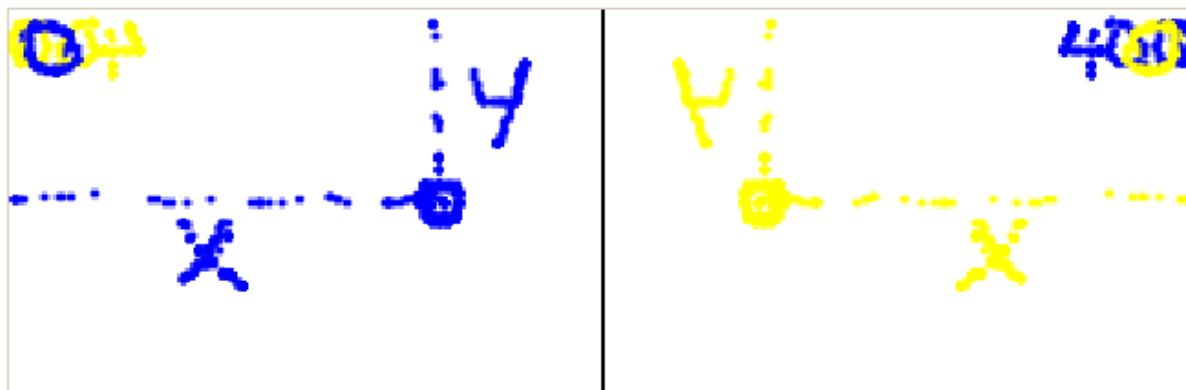
Nauczyciel wyjaśnia, że na kolejnej lekcji uczniowie będą rozwiązywać jedno z zadań. Do wyboru będą dwa problemy:

- Malowanie symetryczne,
- Symulacja spadku swobodnego.

Krótko przedstawia specyfikację obu problemów. Może posługiwać się przygotowanymi programami.

### Malowanie symetryczne

W filmie „Sneak Peek: Making An Awesome” uczniowie obserwowali, w jaki sposób utworzyć kolorowe kółko, które zostawiając ślad, podąża za ruchem kursora myszy – rodzaj pędzla do malowania. Jest to zaczątek edytora graficznego. Zadaniem uczniów będzie utworzenie takiego edytora do malowania symetrycznego. Przesuwanie kursorem myszy po kanwie rysunku powinno prowadzić do powstawania kolorowego śladu ruchu kursora oraz jego odbicia symetrycznego (lustrzanego) względem osi pionowej przechodzącej przez środek kanwy.

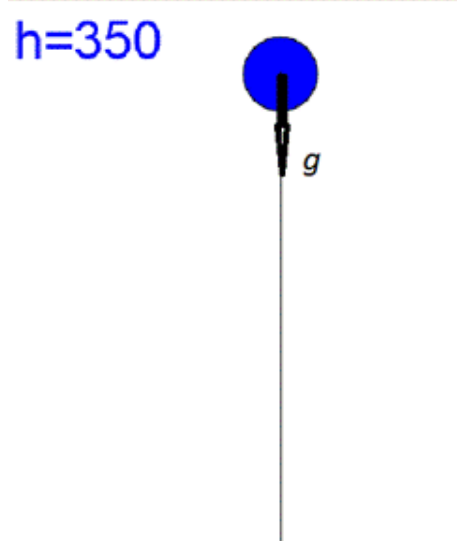


Krótką dyskusja powinna doprowadzić do wyjaśnienia, czy uczniowie rozumieją problem.

### Symulacja spadku swobodnego

Zadaniem uczniów jest utworzenie animacji przedstawiającej spadek swobodny kulki bez uwzględnienia oporu powietrza. Spadek swobodny to ruch jednostajnie przyspieszony przyspieszeniem ziemskim ( $g$ , ok.  $10 \text{ m/s}^2$ ). Drogę w tym ruchu (wzdłuż osi pionowej  $Y$ ) można obliczyć ze wzoru  $s = gt^2/2$ .

Symulacja powinna pokazywać ruch kulki do momentu uderzenia w ziemię, czyli osiągnięcia dolnego brzegu kanwy.



Oba zadania wymagają wykorzystania pętli Draw – specyficznej, dość zaawansowanej konstrukcji środowiska ProcessingJS, umożliwiającej powtarzanie poleceń w kółko, przy czym nie jest konieczne określenie warunku zakończenia pętli.

