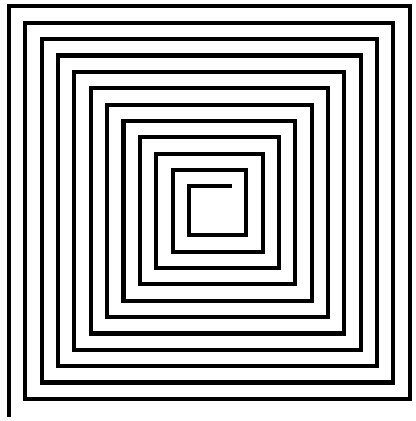
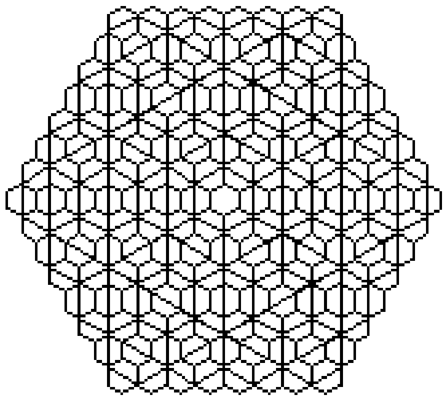
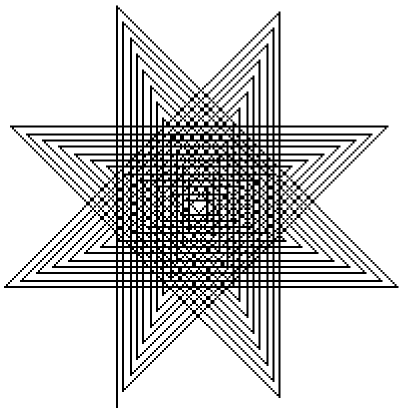
Algorytmy rekurencyjne







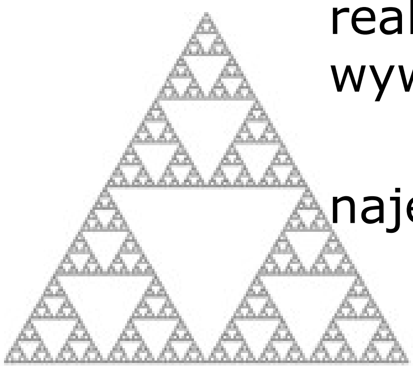
Rekurencja

Rekurencja albo **rekursja** (ang. *recursion*, z łac. *recurrere*, przybiec z powrotem) to w logice, programowaniu i w matematyce odwoływanie się np. funkcji lub definicji do samej siebie. Wbrew próbom rozróżnienia terminów **rekursja** i **rekurencja** w rzeczywistości słowa te mają identyczne znaczenie

Rekurencja polega na rozwiązywaniu problemu w oparciu o rozwiązania tego samego problemu dla danych o mniejszych rozmiarach.

W informatyce rekurencja jest to sposób rozwiązania problemu z zastosowaniem algorytmu rekurencyjnego. Jego realizacją są obliczenia, w którym wydzielony podprogram wywołuje siebie samego.

Rekurencja jest z powodzeniem stosowana w najefektywniejszych algorytmach sortowania.



Rekurencja

Rekurencyjny opis obliczeń jest na ogół bardziej **zwarty** niż opis tych samych obliczeń bez użycia rekurencji. Taki opis jest stosowany np. przy opisie **fraktali**, które są nawet definiowane jako twory podobne do swoich części. Zwartości opisu rekurencyjnego nie zawsze odpowiada jednak efektywność komputerowych realizacji algorytmów.

Rekurencja składa się z podania **wartości brzegowej** (początkowej) i z równania wyrażającego **ogólną wartość** za pomocą wartości wcześniejszych wyrazów. Np.:

$$\left.\begin{matrix} a_1 = 0 \\ a_2 = 1 \end{matrix}\right\} \quad \text{wartości brzegowe}$$

$a_{n+2} = a_n + a_{n+1}$ równanie dla $n \in N_+$

Rekurencja w naszym otoczeniu

W informatyce możemy realizować szczególny rodzaj powtórzeń bez konieczności stosowania pętli – **technikę rekurencji.**

Z techniką tą spotykamy się w życiu codziennym, jej przykładem jest odbicie w lustrze: jeśli popatrzymy w lustro, a za sobą odpowiednio ustawimy drugie, to zauważymy, że odbija się w nim obraz z lustra, które mamy przed sobą.

