Jak definiować ciągi rekurencyjne?

*Filip MAZOWIECKI**

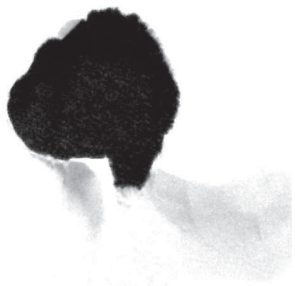
* Université de Bordeaux

Autor dziękuje Michałowi Pilipczukowi za pomoc i uwagi do tekstu.

Tytułowa rekurencja jest jednym z podstawowych pojęć w informatyce, które umożliwia definiowanie ciągów różnych obiektów, pozwalając odwoływać się w definicji danego obiektu do jego poprzedników. Pokażemy dwie klasy takich definicji i omówimy ich równoważność. Zacznijmy od prostych przykładów: ciągu arytmetycznego i ciągu geometrycznego. Rozważmy dwa ciągi $a_0, a_1, \ldots$ i $b_0, b_1, \ldots$ zadane przez

$$a_0 = 1, \quad a_{n+1} = a_n + 3 \qquad b_0 = 1, \quad b_{n+1} = 2 \cdot b_n.$$

(Takie ciągi przydają się, na przykład, do szacowania, jak długo działa program, który rekurencyjnie wywołuje siebie lub inne programy.)



W obu przypadkach, żeby poznać wartość konkretnego a_n lub b_n (korzystając wprost z definicji), trzeba się wciąż odwoływać do wartości poprzednich elementów, aż dojdziemy do pierwszego elementu (a_0 lub b_0). Oczywiście, a_n jest ciągiem arytmetycznym, a b_n ciągiem geometrycznym, więc wiemy, że procedurę obliczania wartości konkretnego elementu ciągu da się przyspieszyć i skorzystać ze wzorów: $a_n = 3n + 1$ oraz $b_n = 2^n$. Nie zawsze jest to takie proste. Trochę ciekawszym przykładem jest ciąg Fibonacciego zadany przez:

$$F_0 = 0, \quad F_1 = 1, \quad F_{n+2} = F_{n+1} + F_n.$$

Zauważmy, że aby obliczyć wartości kolejnych elementów ciągu, musimy się zawsze odwołać do dwóch poprzednich elementów i dlatego potrzebne są aż dwa początkowe elementy. Spróbujmy uogólnić to zjawisko i zdefiniować tak zwane *liniowe ciągi rekurencyjne*, którym będziemy się dalej szerzej przyglądać. Powiemy, że ciąg $c_0, c_1, \ldots$ jest liniowym ciągiem rekurencyjnym stopnia k, jeśli znamy wartości pierwszych k elementów, a pozostałe elementy są wyznaczone przez k poprzednich elementów. Dokładniej:

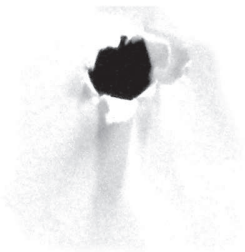
$$(1) \qquad c_0 = s_0, \quad c_1 = s_1, \ldots, c_{k-1} = s_{k-1}, \quad c_{n+k} = r_{k-1} \cdot c_{n+k-1} + \ldots + r_0 \cdot c_n,$$

dla danych liczb rzeczywistych $s_0 \ldots, s_{k-1}, r_0, \ldots r_{k-1}$. Dodatkowo zakładamy, że $r_0 \neq 0$ (inaczej ciąg byłby stopnia $k-1$). Przykładowo ciąg (b_n) jest takim ciągiem stopnia 1 i podobnie (F_n) jest takim ciągiem stopnia 2. Ciąg (a_n) nie spełnia tej definicji, ponieważ a_{n+1} odwołuje się do stałej 3 (zauważmy, że w definicji (1) stałe r_i mogą być tylko przemnożone przez jeden z poprzednich elementów ciągu c_{n+i}). Można to jednak łatwo naprawić: zauważmy, że ciąg (a_n) spełnia $a_n - a_{n-1} = 3$. Tak więc wystarczy w definicji podmienić trójkę na $(a_n - a_{n-1})$. Otrzymujemy w ten sposób równoważną rekurencję stopnia 2:

$$a_0 = 1, \quad a_1 = 4, \quad a_{n+1} = 2 \cdot a_n - a_{n-1}.$$

Pierwsze pytanie, jakim się zajmiemy w tym tekście, to pytanie, czy możemy ograniczyć się wyłącznie do ciągów stopnia 1. Okazuje się, że odpowiedź brzmi „tak", ale – niestety – nie za darmo. Jeśli pozwolimy na to, żeby użyć więcej niż jednego ciągu, to każdy ciąg stopnia k można zamienić na układ k ciągów stopnia 1. Przykładowo ciąg Fibonacciego można zdefiniować, używając jednego dodatkowego ciągu:

$$\begin{cases} F_0 = 0, & F_{n+1} = F_n + G_n \\ G_0 = 1, & G_{n+1} = F_n. \end{cases}$$