### Planowanie nauki (15 minut)

Nauczyciel wyjaśnia, czego trzeba się nauczyć, by móc rozwiązać przedstawione problemy.

Uczniowie powinni przerobić cały dział Drawing basics (Podstawy rysowania) i część działu Variables (Zmienne). Materiał składa się z interaktywnych filmów prezentujących zagadnienie (np. Intro to Drawing, Intro to Coloring...) oraz zadań do wykonania (Challenge...).

Nauczyciel powinien wyznaczyć zakres materiału do przerobienia, np. do zadania Challenge: Funky Frog (Zadanie: przestraszona żaba).

Uczniowie zaczynają uczyć się podstaw rysowania. W razie napotkania na problemy, proszą nauczyciela o pomoc. Resztę materiału uczniowie powinni opanować w domu.

<b>Drawing basics</b> We'll show you the basics of programming and how to draw shapes and color them in.	Intro to Drawing
	Challenge: H for Hopper
	More Drawing!
	Challenge: Simple Shapes!
	Challenge: CRAZY Face
	Intro to Coloring
	Challenge: Ice Cream Code
	Challenge: It's a Beautiful Day
	Project: What's for Dinner?
<b>Variables</b> We'll cover how to use variables to hold values, animate your drawings, and more.	Intro to Variables
	Challenge: Bucktooth Bunny
	More on Variables
	Challenge: Funky Frog

### Materiały źródłowe



1. Zasoby e-learningowe Akademii Khana: [www.khanacademy.org/cs/programming](http://www.khanacademy.org/cs/programming)
2. Filmy: „Sneak Peek: Making An Awesome Program” oraz „A Tour of Programming on Khan Academy”
3. Dział Drawing basics portalu Khan Academy





## LEKCJA NR 2

### TEMAT: Dwa zadania algorytmiczne

#### Streszczenie

Rozwiązywanie dwóch problemów programistycznych w środowisku Akademii Khana.

Dyskusja algorytmów rozwiązywania zadań, programowanie, dyskusja wyników i możliwości rozszerzenia programów.

#### Cel

Poznanie podejścia algorytmicznego przy rozwiązywaniu problemów.

Umiejętność przekształcenia algorytmu w program.

Umiejętność analizowania wyników i formułowania kolejnych zadań.

#### Słowa kluczowe

JavaScript, symetria, spadek swobodny

#### Co przygotować

- Komputer z dostępem do Internetu
- Słuchawki
- Film prezentujący możliwości środowiska „Sneak Peek: Making An Awesome Program” (Zapowiedź: Tworzenie niezwykłego programu, 7,5 min)  
[www.khanacademy.org/cs/programming/intro-to-programming/p/sneak-peek-making-an-awesome-program](http://www.khanacademy.org/cs/programming/intro-to-programming/p/sneak-peek-making-an-awesome-program)
- Listingi do lekcji 2 (materiały pomocnicze 1)

#### Materiał dotyczący rozwiązania problemów:

- Malowanie symetryczne:  
[www.khanacademy.org/cs/symetrycznie/2760836313](http://www.khanacademy.org/cs/symetrycznie/2760836313), [www.khanacademy.org/cs/symetrycznie1/2758056396](http://www.khanacademy.org/cs/symetrycznie1/2758056396)
- Spadek swobodny: [www.khanacademy.org/cs/spadanie/2736517639](http://www.khanacademy.org/cs/spadanie/2736517639)
- Rzut ukośny: [www.khanacademy.org/cs/rzutuk/2761938735](http://www.khanacademy.org/cs/rzutuk/2761938735)



#### Przebieg zajęć

##### *Dyskusja algorytmów (10 minut)*

Nauczyciel w trakcie dyskusowania problemów z uczniami wypisuje algorytmy ich rozwiązania.

##### ▀ **Zadanie 1.** Malowanie symetryczne

Utworzenie edytora do malowania symetrycznego. Przesuwanie kursorem myszy po kanwie rysunku powinno prowadzić do powstawania kolorowego śladu ruchu kursora oraz jego odbicia symetrycznego (lustrzanego) względem osi pionowej przechodzącej przez środek kanwy.