Kolejnym przykładem może być obraz, w którym wkomponowany jest ten sam obraz.

W każdym z tych przykładów dany **obraz jest częścią samego siebie.**

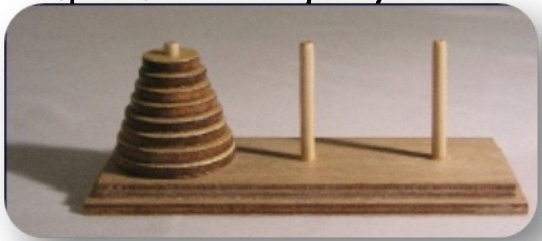
Przykładem rekurencji może być **Wieża Hanoi,** lub **rekurencyjny algorytm Euklidesa, ciąg Fibonacciego.**

Wieże Hanoi

- Wieże z Hanoi to ciekawe zadanie z algorytmiki. Rozwiązanie jest często spotykanym modelem myślenia rekurencyjnego, dlatego warto je poznać.
- Mamy *n* krążków o malejących średnicach. Każdy z nich posiada wydrążoną dziurkę i jest "nadziany" na pierwszy z trzech drążków jakie posiadamy. Pozostałe drążki są puste.

Zadanie polega na przeniesieniu wszystkich krążków z pierwszego drążka na drugi przy użyciu trzeciego. Trzeba to jednak zrobić przy dwóch założeniach:

- wolno przenosić krążki tylko pojedynczo
- ani przez moment krążek większy nie może leżeć na krążku mniejszym



Wieże Hanoi

Zagadka Wież Hanoi stała się znana w **XIX** wieku dzięki matematykowi **Édouardowi Lucasowi**, który proponował zagadkę dla 8 krążków.

Do sprzedawanego zestawu była dołączona (prawdopodobnie wymyślona przez Lucasa) **tybetańska legenda**, według której mnisi w świątyni Brahmy rozwiązują tę łamigłówkę dla 64 złotych krążków.

Legenda mówi, że gdy mnisi zakończą zadanie, nastąpi koniec świata.

Zakładając, że wykonują 1 ruch na sekundę, ułożenie wieży zajmie $2^{64}-1$ = 18 446 744 073 709 551 615 (blisko 18 i pół tryliona) sekund, czyli około 584 542 miliardów lat.

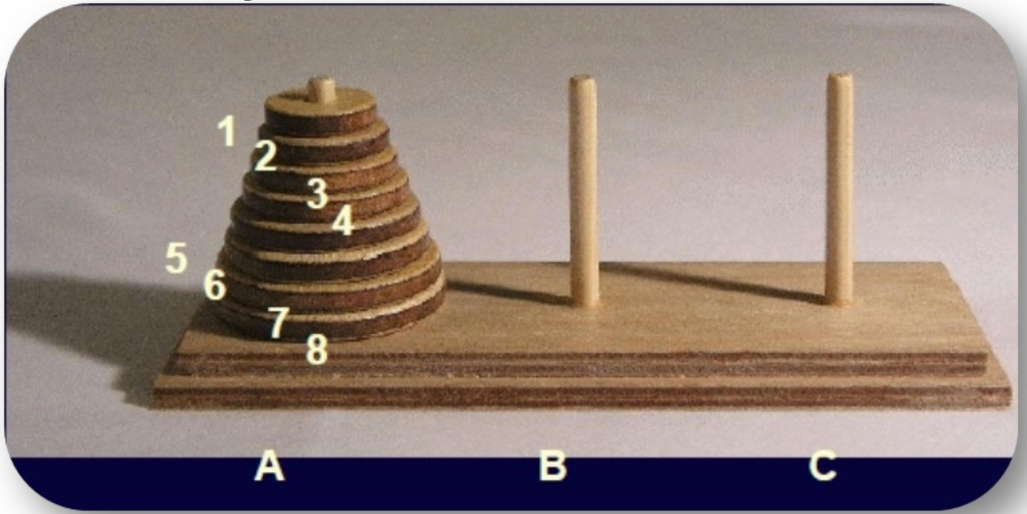
Dla porównania: Wszechświat ma około 13,7 mld lat.