Nietrudno zauważyć, że ciąg (G_n) służy jako bufor do przetrzymywania wartości F_{n-1} i równie nietrudno uogólnić tę metodę na dowolny ciąg stopnia k. W takim razie zadajmy pytanie odwrotne: czy mając układ k ciągów stopnia 1, możemy zdefiniować każdy ciąg z tego układu za pomocą jednego ciągu (być może stopnia większego niż 1)? Rozważmy przykład, który pokaże trudność tego zadania: zdefiniujemy ciąg $d_n = n^2$, używając dwóch dodatkowych ciągów: $e_n = n$ i $f_n = 1$

$$(2) \qquad \begin{cases} d_0 = 0, & d_{n+1} = d_n + 2 \cdot e_n + f_n, \\ e_0 = 0, & e_{n+1} = e_n + f_n, \\ f_0 = 1, & f_{n+1} = f_n. \end{cases}$$

Poprawność tej definicji wynika z prostej tożsamości $(n+1)^2 = n^2 + 2n + 1$. Czy możemy zdefiniować ciąg (d_n) bez użycia dodatkowych ciągów? Warto się chwilę zastanowić nad tym pytaniem, zanim przeczytamy rozwiązanie. Odpowiedź brzmi „tak" i realizuje ją rekurencja poniżej:

$$(3) \qquad d_0 = 0, \quad d_1 = 1, \quad d_2 = 4, \quad d_{n+3} = 3 \cdot d_{n+2} - 3 \cdot d_{n+1} + d_n.$$

Szybkie rachunki pokazują poprawność tego wzoru

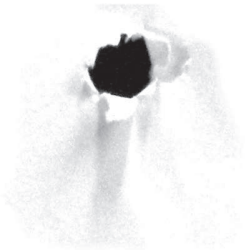
$$3 \cdot (n+2)^2 - 3 \cdot (n+1)^2 + n^2 = n^2 + 6n + 9 = (n+3)^2. \tag{4}$$

Ogólnie prawdą jest, że każdy ciąg zdefiniowany za pomocą układu k ciągów stopnia 1 możemy przekształcić w ciąg zdefiniowany za pomocą tylko jednego ciągu o stopniu k. Dowód tego faktu nie jest bardzo trudny, ale wymaga podstawowej znajomości algebry liniowej, więc nie omówimy go tutaj w pełnej ogólności.

Czytelnik Zaawansowany Algebraicznie zapisze układ k ciągów za pomocą macierzy kwadratowej wymiaru k i skorzysta z Twierdzenia Cayleya–Hamiltona.

Zamiast tego spróbujemy zademonstrować tę własność dla szczególnego przypadku. Zauważmy, że nietrudno uogólnić definicję (2) tak, żeby zdefiniować $d_n = n^k$ dla dowolnego k. Wystarczy rozpisać wzór na $(n+1)^k$ i użyć k dodatkowych ciągów. Wykażemy, że można ten ciąg zdefiniować, używając tylko jednego ciągu stopnia $k+1$. Pierwsze $k+1$ elementów zadajemy w oczywisty sposób jako:

$$d_0 = 0, \ldots, d_{k-1} = (k-1)^k, d_k = k^k.$$



Rekurencję weźmiemy trochę „z kapelusza”. Najpierw rozwiniemy wielomian w równaniu $(x-1)^{k+1} = 0$:

$$(x-1)^{k+1} = \sum_{i=0}^{k+1} \binom{k+1}{i} (-1)^i \cdot x^{k+1-i} = 0; \tag{5}$$

i przeniesiemy wszystkie wyrazy o stopniu mniejszym niż $k+1$ na prawą stronę

$$x^{k+1} = \sum_{i=1}^{k+1} \binom{k+1}{i} (-1)^{i+1} \cdot x^{k+1-i}. \tag{6}$$

