Algorytmy i Struktury Danych

Prof. dr hab. Stanisław Gawiejnowicz
Wydział Biologii UAM
Semestr letni 2022/2023

Plan wykładu nr 5

- Pojęcie rekurencji
- Przykłady funkcji rekurencyjnych
- Zalety i wady rekurencji
- Graficzne zastosowania rekurencji

Pojęcie rekurencji

- Z **rekurencją** mamy do czynienia wtedy, gdy w definicji jakiegoś obiektu matematycznego (np. funkcji) występuje odwołanie do tego obiektu
- Definicje rekurencyjne są zwykle krótsze niż definicje iteracyjne (tzn. nierekurencyjne), wymagają jednak większej pamięci, a ich analiza jest bardziej złożona

Pojęcie rekurencji

- Definicja rekurencyjna składa się z dwu części:
  - pierwszej, podającej **warunek początkowy rekurencji**
  - oraz drugiej, będącej **równaniem rekurencyjnym**
- Warunek początkowy podaje wartość funkcji dla jednego (lub więcej) konkretnego argumentu
- Równanie rekurencyjne podaje zależność między parą (lub m-ką, $m \geq 2$) kolejno obliczanych wartości

Przykłady funkcji rekurencyjnych

- Klasycznym przykładem funkcji rekurencyjnej jest **n!**
- n! to iloczyn wszystkich liczb od 1 do n
- **Przykłady:**
  - 3!=1x2x3=6
  - 4!=1x2x3x4=24
  - 5!=1x2x3x4x5=120

Przykłady funkcji rekurencyjnych

- Rekurencyjna definicja n! dla $n \geq 0$ jest postaci:

```
function SILNIA(n)
if (n==0)
then return 1
else return n*SILNIA(n-1)
```

Przykłady funkcji rekurencyjnych

```
function SILNIA(n)
if (n==0)
then return 1
else return n*SILNIA(n-1)
```

- Warunek po **if** to warunek początkowy, część po **else** to równanie rekurencyjne

Przykłady funkcji rekurencyjnych

- n!! to iloczyn liczb parzystych (nieparzystych) od 2 (od 1) do n, jeśli n jest parzyste (nieparzyste)

- **Przykłady:**
  - 3!!=1x3=3
  - 4!!=2x4=8
  - 5!!=1x3x5=15

Przykłady funkcji rekurencyjnych

- Rekurencyjna definicja n!! dla n>=1 jest postaci:

```
function DSILNIA(n)
if (n==1)
then return 1
else if (n==2)
        then return 2
        else return n*DSILNIA(n-2)
```

Przykłady funkcji rekurencyjnych

- **Największy wspólny dzielnik** liczb a oraz b, nwd(a,b), to największa liczba k taka, że k dzieli a oraz k dzieli b
- Jeśli b==0, to nwd(a,b)=a
- Rekurencyjna definicja nwd(a,b):

function NWD(a,b)

if (b==0)

then return a

else return NWD(a, a **mod** b)

Przykłady funkcji rekurencyjnych

- **Współczynnik dwumianowy (Newtona)** można obliczać ze wzoru rekurencyjnego, pamiętając, że wynosi on 1 jeżeli n==k lub k==0:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

Przykłady funkcji rekurencyjnych

- Rekurencyjna funkcja obliczająca współczynnik dwumianowy dla n, k>=0 jest postaci:

function BINOM(n,k)
if (n==0) **or** (n==k)
then return 1
else return BINOM(n-1,k-1)+BINOM(n-1,k)

Zalety i wady rekurencji

- Podane przykłady sugerują iż rekurencja zawsze prowadzi do szybkich algorytmów
- **Okazuje się jednak, iż stosowanie rekurencji nie zawsze jest opłacalne**
- Istnieją funkcje rekurencyjne, których obliczenie wymaga „bardzo długiego” (formalnie: „wykładniczego”) czasu
- Spowodowane jest to dużą liczbą **wywołań rekurencyjnych**, potrzebnych do obliczenia żądanej wartości wynikowej
- Jednym z przykładów jest rekurencyjne obliczanie liczb Fibonacci'ego