Nauczyciel przypomina, że uczniowie poznali elementy rozwiązania problemu oglądając film „Sneak Peek: Making An Awesome Program”. Może przypomnieć fragment kodu z filmu:

```

var draw = function() {
  if (mouseIsPressed) {
    fill(183, 0, 255);
    ellipse(mouseX, mouseY, 20, 20);
  }
};

```

Algorytm rozwiązania:

ustal rysowanie bez obwódki

powtarzaj:

    jeśli wciśnięty przycisk myszy

        ustal kolor wypełnienia;

        narysuj kółko na pozycji kursora myszy;

        ustal nowy kolor wypełnienia;

        narysuj kółko na pozycji symetrycznej do kursora myszy;

Jeśli położenie kursora myszy oznaczymy przez  $x$ ,  $y$ , to pozycja symetryczna względem osi pionowej przechodzącej przez środek kanwy ( $x=200$ ) będzie miała współrzędną poziomą wynoszącą  $400 - x$  (400 to rozmiar kanwy) oraz współrzędną pionową równą  $y$ .

### ▀ **Zadanie 2.** Symulacja spadku swobodnego

Utworzenie animacji przedstawiającej spadanie swobodne kulki bez uwzględnienia oporu powietrza.

Nauczyciel krótko omawia spadek swobodny, przypomina wzór na drogę  $s = gt^2/2$ .

Program powinien powtarzać rysowanie kulki w kolejnych położeniach (co pewien czas), a przed narysowaniem kulki w kolejnym położeniu wymazywać poprzedni rysunek.

Algorytm rozwiązania:

ustal wartości początkowe: (przyspieszenie  $g = 10$ , wysokość początkowa  $y_0$ , czas  $t = 0$ ,  
krok czasowy  $dt$  i promień kulki);

ustal kolor wypełnienia i narysuj kulkę w początkowym położeniu;

powtarzaj:

    zwiększ czas o  $dt$ ;

    oblicz nową wysokość;

    jeśli wysokość osiąga dolny brzeg kanwy (400) to przerwij;

    wymaż poprzedni rysunek;

    narysuj kulkę w nowym położeniu;

Należy zwrócić uwagę na obliczanie nowego  $y$ . Oś pionowa kanwy jest skierowana w dół, w takim układzie współrzędnych  $y = y_0 + g*t*t/2$  (można tu użyć funkcji potęga:  $\text{pow}(t, 2)$  oznacza podnoszenie  $t$  do kwadratu).

### **Programowanie (20 minut)**

Uczniowie przystępują do realizacji wybranego algorytmu w środowisku programowania Akademii Khana. Nauczyciel pomaga indywidualnie w przypadku pytań lub problemów. Uczniom, którzy szybko poradzą sobie z zadaniem proponuje rozszerzenie programu. W przypadku zadania 1 może to być malowanie w kolejnych dwóch punktach symetrycznych do pozycji kursora myszy (4 kolory) lub uzależnienie koloru od pozycji kursora myszy. W przypadku zadania 2 naturalne rozszerzenia to rzut poziomy i rzut ukośny.



### **Prezentacja programów i wnioski (10 minut)**

Nauczyciel wybiera uczniów, którzy napisali najciekawsze programy. Uczniowie ci prezentują swoje programy, objaśniają je odpowiadając na pytania innych.

Jeśli czas pozwala, to nauczyciel przedstawia możliwości rozszerzenia programów wskazane w poprzednim punkcie. Może skorzystać z przykładowych rozwiązań zaprezentowanych (wraz z odsyłaczami do działających programów) w pliku **ListingiDoLekcji2.pdf**.

### **Planowanie nauki (5 minut)**

Tematem kolejnej lekcji będzie kodowanie fraktali. Uczniowie powinni przerobić kolejne tematy w Akademii Khana oraz wyszukać informacje na temat fraktali.

Nauczyciel wypisuje na tablicy nazwy fraktali, którymi uczniowie będą się zajmować:

- drzewo binarne,
- trójkąt Sierpińskiego,
- płatek Kocha.

W Akademii Khana uczniowie powinni zapoznać się z następującymi tematami z działu **Animation Basics**: Incrementing Shortcuts i Using Math Expressions, cały dział **Functions** oraz pierwszy temat działu **Logic and if Statements**: If Statements.