Z tego wzoru magicznie otrzymujemy wzór rekurencyjny na d_n, podstawiając d_{n+i} pod jednomian x^i dla każdego $0 \leqslant i \leqslant k+1$ (w szczególności pod x^0 podstawiamy d_n):

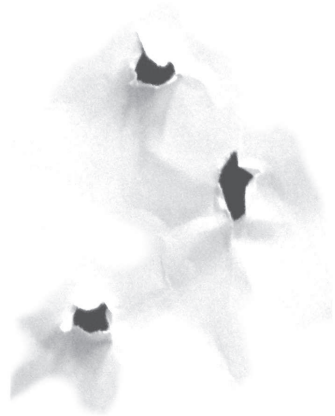
$$d_{n+k+1} = \sum_{i=1}^{k+1} \binom{k+1}{i} (-1)^{i+1} \cdot d_{n+1-i}.$$

Sprawdźmy, że dla $k = 2$ dostajemy $d_{n+3} = 3 \cdot d_{n+2} - 3 \cdot d_{n+1} + d_n$, czyli dokładnie rozwiązanie z (3). Pozostaje wykazać, że rekurencja otrzymana z (6) jest poprawna dla każdego k. Wystarczy sprawdzić, że

$$(n+k+1)^k = \sum_{i=1}^{k+1} \binom{k+1}{i} (-1)^{i+1} \cdot (n+k+1-i)^k,$$

czyli

$$\sum_{i=0}^{k+1} \binom{k+1}{i} (-1)^i \cdot (n+k+1-i)^k = 0. \tag{7}$$



Uzasadnimy to równanie, korzystając z metody *interpretacji kombinatorycznej*. Otóż dla ustalonej wartości i rozważmy $\binom{k+1}{i}(-1)^i \cdot (n+k+1-i)^k$. Przyjmijmy, że mamy dwie grupy elementów: pierwszą grupę G_1, gdzie $|G_1| = k+1$; i drugą G_2, gdzie $|G_2| = n$. Możemy zinterpretować $\binom{k+1}{i}$ jako wybranie podzbioru $A \subseteq G_1$ o rozmiarze i, natomiast $(n+k+1-i)^k$ jako dobranie ciągu z powtórzeniami $(e_1, e_2, \ldots, e_k)$ spośród pozostałych elementów, czyli $e_i \in G_2 \cup (G_1 \setminus A)$. Pozostały czynnik $(-1)^i$ określa, czy to wszystko dodamy, czy odejmiemy od całej sumy.

Obliczymy teraz to samo raz jeszcze, ale tym razem w innej kolejności. Ustalmy najpierw jakiś ciąg z powtórzeniami $(e_1, e_2, \ldots, e_k)$, gdzie $e_i \in G_1 \cup G_2$. Niech $S \subseteq G_1$ będzie zbiorem elementów, które nie występują w ciągu $(e_1, e_2, \ldots, e_k)$ i oznaczmy $m = |S|$. Pytanie brzmi, na ile różnych sposobów można dobrać do tego ciągu zbiór A tak, żeby otrzymać dokładnie to samo co poprzednio. Jedyny warunek, jaki A musi spełniać, to $A \subseteq S$. Przyjmijmy dodatkowo, że chcemy wybrać A ustalonego rozmiaru i. Możemy to zrobić na $\binom{m}{i}$ sposobów i wtedy będziemy brali ten składnik ze znakiem $(-1)^i$. To znaczy, że dla ustalonego ciągu $(e_1, e_2, \ldots, e_k)$ dodamy do sumy $\sum_{i=0}^{m} \binom{m}{i}(-1)^i$. Podstawiając w (5) $m := k+1$ oraz $x := 1$, widzimy, że to wyrażenie jest zawsze równe 0. To znaczy, że sumując po wszystkich ciągach $(e_1, e_2, \ldots, e_k)$, otrzymamy i tak 0, czyli wykazaliśmy (7).

Nasz argument, niestety, wymagał wyczarowania rekurencji otrzymanej z rozpisania (5). Żeby otrzymać ten wzór bez zgadywania, potrzeba ponownie nieco algebry liniowej. Co więcej, zauważmy, że nawet samo udowodnienie poprawności tego wzoru wymagało znajomości podstaw kombinatoryki. Niech będzie to więc przestroga, że i informatykowi dobrze jest znać trochę matematyki.