Zalety i wady rekurencji

- Pseudokod rekurencyjnej funkcji FIBONACCI_3(n) obliczającej n-tą liczbę Fibonacci'ego dla n>=0 jest następujący

```
function FIBONACCI_3(n)
if (n==0)
then return 1
else if (n==1)
        then return 1
        else return FIBONACCI_3(n-1) +
                    FIBONACCI_3(n-2)
```

Zalety i wady rekurencji

- Okazuje się, iż funkcja FIBONACCI_3(n) wykonuje **wykładniczą liczbę wywołań**
- **Twierdzenie** Dla n>=2 liczba wywołań algorytmu FIBONACCI_3(n) jest większa niż 2^(n/2)
- Dowód tego twierdzenia wykorzystuje **indukcję matematyczną** (względem n), zostanie przedstawiony na wykładzie dotyczącym badania **złożoności algorytmów**

Zalety i wady rekurencji

- **Istnieją także funkcje rekurencyjne, których wartości „trudno” obliczyć**
- Wilhelm Ackermann podał w 1928 roku przykład funkcji A(m,n) (nazywanej dziś **funkcją Ackermanna**), która jest jeszcze „trudniej obliczalna” niż FIBONACCI_3(n)
- **Przykład**

  A(1,0)=2,

  A(1,1)=3,

  …

  $A(4,2) \approx 2 \times 10^{19728}$

Zalety i wady rekurencji

- Funkcja Ackermanna A(m,n) jest zdefiniowana dla m,n >=0 następująco:

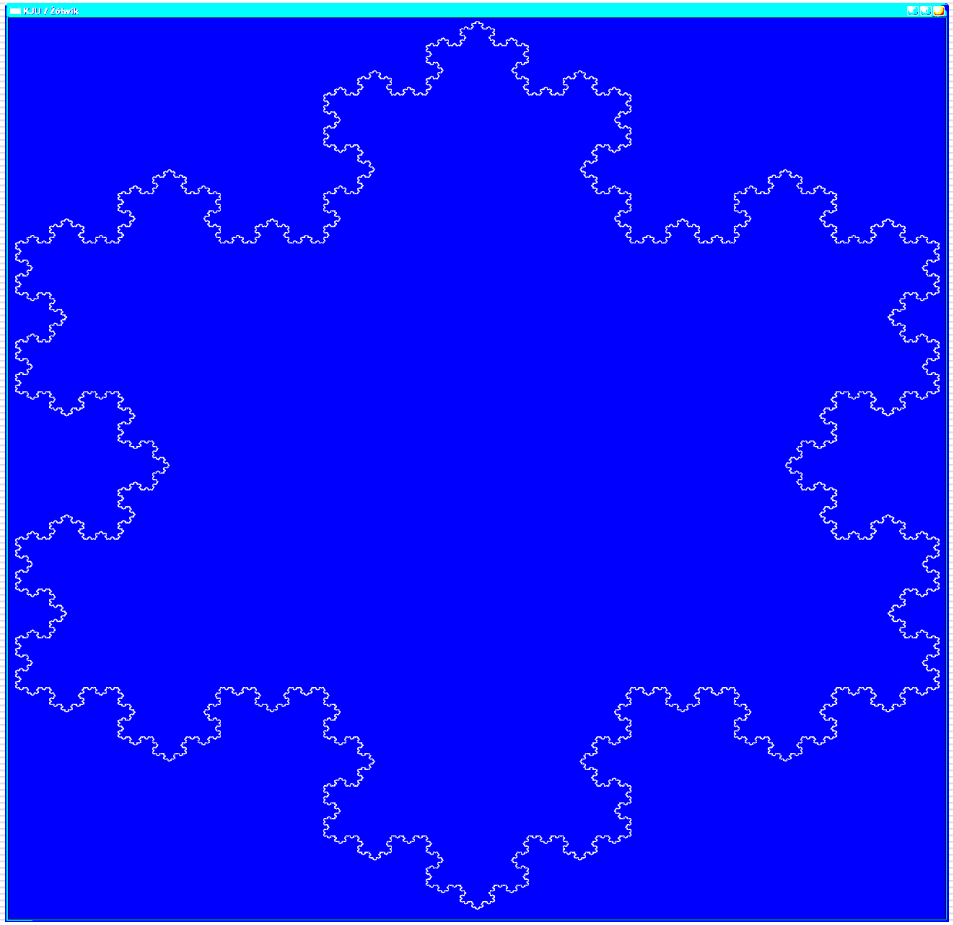
```
function A(m,n)
if (n==0)
then return m+1
else if (n≠0) and (m==0)
        then return A(n-1,1)
        else return A(n-1,A(n,m-1))
```