### **Dostępne pliki**

1. Listingi do lekcji 2 (materiały pomocnicze 1)
1. Zadanie 1, 2 i 3



### **Materiały źródłowe:**

1. Film „Sneak Peek: Making An Awesome Program”.
2. Zasoby e-learningowe Akademii Khana:
  - Malowanie symetryczne: [www.khanacademy.org/cs/symetrycznie/2760836313](http://www.khanacademy.org/cs/symetrycznie/2760836313), [www.khanacademy.org/cs/symetrycznie1/2758056396](http://www.khanacademy.org/cs/symetrycznie1/2758056396)
  - Spadek swobodny: [www.khanacademy.org/cs/spadanie/2736517639](http://www.khanacademy.org/cs/spadanie/2736517639)
  - Rzut ukośny: [www.khanacademy.org/cs/rzutuk/2761938735](http://www.khanacademy.org/cs/rzutuk/2761938735)



## LEKCJA NR 3

### TEMAT: Kodujemy fraktale

#### Streszczenie

W trakcie pierwszej godziny lekcyjnej uczniowie poznają algorytmy rysowania fraktali. Na kolejnej lekcji realizują je w postaci programów w środowisku Akademii Khana.

#### Cel

Poznanie algorytmów rysowania fraktali.

Poznanie rekurencji, jako metody organizowania pętli w programie.

Poznanie własności fraktali.

#### Słowa kluczowe

fraktale, rekurencja, drzewo binarne, płatek Kocha, krzywa Kocha, trójkąt Sierpińskiego

#### Co przygotować



- Komputer z dostępem do Internetu
- Słuchawki



#### Obejrzyć gotowe programy:

- Drzewo binarne – [www.khanacademy.org/cs/drzewobin/2719885296](http://www.khanacademy.org/cs/drzewobin/2719885296)  
[www.khanacademy.org/cs/drzewobinlos/1014877709](http://www.khanacademy.org/cs/drzewobinlos/1014877709) (z losowością)
- Trójkąt Sierpińskiego – [www.khanacademy.org/cs/trojksierp/2757566516](http://www.khanacademy.org/cs/trojksierp/2757566516)
- Płatek Kocha – [www.khanacademy.org/cs/platekkocha/2762339062](http://www.khanacademy.org/cs/platekkocha/2762339062)



#### Filmy:

- Rysowanie drzewa binarnego – **drzewo.mp4**
- Rysowanie trójkąta Sierpińskiego – **Sierpinski.mp4**
- Rysowanie krzywych Kocha – **krzywaKocha.mp4**



- Listingi do lekcji 3 (materiały pomocnicze 2)
- **PoleObwPłatka.xls** – plik z obliczeniami obwodu i pola powierzchni Płatka Kocha (materiały pomocnicze 3)
- Test wiadomości: **TestProgr.doc**

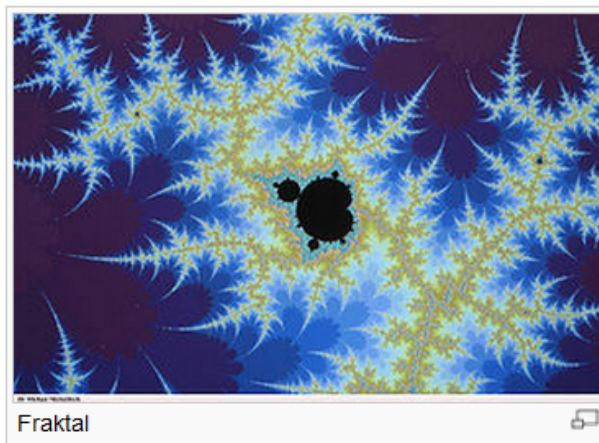
#### Przebieg zajęć

##### Pierwsza godzina lekcyjna

###### *Wprowadzenie – opis fraktali (20 minut)*

Zadaniem uczniów było wyszukanie informacji na temat fraktali. Mogli zajrzeć do Wikipedii, w której znajduje się następująca definicja fraktala:

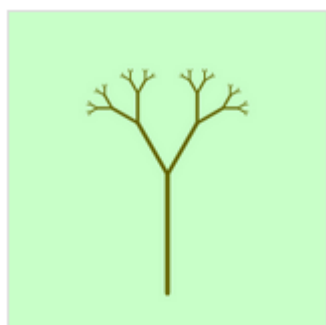
**Fraktal** (łac. *fractus* – złamany, cząstkowy) w znaczeniu potocznym oznacza zwykle obiekt samo-podobny (tzn. taki, którego części są podobne do całości) albo "nieskończenie subtelny" (ukazujący subtelne detale nawet w wielokrotnym powiększeniu). Ze względu na olbrzymią różnorodność przykładów matematycy obecnie unikają podawania ścisłej definicji i proponują określać fraktal jako zbiór, który:



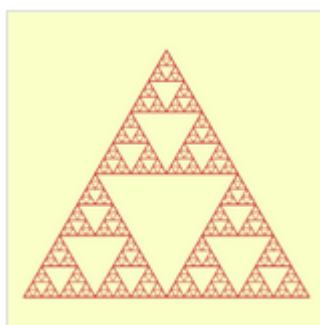
- ma **nietrywialną** strukturę w każdej skali,
- struktura ta nie daje się łatwo opisać w języku tradycyjnej geometrii euklidesowej,
- jest samo-podobny, jeśli nie w sensie dokładnym, to przybliżonym lub **stochastycznym**,
- jego **wymiar Hausdorffa** jest większy niż jego **wymiar topologiczny**,
- ma względnie prostą definicję rekurencyjną,
- ma *naturalny* ("poszarpany", "kłębiasty" itp.) wygląd.

(źródło: [pl.wikipedia.org/wiki/Fraktal](http://pl.wikipedia.org/wiki/Fraktal))

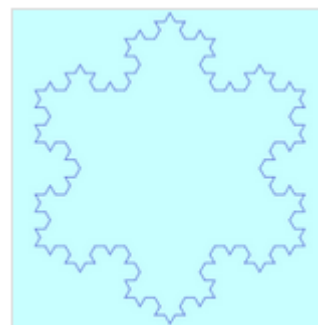
Jest to dobry punkt wyjścia do opisu wybranych fraktali. Są to: drzewo binarne, trójkąt Sierpińskiego i płatek Kocha.



DrzewoBin



trojkSierp



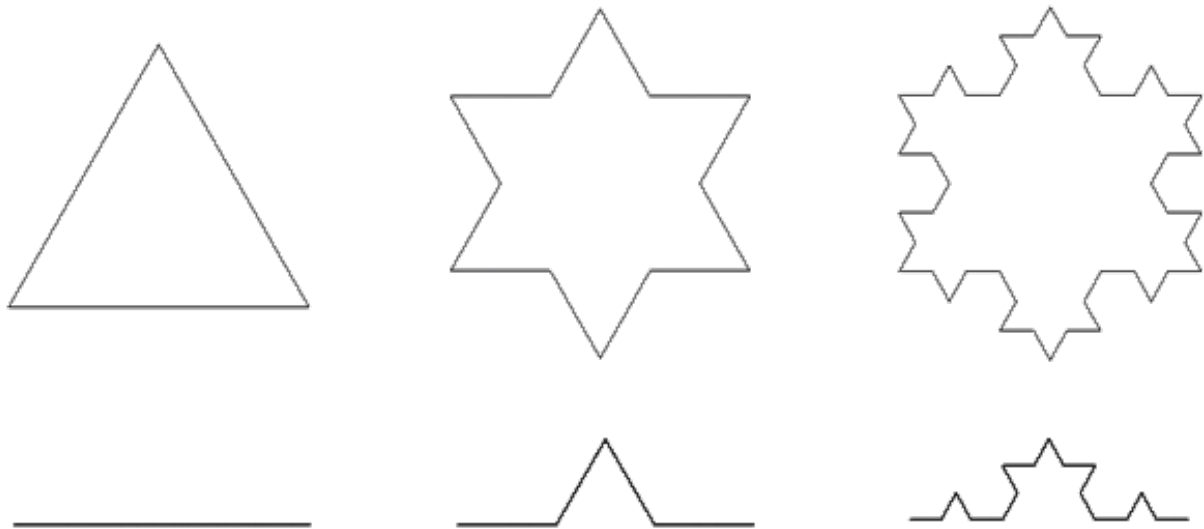
PłatekKocha

Nauczyciel prosi uczniów o wskazanie cechy samopodobieństwa w kolejnych fraktalach.

W trakcie dyskusji uczniowie powinni zauważyć, że:

- drzewo binarne w każdym rozgałęzieniu ma dwa kolejne mniejsze drzewa binarne,
- trójkąt Sierpińskiego składa się z trzech mniejszych trójkątów Sierpińskiego.

Znalezienie samopodobieństwa w przypadku płątka Kocha wymaga dokładniejszego przyjrzenia się krzywej Kocha.



Samopodobieństwo ujawnia się w kolejnych podziałach krzywej Kocha – każdy kolejny jej fragment jest mniejszą krzywą Kocha.