Wieże Hanoi- algorytm rekurencyjny

- oznaczmy podstawki przez **A, B, C**
- niech **n oznacza liczbę krążków**
- ponumerujmy krążki od najmniejszego u góry do największego u dołu

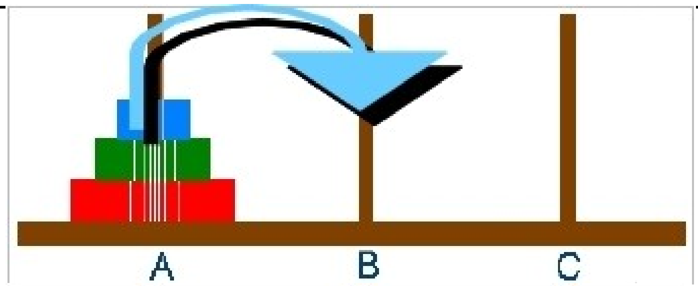
W celu przeniesienia **n** krążków z **A do B** należy:

- przenieść **n-1** krążków z **A** do **C** – wówczas n-ty dysk samotnie pozostaje w **A**
- przenieść n-ty (największy) krążek z **A do B**
- przenieść **n-1 krążków z C do B**

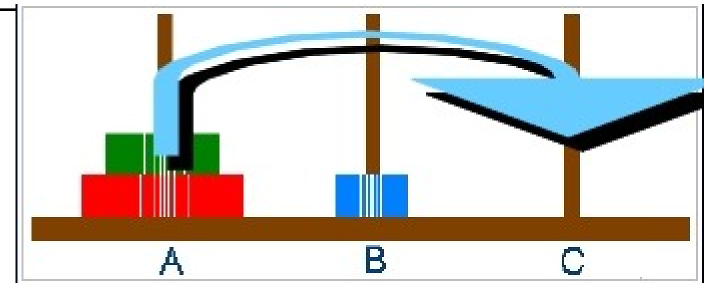


Wieże Hanoi- algorytm rekurencyjny

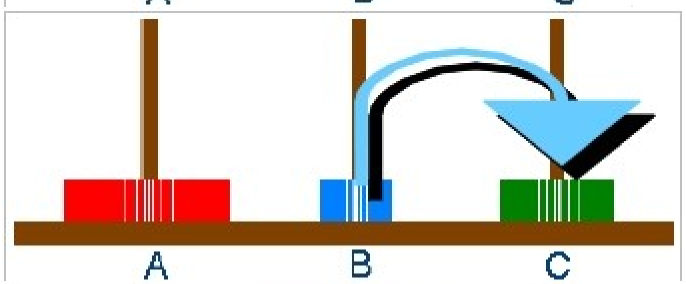
1.



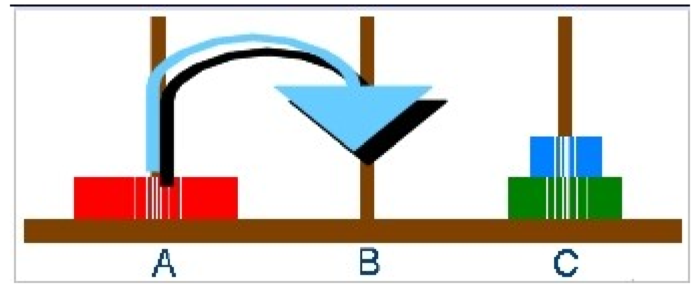
2.



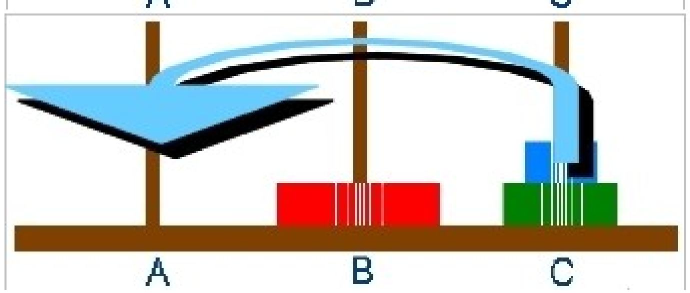
3.