Graficzne zastosowania rekurencji

- Rekurencja ma niekiedy zastosowanie w grafice, do tworzenia **obiektów samopodobnych** (**fraktali**)
- Przykładami fraktali są:
  - płatek Kocha (1904)
  - trójkąt Sierpińskiego (1915)
  - zbiór Mandelbrota (1980)

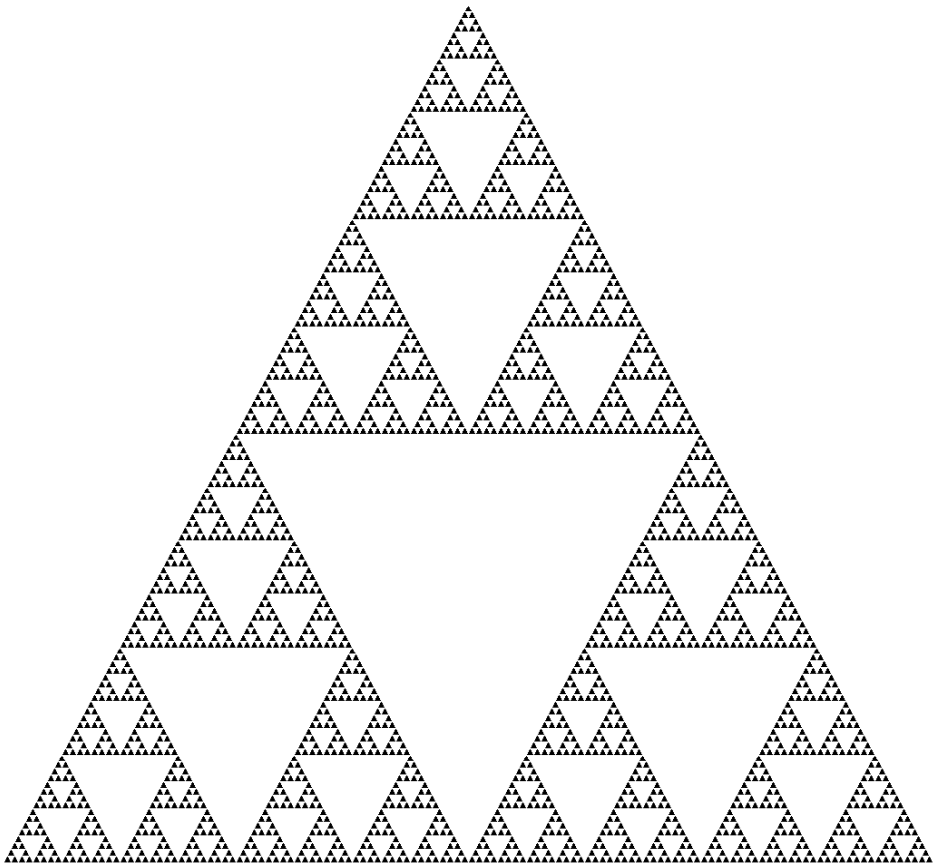
Graficzne zastosowania rekurencji

- Płatek Kocha



Graficzne zastosowania rekurencji

- Trójkąt Sierpińskiego



Graficzne zastosowania rekurencji

- Zbiór Mandelbrota