Algorytmy podzielności przez 7

*Łukasz GRZĄDKO**

* Nokia

Zapewne każdy Czytelnik *Delty* wie, jak sprawdzić, czy nawet duża liczba jest podzielna przez 3, czy przez 8. Metody tego typu wprowadzane są już w młodszych klasach szkoły podstawowej, dzięki czemu są powszechnie znane. Jednak tytułowy problem podzielności akurat przez 7 jest w typowym kursie szkolnym pomijany. W niniejszym artykule postanowiliśmy więc tę lukę uzupełnić i przedstawić przegląd różnych metod na sprawdzenie podzielności przez 7.

A więc do dzieła:

Metoda: pomnóż przez 2 i odejmij

Pojedynczy krok algorytmu jest następujący: jeśli liczba N jest co najmniej trzycyfrowa, to zastępujemy („nadpisujemy") ją liczbą:

$$\left\lfloor \frac{N}{10} \right\rfloor - 2 \cdot (N \bmod 10).$$

Gdy zmienna N stanie się dwucyfrowa, to po prostu sprawdzamy jej podzielność przez 7 wprost. Pozostawiamy Czytelnikom udowodnienie poprawności tej metody. Od strony złożoności algorytmicznej – dostajemy liniowy koszt zarówno czasowy, jak i pamięciowy.

Metoda kolejnych trójek

Skorzystamy tutaj z kongruencji $1000 \equiv -1 \pmod 7$.

Przyjmujemy, że liczba cyfr liczby N jest podzielna przez 3, w przeciwnym razie najbardziej znaczące miejsca możemy uzupełnić zerami.

Notacja $\overline{c_{n-1}c_{n-2}c_{n-3}\ldots c_1c_0}$ oznacza n-cyfrową liczbę N o cyfrach $c_i \in \{0, 1, \ldots, 9\}$. Najmniej znaczącą cyfrą liczby N jest c_0, a najbardziej znaczącą c_{n-1}.

Możemy zatem zapisać N jako $\sum_{i=0}^{n/3-1} \overline{c_{3i+2}c_{3i+1}c_{3i}} \cdot 1000^i$. Stąd, i z powyższej kongruencji, mamy $N \equiv \sum_{i=0}^{n/3-1} \overline{c_{3i+2}c_{3i+1}c_{3i}} \cdot (-1)^i$. Metoda sprawdzenia podzielności sprowadza się więc do arytmetyki liczb trzycyfrowych. Dla przykładu liczba 5 242 636 881 jest podzielna przez 7, bo $5 - 242 + 636 - 881$ jest podzielna przez 7.

Dowód poprawności metody kolejnych trójek można znaleźć m.in. w książce Wacława Sierpińskiego „Teoria Liczb", t. 1. Zauważmy, że rozumowanie w dowodzie działa również dla podzielności przez 11 oraz 13, gdyż $1001 = 7 \cdot 11 \cdot 13$.

Metoda potęgowania trójki

Kolejny sposób bazuje na poniższej obserwacji:

Niech

$$(*) \qquad R = \sum_{i=0}^{n-1} c_i \cdot 3^i.$$

Wówczas N jest podzielna przez 7 wtedy i tylko wtedy, gdy R jest podzielna przez 7.

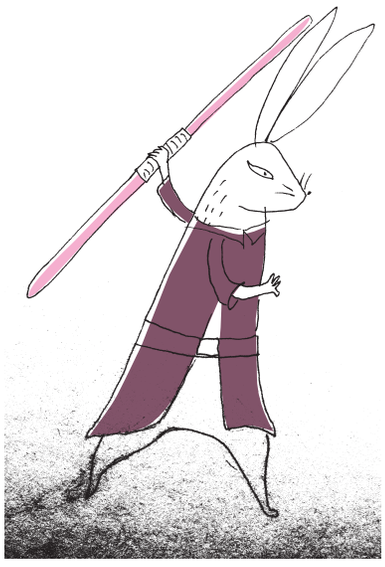
Powyższe stwierdzenie wynika wprost z faktu, że dla każdego całkowitego $i \geqslant 0$, $3^i \equiv 10^i \pmod 7$.