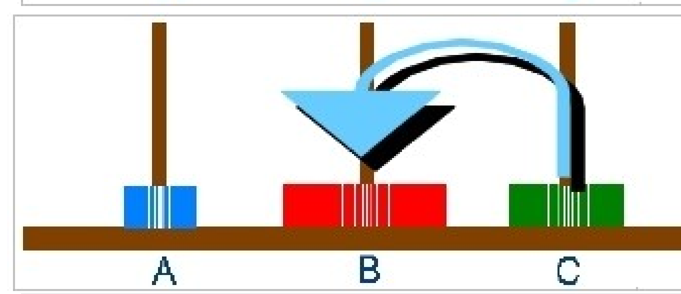
4.



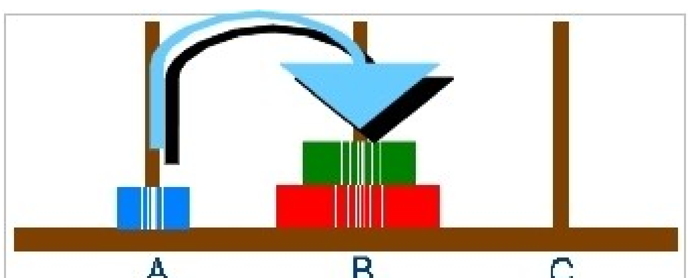
5.



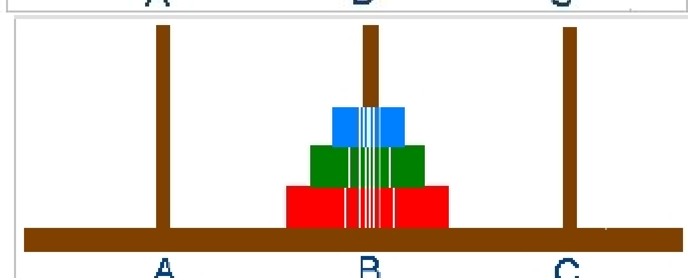
6.



7.



8.



Wieże Hanoi- algorytm rekurencyjny

przez **X→Y** oznaczmy czynność przenoszenia krążka z położenia **X** do położenia **Y,**

w takim razie zadanie wykonane na poprzednim slajdzie można zapisać następująco:

A→B	**A→C**	**B→C**	**A→B**	**C→A**	**C→B**	**A→B**
1	2	3	4	5	6	7

Dane wejściowe do algorytmu to liczba krążków **n**, a danymi wyjściowymi jest lista ruchów **X** → **Y,** które należy wykonać aby rozwiązać zadanie.

Lista ruchów dla **n=3** wygląda tak jak powyżej.

Dla **n=3** wymaganych jest $\mathbf{2^n - 1 = 7}$ ruchów.

Algorytm Euklidesa

Kolejnym przykładem algorytmu rekurencyjnego jest **algorytm Euklidesa**. Co ciekawe algorytmu nie wymyślił Euklides, a **Eudoksos z Knidos** grecki astronom, matematyk, filozof i geograf żyjący w drugiej połowie IV wieku p.n.e.

Euklides **z Aleksandrii** – matematyk grecki pochodzący z Aten, przez większość życia działający w Aleksandrii, jedynie algorytm ten zawarł i udowodnił jego poprawność w swoim dziele *Elementy*

Algorytm ten - **Największy Wspólny Dzielnik** (**NWD**) dwóch liczb jest największą liczbą naturalną spośród tych, które dzielą obie te liczby bez reszty.

W codziennej praktyce NWD służy nam do skracania ułamków do postaci właściwej.

Eudoksos z Knidos ok. 408-355 p.n.e.

Euklides z Aleksandrii ok. 365-300 p.n.e.

Algorytm Euklidesa

W algorytmie wykorzystywana jest zależność:

$$NWD(a,b) = \begin{cases} a & \text{dla } b = 0 \\ NWD(b, a \bmod b) & \text{dla } b \geqslant 1 \end{cases}$$

Dane:

a, b – dwie liczby naturalne

Dane wyjściowe:

liczba naturalna będąca największym wspólnym dzielnikiem liczb *a* i b.

