

§ Obsesja dużych liczb

*Uniwersytet Pedagogiczny w Krakowie

Karol GRYSZKA*

Kiedy miałem kilka, kilkanaście lat, wraz ze starszym bratem często graliśmy w grę. Należało w swojej kolejce podać liczbę większą od wskazanej przez poprzednika. Przegrywał oczywiście ten, kto nie był w stanie podać liczby większej. Czasami ponosiła nas fantazja i mówiliśmy „nieskończoność” albo „nieskończoność plus nieskończoność”. Dziś już wiem, że nieskończoność liczbą nie jest, a działania na nieskończonościach są bardziej wyrafinowane, niż podejrzewałem. Gdyby i Tobie, drogi Czytelniku, przyszło kiedyś wymienić (albo usłyszeć) jakąś dużą liczbę, możesz sięgnąć do poniższej listy. Nie są to bowiem byle jakie liczby...

$\pi = 3,1415926535897932384$
626433832795028841971...

Jeden z przykładów dowodu twierdzenia Pitagorasa, w wersji mechanicznej, można znaleźć na okładce numeru.

Dane szacunkowe – stan na sierpień 2016.

Wspomniane 200 terabajtów to mniej więcej tyle, ile wynoszą zasoby elektroniczne Biblioteki Kongresu – amerykańskiej biblioteki w Waszyngtonie, będącej największą tego typu instytucją na świecie. Poza materiałami elektronicznymi znajduje się w niej m.in. 142 mln dokumentów, 29 mln książek, 12 mln fotografii oraz ponad pół miliona filmów. Materiały zajmują w sumie około 856 km półek.

Tylko 1/3 wszystkich orientacji rogów i tylko połowa wszystkich orientacji krawędzi jest dopuszczalna. Wśród wszystkich permutacji krawędzi i rogów (bez zmiany orientacji!) tylko połowa jest dopuszczalna (jest to tak zwany problem parzystości), stąd dzielimy przez $2 \cdot 2 \cdot 3 = 12$.

40. Choć znamy wiele miliardów cyfr rozwinięcia dziesiętnego liczby π , to pierwsze 40 wystarcza, aby zmierzyć rozmiary znanego nam Wszechświata z dokładnością do rozmiaru atomu wodoru – najmniejszego atomu występującego w przyrodzie. Dokładność jest więc iście imponująca!

370 – istnieje co najmniej tyle różnych metod dowodzenia Twierdzenia Pitagorasa. Jest to rekordzista dowodowy wśród twierdzeń matematycznych.

7 400 000 000 – tylu jest w przybliżeniu mieszkańców planety Ziemia. Dokładna liczba nie może być podana, gdyż, po pierwsze, nie jest znana, a po drugie, ulega częstej zmianie (statystycznie kilka razy na sekundę). Według szacunków ONZ liczba ta do końca wieku wzrośnie do około 11 miliardów.

106 000 000 000 ($106 \cdot 10^9$) – szacunkowa liczba wszystkich ludzi, jacy kiedykolwiek żyli na Ziemi. Zaczynamy nasze obliczenia dziesiątki tysięcy lat wstecz i uwzględniamy wszystkie osoby zmarłe aż do dnia dzisiejszego – oczywiście nie zapominamy o ponad siedmiu miliardach żyjących obecnie. Zauważmy, że aktualnie żyje aż 7% wszystkich osób, jakie kiedykolwiek się urodziły.

219 902 325 555 200 bajtów (około 200 terabajtów) – takiej mniej więcej ilości danych użyto do rozstrzygnięcia hipotezy *dwukolorowości trójek pitagorejskich*. Zajęło to w sumie 35 000 godzin pracy komputerów – na szczęście pracowało nad tym wiele komputerów równocześnie. Treść problemu jest następująca: czy jest możliwe pokolorowanie każdej liczby naturalnej na czerwono lub niebiesko w taki sposób, by żadna trójka a, b, c , spełniająca równanie $a^2 + b^2 = c^2$, nie była jednokolorowa. Na przykład, jeżeli 5 i 12 są niebieskie, to 13 musi być czerwone. Hipoteza ta jest fałszywa, daje się we wskazany sposób pokolorować liczby naturalne od 1 do 7824 (nawet na parę sposobów), ale gdy dołączymy liczbę 7825, odpowiednie kolorowanie nie istnieje co sumiennie sprawdziły komputery.