Potęgowanie jest dość czasochłonne, ale ponieważ interesuje nas tylko reszta z dzielenia, więc możemy odpowiednie potęgi 3 zastąpić odpowiednim wynikiem modulo 7:

$$3^0 \equiv 1 \pmod 7, \quad 3^1 \equiv 3 \pmod 7, \quad 3^2 \equiv 2 \pmod 7, \quad 3^3 \equiv 6 \pmod 7,$$
$$3^4 \equiv 4 \pmod 7, \quad 3^5 \equiv 5 \pmod 7, \quad 3^6 \equiv 1 \pmod 7, \quad \ldots$$

(Kolejne reszty pojawiają się cyklicznie w cyklu długości 6.)

Możemy zatem zapamiętać permutację $(1, 3, 2, 6, 4, 5)$ i podstawiać cyklicznie jej elementy w miejsce kolejnego mnożnika 3^i we wzorze $(*)$, i na końcu sprawdzić, czy otrzymana liczba dzieli się przez 7. Co ciekawe, każde przesunięcie cyklu również poprawnie rozstrzygnie podzielność przez 7, tj. możemy mnożyć kolejne cyfry przez np. $(3, 2, 6, 4, 5, 1)$ czy $(2, 6, 4, 5, 1, 3)$. Ta ważna własność – która będzie kluczowa również w rozumowaniu pod koniec tego tekstu – wynika z tego, że dla



dowolnego naturalnego k liczba $3^k \cdot R$ jest podzielna przez 7 wtedy i tylko wtedy, gdy R jest podzielna przez 7, ponieważ liczby 3 oraz 7 są względnie pierwsze.

Mamy zatem 6 różnych permutacji, które można zastosować równoważnie w algorytmie. Dla przykładu, żeby sprawdzić, czy liczba 12 345 678 jest podzielna przez 7, możemy sprawdzić sumę (wybraliśmy cykl $(3, 2, 6, 4, 5, 1)$):

$$3 \cdot 8 + 2 \cdot 7 + 6 \cdot 6 + 4 \cdot 5 + 5 \cdot 4 + 1 \cdot 3 + 3 \cdot 2 + 2 \cdot 1 = 125;$$

następnie $3 \cdot 5 + 2 \cdot 2 + 6 \cdot 1 = 25$. Ostatnia suma nie jest podzielna przez 7, zatem wyjściowa liczba też nie jest podzielna przez 7. Natomiast 12 345 683 jest podzielna przez 7, gdyż odpowiednia suma cyfr po analogicznym podstawieniu wynosi 112, a ta jest podzielna przez 7.

Zauważmy też, że permutację $(1, 3, 2, 6, 4, 5)$ możemy zapisać równoważnie jako $(1, 3, 2, -1, -3, -2)$, co istotnie może ułatwić zapamiętanie metody (de facto pamiętamy cykl tylko trzech liczb i pilnujemy zmiany znaku po każdym obrocie).

Jeśli metodę implementujemy na komputerze i chcemy naprawdę efektywnie to zrobić, jeszcze lepiej zapisać cykl jako $(1, -4, 2, -1, 4, -2)$, gdyż mnożenie przez małe potęgi 2 jest dla komputera wyjątkowo naturalne (ten i następny wariant algorytmu nie był znany autorowi tekstu wcześniej).

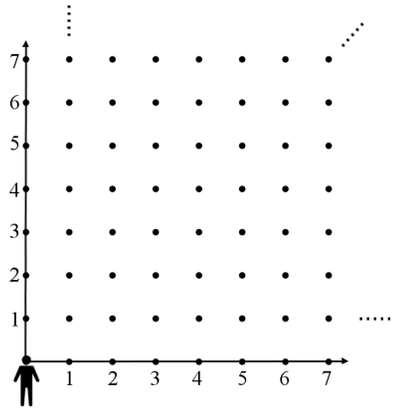
Zauważmy, że powyższa metoda może działać w stałej pamięci, jeśli liczba jest podawana na wejściu „strumieniowo” (cyfra po cyfrze) od najmniej znaczącej cyfry.