### **Przedstawienie rekurencji na przykładzie drzewa binarnego (30 minut)**

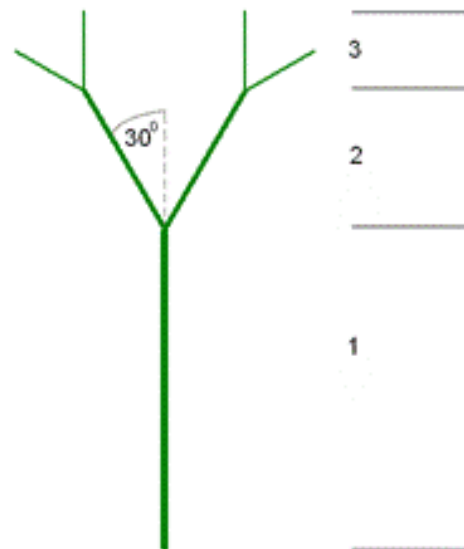
Niektórzy uczniowie spotkali się z rekurencją na lekcjach informatyki w gimnazjum. Jednak jest to metoda trudna i wymaga wprowadzenia. Najłatwiej będzie ją zaprezentować na przykładzie konstrukcji drzewa binarnego. Ten fraktal nauczyciel powinien zbudować razem z uczniami. Przy okazji w środowisku programowania Akademii Khana zostanie zaimplementowana grafika żółwia.

Zaczynamy od rysunku drzewa binarnego.

Wprowadzamy stopień drzewa: sam pień jest drzewem 1 stopnia, gdy dodamy dwie gałęzie, otrzymamy drzewo 2 stopnia itd. Zauważamy, że drzewo składa się z pnia i dwóch drzew niższego stopnia.

Plan – algorytm rysowania drzewa przedstawimy używając poleceń grafiki żółwia, zaczerpniętych z języka Logo:

- naprzód – np(odl),
- wstecz – ws(odl),
- pravo – pw(kąt),
- lewo – lw(kąt).



### **Algorytm rysowania drzewa binarnego** stopnia $n$ o długości pnia $d$ i kącie rozchylenia gałęzi $alf$ :

- jeśli  $n = 0$  zatrzymaj wykonywanie,
- idź rysując na odległość  $d$ , skręć w lewo o  $alf$  stopni,
- rysuj drzewo binarne stopnia  $n - 1$  o wielkości pnia 2 razy mniejszej i kącie rozchylenia  $alf$ ,
- skręć w prawo o  $2 * alf$  stopni,
- rysuj drzewo binarne stopnia  $n - 1$  o wielkości pnia 2 razy mniejszej i kącie rozchylenia  $alf$ ,
- wrót do podstawy drzewa.

Nauczyciel może teraz pokazać film ilustrujący rysowanie drzewa binarnego (drzewo.mp4).



**Implementacja grafiki żółwia** w środowisku programowania Akademii Khana.

Nauczyciel zaczyna tworzyć kod. Najpierw tłumaczy sposób definiowania funkcji:

```
var Nazwa = function(lista parametrów) {kolejne polecenia;}
```

Następnie pokazuje definicje funkcji grafiki żółwia:

```
//grafika żółwia
var np = function(odl) {
  line(0, 0, 0, - odl);
  translate(0, - odl); };
var ws = function(odl) {np(-odl);};
var pw = function(kt) {rotate(kt*PI/180);};
var lw = function(kt) {pw(-kt);};
```

Uczniowie wpisują kod i zaczynają tworzyć funkcję drzewo.

Przykładowy kod rysowania drzewa binarnego znajduje się w pliku (materiały pomocnicze 2)

Nauczyciel zadaje uczniom tematy do przerobienia w domu: działy **Text** oraz **Looping** (Pętle).

## Druga godzina lekcyjna

### **Algorytmy tworzenia wybranych fraktali (10 minut)**

Uczniowie wybierają jeden z pozostałych fraktali (trójkąt Sierpińskiego, płatek Kocha) i analizując rysunki, formułują algorytm jego rysowania, wykorzystując polecenia grafiki żółwia. Dyskutując z uczniami nauczyciel zapisuje kolejne kroki obu algorytmów:

Trójkąt Sierpińskiego składa się z trzech dwukrotnie mniejszych trójkątów Sierpińskiego umieszczonych w wierzchołkach dużego trójkąta.

**Algorytm rysowania trójkąta Sierpińskiego** stopnia  $n$  o boku  $a$ :

jeśli  $n = 0$ , zatrzymaj wykonywanie, powtórz trzy razy  
narysuj trójkąt Sierpińskiego stopnia  $n - 1$  o boku  $a/2$ ,  
przesuń się wzdłuż boku do kolejnego wierzchołka,  
obróć się o 120 stopni.