43 252 003 274 489 856 000 (około 43 tryliony lub $43 \cdot 10^{18}$) – to wszystkie możliwe konfiguracje elementów na kostce Rubika. Na całym świecie oryginalną kostkę oraz jej warianty sprzedano w kilkuset milionach egzemplarzy, co czyni ją najpopularniejszą zabawką w historii.

Jak wyznaczyć liczbę konfiguracji? Każdy róg (element trój kolorowy) może być ustawiony w jednym z ośmiu miejsc i w jednej z trzech orientacji, co daje łącznie $8! \cdot 3^8$ możliwości. Każda krawędź (element dwukolorowy) może być ustawiona w jednym z dwunastu miejsc i w jednej z dwóch orientacji – łącznie $12! \cdot 2^{12}$ ustawień. Każdy środek (element jednokolorowy) ma stałe położenie względem ustalonej orientacji, wobec tego jest tylko jeden istotny sposób na ich ustawienie. Ostatecznie otrzymujemy liczbę $8! \cdot 12! \cdot 2^{12} \cdot 3^8$.

Ale nie jest to poprawna odpowiedź. Istnieją ograniczenia narzucane przez mechanizmy, które redukują liczbę dopuszczalnych możliwości.

Ostateczna liczba zmniejsza się dwunastokrotnie, a $8! \cdot 11! \cdot 2^{12} \cdot 3^8$ to właśnie 43 252 003 274 489 856 000.

808 017 424 794 512 875 886 459 904 961 710 757 005 754 368 000 000 000 (około 808 oktyliardów lub $808 \cdot 10^{51}$). W XX wieku ważnym zagadnieniem w matematyce (dokładniej teorii grup) była kwestia opisanego, klasyfikacji pewnych obiektów (tak zwanych skończonych grup prostych, cokolwiek to oznacza).

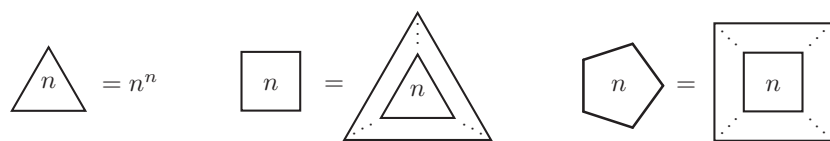
Pełna klasyfikacja została osiągnięta nakładem pracy wielu matematyków oraz ponad dziesięciu tysięcy stron publikacji. Jej efektem było wyróżnienie „cegiełek”, elementów, z których można zbudować wszystkie obiekty klasyfikowane. Okazało się również, że istnieje największa taka cegiełka – grupa mająca tyle elementów, ile wynosi liczba otwierająca poprzedni akapit. Grupa ta nazywa się po angielsku *Monster group*, co możemy dość swobodnie przetłumaczyć na *grupę potworną* (żartobliwie i pieszczotliwie nazywaną również *potworkiem*).

10^{80} (100 tridecyliardów). Szacowana liczba atomów w obserwowalnym Wszechświecie. Nie jesteśmy w stanie dokładnie ich policzyć, bazujemy jedynie na średniej gęstości materii w stosunku do znanych rozmiarów Wszechświata i na tej podstawie wyznaczamy wspomnianą liczbę. Jak ogromna jest to liczba? Przypuśćmy, że atomy ustawiliśmy w linii prostej. Każdy z nich ma średnicę około jednego nanometra (czyli 10^{-9} metra). Wszystkie utworzą linię długości około 10^{68} kilometrów czyli 10^{55} lat świetlnych.

10^{100} (10 seksdecyliardów lub *googol*) razem z 10^{googol} , czyli *googolplexem* zdobyły popularność w różnych teleturniejach. W brytyjskim wydaniu „Milionerów” w pytaniu za milion funtów padło: Jak jest nazywana liczba 1 ze stoma zerami?

$2^{74207281} - 1$ oznaczana także jako $M_{74207281}$. Jest największą znaną liczbą pierwszą, składa się z 22 338 618 cyfr. Gdybyśmy wydrukowali tę liczbę na papierze, mieszcząc 50 linii na stronie i 100 cyfr w każdej linii, zadrukowalibyśmy 4 468 stron. Taki plik kartek miałby około pół metra grubości i ważył kilka kilogramów. Papierowa wersja została zaprezentowana przez Matta Parkera na kanale YouTube „Numberphile”.

Mega i Liczba Mosera. Wprowadzimy notację, pochodząca od Hugona Steinhausa, rozszerzoną później przez Leo Mosera.