A co, gdy liczba jest podawana na wejściu od cyfry najbardziej znaczącej?

Tutaj też poradzimy sobie w stałej pamięci. Wystarczy tylko pewien cykl odwrócić (np. przyjąć $(1, -2, 4, -1, 2, -4)$) oraz postępować dalej podobnie jak w klasycznym algorytmie – łatwo sprawdzić, że wówczas obliczymy tę samą sumę, co w standardowej procedurze (choć, co ciekawe: dla pewnego – nieznanego z góry – spośród sześciu poprawnych cykli)!

Metodę z cyklem $(1, -2, 4, -1, 2, -4)$ da się wyprowadzić bezpośrednio, tym razem analizując kolejne potęgi 5 (równoważnie: (-2)), ale rosnące odwrotnie niż w klasycznym algorytmie – to znaczy od najbardziej znaczącej, a nie od najmniej znaczącej cyfry. Taka zresztą była geneza powstania oraz analizy tej metody przeprowadzona przez autora tekstu – Czytelnik Zaciekawiony może spróbować odtworzyć to rozumowanie samodzielnie.

Widoczność w nieskończonym lesie



Stoimy u progu nieskończenie milowego, nad wyraz uporządkowanego lasu. Najlepszym miejscem na uporządkowany las jest oczywiście układ współrzędnych. Pnie drzew, które są odcinkami, umieszczone są w punktach o współrzędnych całkowitych nieujemnych. Nasz wzrok z punktu $(0, 0)$, w którym drzewa nie ma, przygląda się temu zjawisku (patrz rysunek). Taki las ciągnie się nieskończenie daleko...

Kiedy patrzymy na drzewo $(1, 1)$, to zasłania ono wszystkie inne drzewa o współrzędnych (k, k) (dla dowolnego $k \in \mathbb{N}$). Podobnie drzewo $(1, 2)$ zasłania wszystkie drzewa o współrzędnych $(k, 2k)$, a drzewo $(7, 5)$ wszystkie drzewa o współrzędnych $(7k, 5k)$.

Czy możliwe jest, aby z punktu $(0, 0)$ spojrzeć na wskroś tego lasu, tak aby nie zobaczyć absolutnie żadnego drzewa?

Zauważmy, że spoglądając na drzewo (k, l), patrzymy wzdłuż prostej $y = \frac{l}{k}x$. Oznacza to, że spoglądając na dowolne drzewo, będziemy zawsze patrzeć wzdłuż prostej, której współczynnik kierunkowy jest liczbą wymierną. Aby nie mieć na linii wzroku żadnego drzewa, wystarczy spojrzeć w stronę punktu, którego jedna współrzędna jest liczbą niewymierną.

Losujemy liczbę a z przedziału $[0, \infty)$. Jaka jest szansa, że patrząc wzdłuż prostej $y = ax$, zobaczymy drzewo?

Pytanie sprowadza się do zbadania, jaką część liczb rzeczywistych z przedziału $[0, \infty)$ stanowią liczby wymierne. Co z kolei prowadzi do stwierdzenia, że szansa na zobaczenie drzewa wynosi 0. Czytelnikom Niedowierzającym i tym, którzy dopiero rozpoczynają znajomość z przeliczalnością zbiorów, polecamy artykuł Joanny Jaszuńskiej w Δ_{13}^7.

K.Ł.

Czytelnik Purysta zapewne dostrzeże, że określenie „losujemy liczbę a z przedziału $[0, \infty)$” nie jest precyzyjne, gdyż nie podajemy rozkładu, z jakim to losowanie przebiega. Jednakże w tym miejscu nie prowadzi to do niejednoznaczności, ponieważ dla każdego rozkładu ciągłego zadanego na przedziale miara każdego jego podzbioru przeliczalnego i tak zawsze wynosi 0.