W pierwszej sekcji algorytmu sprawdzamy czy *a*>b i jeżeli tak jest to zamieniamy miejscami wartości zmiennych *a* i *b*, aby był spełniony wymagany warunek ***a<=b***

Algorytm Euklidesa

- Sprawdzamy, czy $b \neq 0$, jeśli tak, to następne czynności wykonujemy aż do momentu gdy $b=0$

  $c = (b, a \bmod b)$

  $a = b$

  $b = c$

- **mod** - kongruencja modulo n (b, a mod b) – zapis oznacza, że różnica $a - b$ dzieli się bez reszty przez n
- wynik jest w zmiennej a

Przykład

NWD(1071, 1029) = *NWD*(1029, 1071 mod 1029 = 42) =
NWD(1029, 42) = *NWD*(42, 1029 mod 42 = 21) =
NWD(42, 21) = *NWD*(21, 42 mod 21 = 0)
Stąd: *NWD*(1071, 1029 = 21

Ciąg Fibonacciego

Kolejnym przykładem algorytmu rekurencyjnego jest obliczanie elementów ciągu Fibonacciego.

Fibonacci (**Leonardo z Pizy**) – był włoskim matematykiem, twierdził, że *„matematyka jest wszędzie – w całym otaczającym nas świecie...”*

W Australii w okresie jej kolonizowania plagą stały się króliki. Mimo, że w ogóle nie występowały na tym kontynencie przed przybyciem Europejczyków, już w parę lat po ich sprowadzeniu króliki zdziczały i rozmnożyły się tak bardzo, iż zaczęły zagrażać pastwiskom owiec.

Ich płodność musiała być przysłowiowa, skoro już w 1202 roku **Leonardo Fibonacci** zajął się tym problemem.

Leonardo Fibonacci
ur. ok. 1175 r. - 1250 r.

Ciąg Fibonacciego

Każda para królików staje się płodna po miesiącu.

Króliki rodzą co miesiąc nową parę.

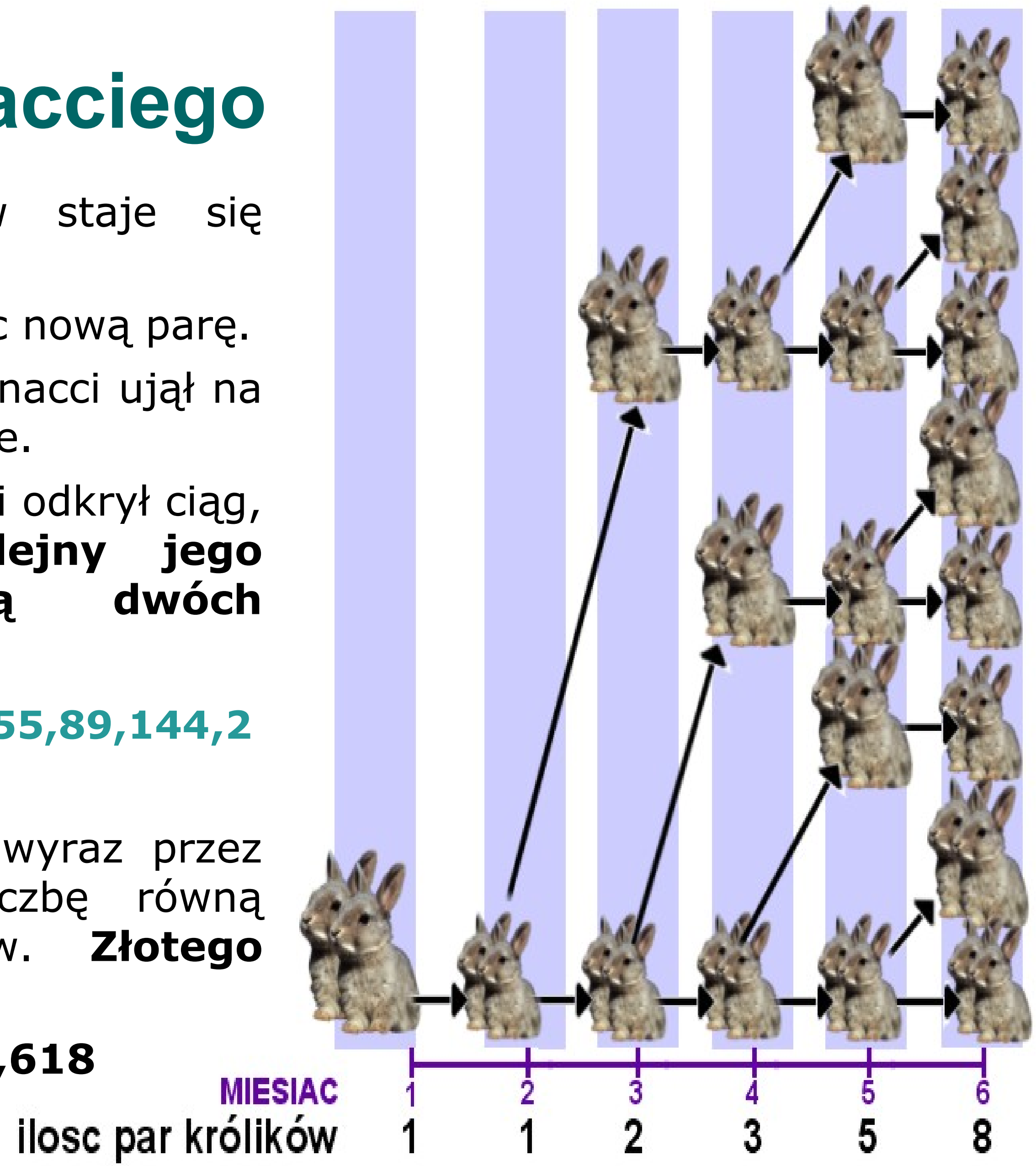
Etapy ich rozwoju Fibonacci ujął na przedstawionym schemacie.

W ten sposób Fibonacci odkrył ciąg, w którym **każdy kolejny jego wyraz, jest sumą dwóch poprzednich**.

1,1,2,3,5,8,13,21,34,55,89,144,233…

Zauważył, że dzieląc wyraz przez poprzedni, otrzymuje liczbę równą **Φ**(fi) – znaną z tzw. **Złotego podziału**

233/144 **=> 1,618**



Ciąg Fibonacciego – złoty podział

I odwrotnie (poprzedni wyraz dzielimy przez następ

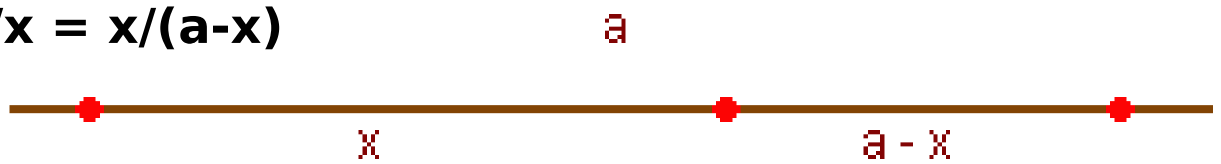
89/144 **=> 0,618**

Z kolei dzieląc wyraz przez drugi po nim(lub przed nim) itp, otrzymujemy pewne poziomy:

89/233 **=> 0,382** 55/233 **=> 0,236**

233/89 **=> 2,62** 233/55 **=> 4,236**

Złoty podział odcinka jest to wewnętrzny podział tego odcinka na dwie nierówne części tak, aby stosunek odcinka a do jego większej części x był równy stosunkowi części x do części mniejszej (a-x), tj. aby **a/x = x/(a-x)**

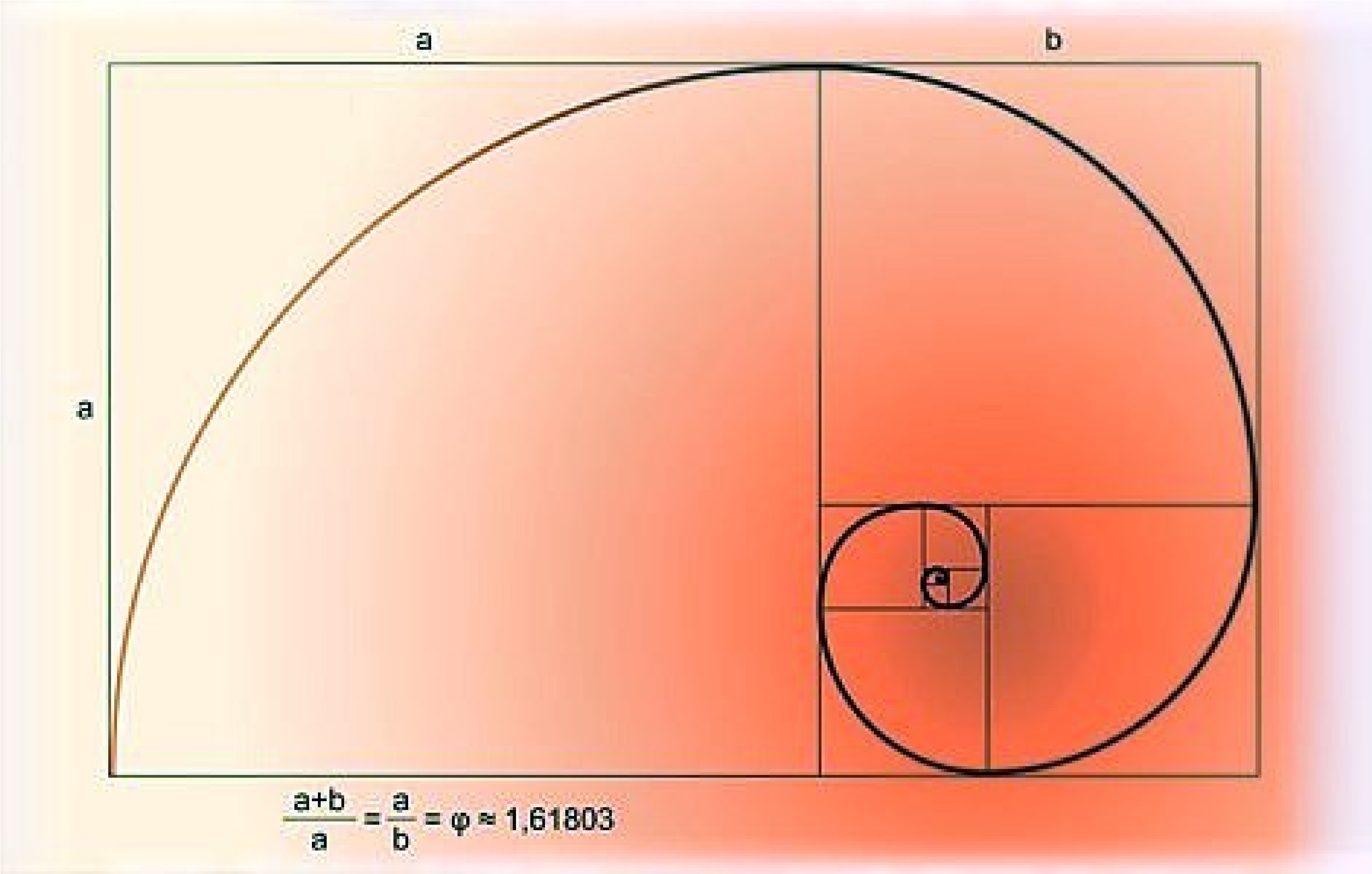


a/x = Φ = 1,618

x/a = Φ = 0,618

Ciąg Fibonacciego – złoty podział

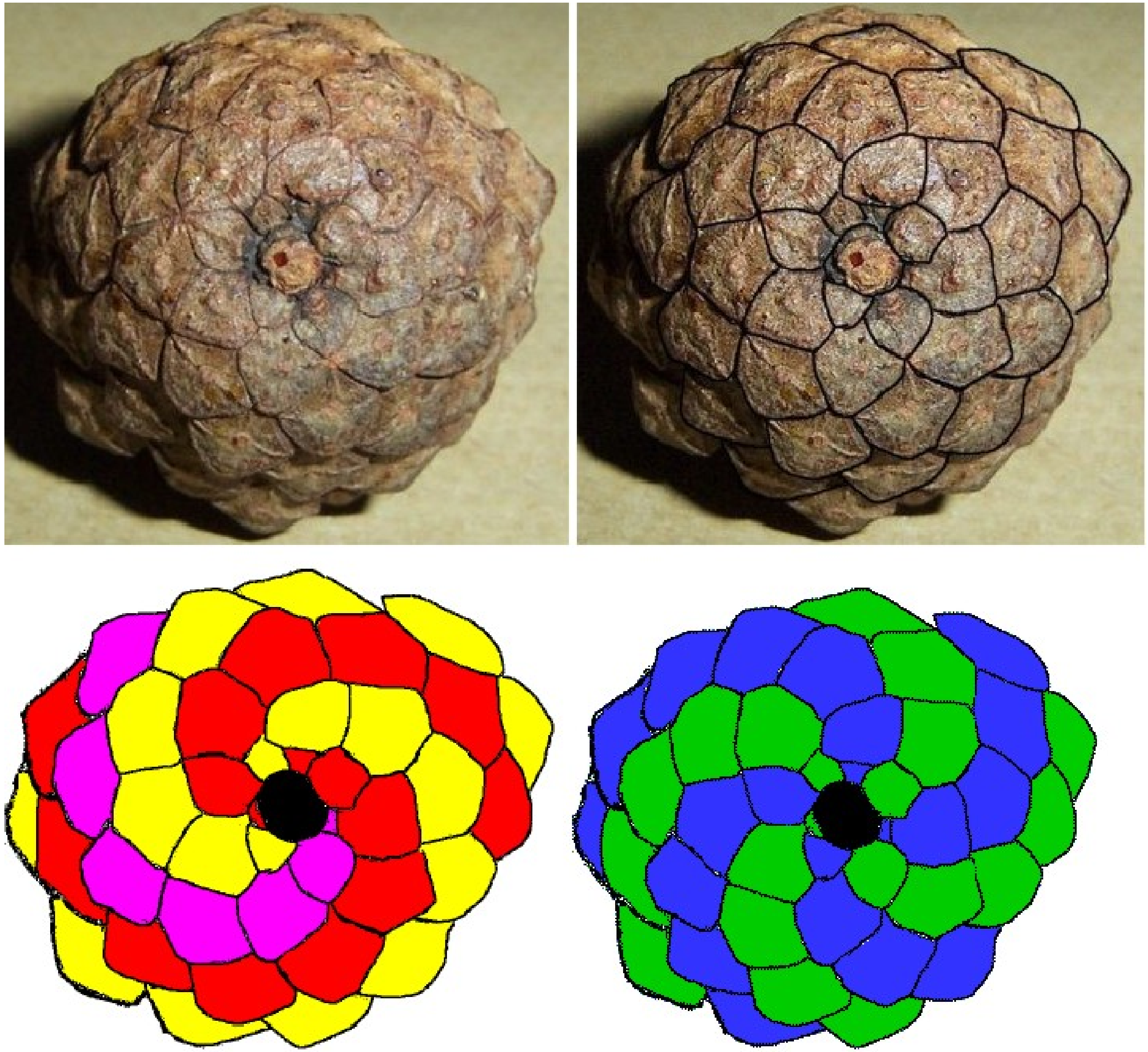
Dzięki liczbom oraz dzięki złotemu podziałowi skonstruował też spiralę



Ciąg Fibonacciego

Fibonacci nawet nie zdawał sobie sprawy, że wzmianka o ciągu który umieścił na marginesie księgi „**Liber Abaci** ” opisuje tak wiele zjawisk w przyrodzie. Ponadto ściśle wiąże się z geometrią i sztuką – **techniką rekurencji** (zjawiska oparte na nim sprawiają, że są atrakcyjne dla ludzkich zmysłów).

Z tego powodu, spośród wszystkich ciągów geometrycznych, ciąg Fibonacciego okazał się najciekawszy i niezwykle „popularny ” w otaczającym nas świecie.



Ciąg Fibonacciego

Ciąg Fibonacciego możemy najprościej zapisać używając definicję rekurencyjną:

fib(1) = 1

fib(2) = 1

…..

fib(n) = f (n-1) + f (n-2) dla n > 2

Na schemacie przedstawiona została prezentacja graficzna ciągu Fibonacciego dla fib(5) .