Płatek Kocha składa się z trzech krzywych Kocha.

**Algorytm rysowania krzywej Kocha** stopnia  $n$  o boku  $a$ :

jeśli  $n = 1$ , narysuj odcinek o długości  $a$  i zatrzymaj wykonywanie,  
narysuj krzywą Kocha stopnia  $n - 1$  o boku  $a/3$ ,  
obróć się w lewo o 60 stopni,  
narysuj krzywą Kocha stopnia  $n - 1$  o boku  $a/3$ ,  
obróć się w prawo o 120 stopni,  
narysuj krzywą Kocha stopnia  $n - 1$  o boku  $a/3$ ,  
obróć się w lewo o 60 stopni,  
narysuj krzywą Kocha stopnia  $n - 1$  o boku  $a/3$ .

Nauczyciel może teraz pokazać filmy ilustrujące rysowanie fraktali (Sierpinski.mp4, krzywaKocha.mp4).

### Tworzenie wybranych fraktali (25 minut)

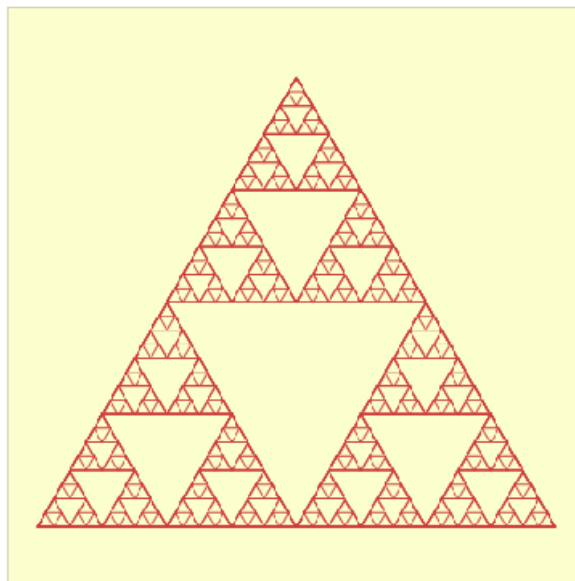
Korzystając z algorytmów uczniowie przystępują do ich kodowania w środowisku Akademii Khana.

Plan ich programów powinien być następujący:

1. utworzenie potrzebnych zmiennych,
2. zaimplementowanie grafiki żółwia,
3. zdefiniowanie funkcji rysującej fraktal według algorytmu,
4. realizacja rysunku fraktala dla wybranych wartości parametrów.

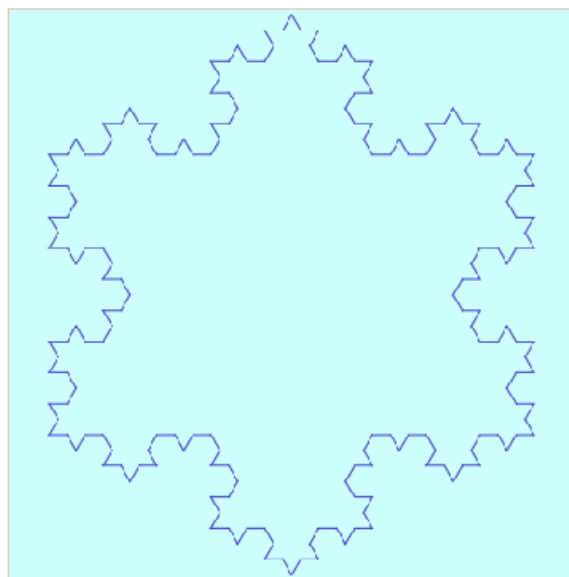
Przykładowa realizacja programu rysującego trójkąt Sierpińskiego:

```
/** Trójkąt Sierpińskiego (jak w Logo) **/  
var bok = 360; //największy bok trójkąta  
var n = 6; //stopień, liczba zagłębień  
stroke(200, 50, 50);  
//grafika żółwia  
var np = function(odl) {  
  line(0, 0, 0, -odl);  
  translate(0, -odl); };  
var ws = function(odl) {np(-odl);};  
var pw = function(kt) {rotate(kt*PI/180);};  
var lw = function(kt) {pw(-kt);};  
  
//rekurencyjny trójkąt Sierpińskiego  
var trojkatSierp = function(a, n) {  
  if (n === 0) {return;}  
  trojkatSierp(a/2,n-1); np(a); pw(120);  
  trojkatSierp(a/2,n-1); np(a); pw(120);  
  trojkatSierp(a/2,n-1); np(a); pw(120);  
};  
//realizacja  
background(250, 255, 200);  
translate(20, 360); pw(30);  
trojkatSierp(bok, n);
```