Liczba n w kwadracie oznacza liczbę n w n zagnieżdżonych trójkątach (w wieży potęgowej jest 2^n takich samych liczb). Podobnie w dalszej części mamy n wzajemnie zagnieżdżonych kwadratów. Ogólnie można zdefiniować liczbę n wpisaną w wielokąt o k bokach jako liczbę n wpisaną w n zagnieżdżonych wielokątach o $k - 1$ bokach.

Steinhaus upodobał sobie szczególnie: *mega* – dwójka w pięciokącie – oraz *megiston* – dziesiątka w pięciokącie. Te dwie liczby czasami zamiast w pięciokącie zapisuje się w okręgu.

Liczba Mosera to 2 w wielokącie, którego liczba boków jest równa liczbie mega (wielokąt taki nazywamy *megagonem*). Żeby móc porównać te liczby z dotychczasowymi, podamy oszacowanie (choćby grube). Jeżeli $a \uparrow \uparrow b$ oznacza $a^{a^{\dots^a}}$, gdzie wieża potęgowa liczb a zawiera b liczb, to

$$10 \uparrow \uparrow 257 < \text{mega} < 10 \uparrow \uparrow 258.$$

Prędkość światła w próżni to w przybliżeniu 300 000 km/s, a rok juliański to 31 557 600 sekund. Ich iloczyn to 9 467 280 000 000 km, czyli w przybliżeniu 10^{13} kilometrów na każdy rok. Wielkość ta nosi nazwę *roku świetlnego*.

Ogólnie liczby postaci $2^p - 1$, gdzie p jest liczbą pierwszą, nazywamy *liczbami Mersenne'a*.

Liczba cyfr liczby naturalnej n wynosi $\lceil \log_{10} n \rceil + 1$. Wyznaczmy, ile cyfr ma liczba $M_{74207281}$:

$$\begin{aligned} \log_{10} 2^{74207281} &= 74207281 \cdot \log_{10} 2 = \\ &= 74207281 \cdot 0,30102999566398 \dots = \\ &= 22338617,47766 \dots \end{aligned}$$

Zatem największa znana liczba pierwsza ma

$$\lceil 22338617,47766 \dots \rceil + 1 = 22338618 \text{ cyfr}$$

($\lceil x \rceil$ oznacza część całkowitą z liczby x). Od początkowej liczby nie zostało odjęte 1, ale nie ma to wpływu na liczbę cyfr.

$$\boxed{2} = \triangle \triangle \triangle = \triangle \triangle \triangle = 4^4$$

$$\boxed{3} = ?$$

$$\boxed{4} = ?$$

Liczba Grahama. Rozszerzamy notację strzałkową, będziemy pisali

$$\uparrow^n = \underbrace{\uparrow \dots \uparrow}_n,$$

gdzie

$$a \uparrow^n b = \begin{cases} ab, & \text{dla } n = 0, \\ 1 & \text{dla } n \geq 0 \text{ oraz } b = 0, \\ a \uparrow^n b = \underbrace{a \uparrow^{n-1} (a \uparrow^{n-1} (\dots \uparrow^{n-1} a))}_{b \text{ kopii } a} & \text{w przeciwnym wypadku.} \end{cases}$$

$$\begin{aligned} 3 \uparrow^3 3 &= 3 \uparrow^2 (3 \uparrow^2 3) \\ 3 \uparrow^2 3 &= 3^{3^3} = 3^{27} = 7\,625\,597\,484\,987 \\ a \uparrow^4 b &= \underbrace{a \uparrow^3 (a \uparrow^3 (\dots \uparrow^3 a))}_{b \text{ kopii } a} \end{aligned}$$

Niech teraz $f(n) = 3 \uparrow^n 3$, wtedy **liczba Grahama** zdefiniowana jest jako $G = f^{64}(4)$, gdzie potęga przy funkcji oznacza liczbę powtórzeń. Liczba ta wystąpiła w badaniach nad uogólnionym problemem Ramseya.

Ciekawostka: G jest „znacznie” większe od liczby Mosera. Pokazano, że ta ostatnia nie jest większa od $f^3(4)$.

Posłowie I: Naprawdę duże liczby

Pojęciem dualnym do wspomnianego obok jest *złożoność Kolmogorowa* (od Andrieja Kolmogorowa) liczby $n \in \mathbb{N}$ – długość najkrótszego programu, który zwraca liczbę n .

Jaką największą liczbę może zwrócić program długości co najwyżej k (o co najwyżej k znakach)?

Przez $\text{gener}(k)$ oznaczmy odpowiedź na powyższe pytanie ($k \in \mathbb{N}$). Program, który ma 100 znaków, może zwrócić liczbę zdecydowanie większą niż 100. Program złożony z 1000 znaków może zwrócić liczbę bardzo zdecydowanie większą niż 1000, ale jak bardzo zdecydowanie? Jak szybko rosną możliwości programu wraz ze wzrostem liczby jego znaków?

Znaków, których możemy użyć do napisania programów, jest skończenie wiele. Stąd programów o długości co najwyżej k również jest skończenie wiele, a tylko niektóre z nich będą działać poprawnie i zwracać liczby. Niewątpliwie któryś z nich wygeneruje liczbę największą.

Może zdarzyć się, że żaden program o długości co najwyżej k nie zwraca liczby. W takiej sytuacji ustalmy, że $\text{gener}(k) = 0$. To się zdarzy tylko dla małych k .

Okazuje się, że $\text{gener}(i)$ dla niedużych $i \in \mathbb{N}$ są naprawdę pokaźne, przy nich liczby Mosera lub Grahama to zupełne mikrusy. Co ciekawe, kiedy wybierzemy konkretne i , wartość $\text{gener}(i)$ jest *nieobliczalna*, nie da się obliczyć jej dokładnej wartości – wyjaśnienie w tekście na sąsiedniej stronie. Można natomiast pokazać, od jakiej liczby $\text{gener}(i)$ na pewno jest większe – stąd śmiałość w nazywaniu liczby Mosera mikrusem. Przyjrzyjmy się poniższemu programowi.

```

1 Function arrow(a, n, b, k)
2   if (k > 1) then
3     | return arrow(a, n, arrow(a, n, b, k - 1), 1)
4   else if (n == 0) then
5     | return ab
6   else if (b == 0) then
7     | return 1
8   else
9     | return arrow(a, n - 1, a, b - 1)
10 return arrow(3, 4, 3, 64)

```

Funkcja $\text{arrow}(a, n, b, k)$ zwraca liczbę $\underbrace{a \uparrow^n (a \uparrow^n (\dots (a \uparrow^n b) \dots))}_{k \text{ strzałek}$. Przyjrzyjmy

się dlaczego tak jest. Linie 4-9 rozpisują definicję notacji strzałkowej, opisanej we wcześniejszym artykule. Co się dokładnie dzieje?

- Linie 2-3 – rozważamy przypadek, gdy liczba strzałek k jest większa niż 1. Wtedy trzeba zrobić jedną strzałkę, a potem jeszcze $k - 1$.
- Zachodzi $a \uparrow^0 b = ab$, co obsługują linie 4-5 oraz $a \uparrow^n 0 = 1$, co obsługują linie 6-7.
- $a \uparrow^n b = \underbrace{a \uparrow^{n-1} (a \uparrow^{n-1} (\dots (a \uparrow^{n-1} a) \dots))}_{b - 1 \text{ strzałek}}$, co implementują linie 8-9.



W powyższym programie, po zdefiniowaniu funkcji *arrow*, w ostatniej linii pytamy program o wartość $arrow(3, 4, 3, 64)$, czyli o liczbę Grahama. Program ma dokładnie 201 znaków. A więc wiadomo, że $gener(201) \geq G$ – być może istnieje program o takiej długości zwracający większą liczbę. A co powiecie o $gener(G)$?

W. C.

Posłowie II: Dowód nierozstrzygalności

Spróbujmy ściśle uzasadnić, że funkcja $gener$, o której mowa wyżej, jest faktycznie *nieobliczalna*. Bardziej formalnie: chcemy udowodnić, że nie może istnieć program komputerowy P taki, który dla danej na wejściu liczby n obliczy $gener(n)$.

Nasze rozumowanie prowadzić będziemy nie wprost. Zakładamy więc, że jednak taki program P istnieje. Jego długość (okaże się bardzo ważna) oznaczmy przez k . (To znaczy: kod programu P jest zapisany jako ciąg k znaków.)

Rozważmy teraz program (a raczej *szablon* programu) Q_n zdefiniowany następująco:

```

1 Function Z(i)
2 | kod programu P
3 return Z([wpisana na sztywno stała n])

```

Dla jasności czym jest szablon, napiszmy kod *konkretnego* programu Q_{83} :

```

1 Function Z(i)
2 | kod programu P
3 return Z(83)

```

Programy z szablonu Q_n są dziwaczne: mają wbudowany program P jako wewnętrzną funkcję Z ; nie przyjmują żadnego wejścia oraz w swoim kodzie mają wpisaną „na sztywno” stałą n . Zastanówmy się, jaką długość ma sam program Q_n . W kod Q_n wchodzi kod P , więc tutaj zużywamy k znaków. Poza tym mamy jakąś niewielką liczbę innych znaków (jak „r”, „e”, „t”, „u”, „r”, „n” itd.) oraz stałą, której zapis zużyje $\lceil \log_{10} n \rceil$ znaków „0”, „1”, „2”, ..., „9”. Łącznie: $k + c + \lceil \log_{10} n \rceil$ znaków dla pewnego $c \in \mathbb{N}$.

Jak widać, program Q_n oblicza liczbę $P(n)$. A zatem mamy, że $gener(k + c + \lceil \log_{10} n \rceil) \geq P(n)$ oraz, oczywiście, $gener(n) = P(n)$. Jednak funkcja $gener$ jest niemalejąca, czyli musiałyby być $k + \lceil \log_{10} n \rceil + c \geq n$ dla dowolnego n , a to jest, oczywiście, sprzeczność dla dostatecznie dużych n .

T. K.

Posłowie III: Ortografia

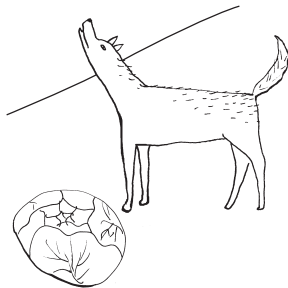
W tekście „Naprawdę duże liczby” zostało użyte następujące rozumowanie: *Skoro znaków używanych do napisania programu jest skończenie wiele i skoro program ma ograniczoną długość, to liczb definiowanych przez ten program może być też tylko skończenie wiele, a więc wśród nich musi być największa.* Powstaje pytanie, jakie dodatkowe założenia muszą zostać użyte w tym rozumowaniu, by było ono poprawne i dawało prawdziwy wniosek.

O tym, że takie pytanie ma rację bytu świadczy następujący przykład rozumowania dość podobnego do przytoczonego wyżej: *W języku polskim używamy skończonej liczby liter, więc utworzonych z nich napisów ograniczonej długości jest skończenie wiele. W szczególności skończenie wiele jest napisów używających co najwyżej stu liter. Znacznie mniej (czyli też skończenie wiele) jest wśród nich napisów sensownych, a jeszcze mniej tych, które oznaczają liczbę. Liczb dających się tak zapisać jest zatem skończenie wiele, a więc jest też największa taka liczba.*

Jak to pogodzić z faktem, że napis *liczba co najmniej o jeden większa od każdej, którą można zapisać za pomocą co najwyżej stu liter* ma tylko osiemdziesiąt liter?

Owe dodatkowe założenia użyte w cytowanym tekście to reguły ortograficzne (tak, ortograficzne!), których musimy przestrzegać przy pisaniu programów.

M. K.



Liczba n ma $\lceil \log_{10} n \rceil$ cyfr w zapisie dziesiętnym.



Rozwiązanie zadania F 929.

Niech jednej fali odpowiada drgania wektora pola elektrycznego

$\mathbf{E}_1 = \mathbf{E}_{10} \cos \omega t$, a drugie

$\mathbf{E}_2 = \mathbf{E}_{20}(\cos \omega t + \delta)$, gdzie δ jest różnicą

faz między drganiami. Drganie

wypadkowe będzie dane wzorem

$\mathbf{E} = \mathbf{E}_1 + \mathbf{E}_2 = \mathbf{E}_{10} \cos \omega t + \mathbf{E}_{20} \cos(\omega t + \delta)$,

a stąd

$$\mathbf{E}_2 = \mathbf{E}_{10}^2 \cos^2 \omega t + \mathbf{E}_{20}^2 \cos^2(\omega t + \delta) + 2\mathbf{E}_{10}\mathbf{E}_{20} \cos \omega t \cos(\omega t + \delta).$$

Natężenie drgań będzie równe średniej względem czasu wartości \mathbf{E}_2 , czyli

$$I = 1/2\mathbf{E}_{10}^2 + 1/2\mathbf{E}_{20}^2 + 2\mathbf{E}_{10}\mathbf{E}_{20}(\cos \omega t \cos(\omega t + \delta)),$$

gdzie $\langle \rangle$ oznacza średnią względem czasu. Wyrażenie to będzie równe zero

niezależnie od wartości przesunięcia

fazowego δ jeżeli iloczyn skalarny $\mathbf{E}_{10}\mathbf{E}_{20}$

będzie równy zero, a to oznacza, że

kierunki drgań muszą być względem

siebie prostopadłe.