Przykładowa realizacja programu rysującego płatek Kocha:

```
/** Płatek Kocha (jak w Logo) **/  
var bok = 340; //największy bok trójkąta  
var n = 4; //stopień, liczba zagłębień  
stroke(50, 50, 200);  
//grafika żółwia  
var np = function(odl) {  
  line(0, 0, 0, -odl);  
  translate(0, -odl); };  
var ws = function(odl) {np(-odl);};  
var pw = function(kt) {rotate(kt*PI/180);};  
var lw = function(kt) {pw(-kt);};  
//rekurencyjna krzywa Kocha  
var kKocha = function(a, n) {  
  if (n < 2) {np(a); return;}  
  kKocha(a/3,n-1); lw(60);  
  kKocha(a/3,n-1); pw(120);  
  kKocha(a/3,n-1); lw(60);  
  kKocha(a/3,n-1); };  
//realizacja  
resetMatrix();  
background(200, 255, 255);  
translate(28, 298);  
pw(30);  
for(var i = 1; i < 4; i+=1) {  
  kKocha(bok, n); pw(120);}
```



Nauczyciel pomaga uczniom indywidualnie.





### Prezentacja programów (10 minut)

Uczniowie, którzy utworzyli najlepsze programy demonstrują je i opisują ich budowę oraz działanie. Jeśli zostało trochę czasu, warto zająć się szczególnymi własnościami płątka Kocha, policzyć obwód i pole powierzchni dla kolejnych przybliżeń płątka.

Obliczenia te są zawarte w pliku **PoleObwPłatka.xls**. Można zauważyć, że dla większych stopni ( $n=20$ ) obwód wyraźnie rośnie, a pole prawie się nie zmienia.

### Sprawdzenie wiedzy

Weryfikacja wiedzy polega na sprawdzeniu umiejętności rozwiązywania problemów oraz umiejętności korzystania ze środowiska programistycznego. Nauczyciel powinien w związku z tym obserwować aktywność uczniów w trakcie pracy oraz skontrolować wynik tej pracy w postaci opracowanych programów. Szczególną uwagę należy zwrócić na samodzielną pracę uczniów przy nauce programowania na platformie Akademii Khana. Na końcu ostatniej lekcji można przeprowadzić dołączony do scenariusza test.

### Ocenianie

Ocenie mogą podlegać następujące elementy:

- aktywna praca i udział w prezentacji programu oraz dyskusji,
- jakość i poprawność wykonania programu na drugiej lekcji,
- jakość i poprawność wykonania programu na trzeciej lekcji,
- aktywność ucznia w uczeniu się programowania na platformie Akademii Khana (nauczyciel widzi tematy przerabiane przez ucznia i czas, który poświęcił na ich przyswojenie),
- wynik testu.

### Dostępne pliki

1. Materiały pomocnicze 2 i 3
2. Test



### Materiały źródłowe

1. Zasoby e-learningowe Akademii Khana:
  - Drzewo binarne – [www.khanacademy.org/cs/drzewobin/2719885296](http://www.khanacademy.org/cs/drzewobin/2719885296)  
[www.khanacademy.org/cs/drzewobinlos/1014877709](http://www.khanacademy.org/cs/drzewobinlos/1014877709) (z losowością)
  - Trójkąt Sierpińskiego – [www.khanacademy.org/cs/trojksierp/2757566516](http://www.khanacademy.org/cs/trojksierp/2757566516)
  - Płatek Kocha – [www.khanacademy.org/cs/platekkocha/2762339062](http://www.khanacademy.org/cs/platekkocha/2762339062)
2. Filmy:
  - Rysowanie drzewa binarnego – drzewo.mp4
  - Rysowanie trójkąta Sierpińskiego – Sierpinski.mp4
  - Rysowanie krzywych Kocha – krzywaKocha.mp4



*Człowiek - najlepsza inwestycja*



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI



WARSZAWSKA  
WYŻSZA SZKOŁA  
INFORMATYKI

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego