

Sortowanie naleśników

Marcin PECZARSKI*

* Wydział Matematyki, Informatyki
i Mechaniki, Uniwersytet Warszawski

Warto przy okazji wspomnieć, że 2 lutego obchodzimy *Dzień Naleśnika*, zaś w tym roku 12 lutego wypada *Thusty Czwartek*. Smacznego!

Jacob Eli Goodman (pod pseudonimem Harry Dweighter) w roku 1975 zamieścił w *The American Mathematical Monthly* następujący problem: „Nasz szef kuchni jest niechlujny, i gdy przygotowuje stos naleśników, wychodzą one wszystkie różnej wielkości. Dlatego, gdy niosę je klientowi, w drodze do stolika porządkuję je (aby najmniejszy znalazł się na górze i kolejno aż do największego na spodzie) chwytając kilka z góry i odwracając je, powtarzając tę czynność (zmieniając liczbę odwracanych [naleśników]) tyle razy, ile jest to konieczne. Jeśli jest n naleśników, jaka jest maksymalna liczba odwróceń (jako funkcja n), które będą musiały wykonać, aby je uporządkować?”

Funkcję, o której mowa powyżej, oznaczamy $f(n)$. Stos naleśników reprezentujemy jako permutację liczb od 1, która reprezentuje najmniejszy naleśnik, do n , która reprezentuje największy naleśnik, gdzie $n \geq 2$. Porządkowanie naleśników odpowiada sortowaniu rosnąco elementów permutacji przez odwracanie kolejności elementów w jej prefiksach, dlatego problem pojawia się w literaturze również pod nazwą *sorting by prefix reversal*. Sortowanie permutacji $(4, 6, 2, 5, 1, 3)$ może wyglądać na przykład tak (liczba nad strzałką oznacza długość odwracanego prefiksu):

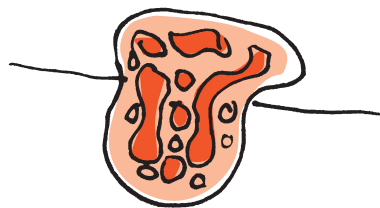
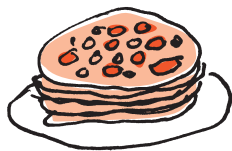
$$\begin{aligned}(4, 6, 2, 5, 1, 3) &\xrightarrow{3} (2, 6, 4, 5, 1, 3) \xrightarrow{4} (5, 4, 6, 2, 1, 3) \\ &\xrightarrow{2} (4, 5, 6, 2, 1, 3) \xrightarrow{5} (1, 2, 6, 5, 4, 3) \\ &\xrightarrow{6} (3, 4, 5, 6, 2, 1) \xrightarrow{4} (6, 5, 4, 3, 2, 1) \xrightarrow{6} (1, 2, 3, 4, 5, 6).\end{aligned}$$

Aby posortować dowolną permutację, rozważamy kolejno elementy $t = n, n-1, \dots, 3$, zachowując niezmiennik, że elementy większe od t są już na właściwych pozycjach. Jeśli t nie jest na pozycji t ani na początku permutacji, to jest na pozycji od 2 do $t-1$ i odwracamy tyle elementów, aby element t znalazł się na początku permutacji. Jeśli t jest już na początku permutacji, to odwracamy t elementów, co umieszcza t na pozycji t . W ten sposób za pomocą co najwyżej $2(n-2)$ odwróceń umieszczamy na docelowych pozycjach elementy od 3 do n . Jeśli po tym elementy 1 i 2 nie są we właściwej kolejności, to za pomocą jednego odwrócenia ustawiamy je w takiej kolejności. Powyższy algorytm pokazuje, że $f(n) \leq 2n-3$ dla $n \geq 2$.

E. Györi, G. Turán, *Stack of pancakes* (1978).

W.H. Gates, Ch.H. Papadimitriou, *Bounds for sorting by prefix reversal* (1979).

Przedstawimy teraz lepszy algorytm. Wymyślili go Ervin Györi i György Turán oraz niezależnie od nich William Henry Gates III (bardziej znany jako Bill Gates) i Christos Harilaos Papadimitriou. Równolegle z opisem samego algorytmu będziemy analizować jego złożoność, a w tym celu potrzebujemy wprowadzić pewne definicje. Dwa elementy na sąsiednich pozycjach permutacji tworzą *dobre sąsiedztwo*, jeśli ich wartości różnią się o 1. Przyjmujemy ponadto, że dobre sąsiedztwo tworzą też elementy 1 i n . Maksymalny podciąg elementów (co najmniej dwóch) tworzących dobre sąsiedztwa nazywamy *blokiem*. Element nienależący do żadnego bloku nazywamy *wolnym*. Przykładowo w permutacji $(3, 5, 4, 7, 1, 2, 6)$ mamy dobre sąsiedztwa $(5, 4)$, $(7, 1)$ i $(1, 2)$, bloki $(5, 4)$ i $(7, 1, 2)$ oraz elementy wolne 3 i 6. Dodawanie liczby całkowitej do elementu permutacji lub odejmowanie liczby całkowitej od elementu permutacji wykonujemy cyklicznie: $n+1=1$, $n+2=2$, $1-1=n$, $1-2=n-1$ itd. Przyjmujemy, że po wykonaniu ℓ odwróceń *potencjał* uzyskanej permutacji jest równy $\Phi_\ell = \ell + \alpha w + \beta b$, gdzie w jest liczbą elementów wolnych w tej permutacji, b jest liczbą bloków w tej permutacji, a α i β są pewnymi stałymi, których wartości wyznaczmy później. Przez $\Delta\Phi$ oznaczamy zmianę potencjału w sekwencji odwróceń. Wielokropkiem zastępujemy podciąg elementów permutacji nietworzący dobrych sąsiedztw z elementami go poprzedzającym i następującym po nim. Wielokropek może też oznaczać podciąg pusty. Podkreśleniem zastępujemy podciąg bloku, być może pusty. Niech $d \in \{-1, 1\}$. Zależnie od postaci permutacji rozpatrujemy następujące przypadki, za każdym razem przedstawiając również odwrócenia, jakich należy wówczas dokonać.



1. Permutacja zaczyna się elementem wolnym t .

(a) Istnieje element wolny $t + d$:

$$(t, \dots, t + d, \dots) \rightarrow (\dots, t, t + d, \dots).$$

Jedno odwrócenie zamienia dwa elementy wolne w jeden blok, czyli $\Delta\Phi = 1 - 2\alpha + \beta$.

(b) Istnieje blok zaczynający się elementem $t + d$:

$$(t, \dots, t + d, _, \dots) \rightarrow (\dots, t, t + d, _, \dots).$$

Jedno odwrócenie zmniejsza o jeden liczbę elementów wolnych, czyli $\Delta\Phi = 1 - \alpha$.

(c) Istnieją bloki kończące się elementami $t + d$ i $t - d$:

$$\begin{aligned} (t, \dots, _, t + d, \dots, _, t - d, \dots) \\ \rightarrow (t + d, _, \dots, t, \dots, _, t - d, \dots) \\ \rightarrow (\dots, _, t + d, t, \dots, _, t - d, \dots) \\ \rightarrow (t - d, _, \dots, t, t + d, _, \dots) \\ \rightarrow (\dots, _, t - d, t, t + d, _, \dots). \end{aligned}$$

Cztery odwrócenia zmniejszają o jeden liczbę elementów wolnych i bloków, czyli $\Delta\Phi = 4 - \alpha - \beta$.

2. Permutacja zaczyna się blokiem o długości k , gdzie $1 < k < n - 1$. Pierwszym elementem bloku jest t , a ostatnim $t + (k - 1)d$.

(a) Istnieje element wolny $t - d$:

$$\begin{aligned} (t, _, t + (k - 1)d, \dots, t - d, \dots) \\ \rightarrow (\dots, t + (k - 1)d, _, t, t - d, \dots). \end{aligned}$$

Jedno odwrócenie zmniejsza o jeden liczbę elementów wolnych, czyli $\Delta\Phi = 1 - \alpha$.

(b) Istnieje blok zaczynający się elementem $t - d$:

$$\begin{aligned} (t, _, t + (k - 1)d, \dots, t - d, _, \dots) \\ \rightarrow (\dots, t + (k - 1)d, _, t, t - d, _, \dots). \end{aligned}$$

Jedno odwrócenie zmniejsza o jeden liczbę bloków, czyli $\Delta\Phi = 1 - \beta$.

(c) Istnieją blok kończący się elementem $t - d$ i element wolny $t + kd$. Zależnie od ich wzajemnego położenia stosujemy odwrócenia

$$\begin{aligned} (t, _, t + (k - 1)d, \dots, _, t - d, \dots, t + kd, \dots) \\ \rightarrow (t + kd, \dots, t - d, _, \dots, t + (k - 1)d, _, t, \dots) \\ \rightarrow (\dots, _, t - d, \dots, t + kd, t + (k - 1)d, _, t, \dots) \\ \rightarrow (t, _, t + (k - 1)d, t + kd, \dots, t - d, _, \dots) \\ \rightarrow (\dots, t + kd, t + (k - 1)d, _, t, t - d, _, \dots) \end{aligned}$$

lub

$$\begin{aligned} (t, _, t + (k - 1)d, \dots, t + kd, \dots, _, t - d, \dots) \\ \rightarrow (t + kd, \dots, t + (k - 1)d, _, _, t, \dots, _, t - d, \dots) \\ \rightarrow (\dots, t + kd, t + (k - 1)d, _, t, \dots, _, t - d, \dots) \\ \rightarrow (t - d, _, \dots, t, _, t + (k - 1)d, t + kd, \dots) \\ \rightarrow (\dots, _, t - d, t, _, t + (k - 1)d, t + kd, \dots). \end{aligned}$$

Cztery odwrócenia zmniejszają o jeden liczbę elementów wolnych i bloków, czyli $\Delta\Phi = 4 - \alpha - \beta$.

(d) Istnieje blok, którego pierwszym elementem jest $t + kd$:

$$\begin{aligned} (t, _, t + (k - 1)d, \dots, t + kd, _, \dots) \\ \rightarrow (t + (k - 1)d, _, t, \dots, t + kd, _, \dots) \\ \rightarrow (\dots, t, _, t + (k - 1)d, t + kd, _, \dots). \end{aligned}$$

Istnieje blok, którego ostatnim elementem jest $t + kd$:

$$\begin{aligned} (t, _, t + (k - 1)d, \dots, _, t + kd, \dots) \\ \rightarrow (t + kd, _, \dots, t + (k - 1)d, _, t, \dots) \\ \rightarrow (\dots, _, t + kd, t + (k - 1)d, _, t, \dots). \end{aligned}$$

Dwa odwrócenia zmniejszają o jeden liczbę bloków, czyli $\Delta\Phi = 2 - \beta$.

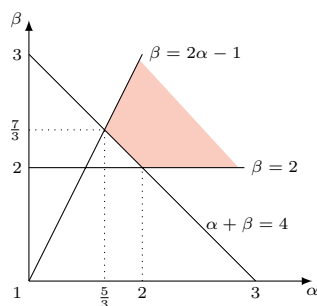
3. Jeśli permutacja nie pasuje do żadnego z powyższych przypadków, to jest blokiem. Jeśli nie jest to permutacja $(1, _, n)$, to zależnie od jej postaci mamy następujące przypadki:

- (a) $(n, _, 1) \xrightarrow{n} (1, _, n)$,
- (b) $(n - 1, _, 1, n) \xrightarrow{n-1} (1, _, n)$ dla $n \geq 3$,
- (c) $(n, 1, _, n - 1) \xrightarrow{n} (n - 1, _, 1, n) \xrightarrow{n-1} (1, _, n)$ dla $n \geq 3$,
- (d) $(2, _, n, 1) \xrightarrow{n-1} (n, _, 1) \xrightarrow{n} (1, _, n)$ dla $n \geq 3$,
- (e) $(1, n, _, 2) \xrightarrow{n} (2, _, n, 1)$ dla $n \geq 3$ i dalej jak w punkcie 3.d,
- (f) $(t + 1, _, n, 1, _, t) \xrightarrow{n-t} (n, _, t + 1, 1, _, t) \xrightarrow{n} (t, _, 1, t + 1, _, n) \xrightarrow{t} (1, _, n)$ dla $2 \leq t \leq n - 2$,
- (g) $(t, _, 1, n, _, t + 1) \xrightarrow{n} (t + 1, _, n, 1, _, t)$ dla $2 \leq t \leq n - 2$ i dalej jak w punkcie 3.f.

Będziemy wymagać, aby po odwróceniach z punktów 1 i 2 potencjał permutacji nie zwiększał się, czyli $\Delta\Phi \leq 0$, więc muszą być spełnione nierówności:

$$\begin{aligned} \alpha &\geq 1, \\ \alpha + \beta &\geq 4, \\ 2 &\leq \beta \leq 2\alpha - 1. \end{aligned} \tag{6}$$

Potencjał Φ_0 początkowej permutacji wynosi $\alpha w + \beta b$. Ponieważ każdy blok zawiera co najmniej dwa elementy, więc $w \leq n - 2b$, czyli $\Phi_0 \leq \alpha n + (\beta - 2\alpha)b$. Korzystając z ostatniej nierówności w (6), dostajemy $\Phi_0 \leq \alpha n - b \leq \alpha n$. Równość $\Phi_0 = \alpha n$ zachodzi, gdy permutacja ma n elementów wolnych. Każda sekwencja odwróceń z punktów 1 i 2 zmniejsza liczbę elementów wolnych lub bloków, więc po skończonej liczbie m odwróceń dochodzimy do punktu 3. Wtedy $\Phi_m = m + \beta$. W punkcie 3, aby dokończyć sortowanie, wykonujemy co najwyżej cztery odwrócenia, zatem $f(n) \leq m + 4 = \Phi_m - \beta + 4$. Ponieważ założyliśmy, że potencjał



B. Chitturi et al., An $(18/11)n$ upper bound for sorting by prefix reversals (2009).

permutacji nie rośnie, więc $\Phi_m \leq \Phi_0$ i stąd $f(n) \leq \alpha n - \beta + 4$. Pozostaje znaleźć rozwiązanie układu nierówności (6) minimalizujące wartość α . Jest to przykład zagadnienia programowania liniowego. Dla dwóch niewiadomych możemy je rozwiązać, stosując interpretację geometryczną, patrz rysunek na marginesie. Otrzymujemy $\alpha = \frac{5}{3}$ i $\beta = \frac{7}{3}$. Zatem $f(n) \leq 5(n+1)/3$.

Nieco lepsze górne oszacowanie $f(n) \leq (18/11)n + O(1)$ udowodnił Bhadrachalam Chitturi ze współpracownikami. Ich dowód wymaga jednak rozpatrzenia 2220 przypadków. Znamy dokładnie wartości $f(n)$ dla $n \leq 19$, patrz tabela niżej (jest to ciąg o numerze A058986 w *The On-Line Encyclopedia of Integer Sequences*).

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
$f(n)$	1	3	4	5	7	8	9	10	11	13	14	15	16	17	18	19	20	22

W mojej pracy *Note on pancake sorting*, opublikowanej w czasopiśmie *Information Processing Letters*, zajmowałem się oszacowaniami dolnymi. Skonstruowałem dla większych n permutacje świadczące, że $f(n) \geq \lfloor (15n+9)/14 \rfloor$.

Na koniec zauważmy, że w całym artykule milcząco zakładaliśmy, że strony naleśnika są nierozróżnialne. Każdy, kto smażył naleśniki, wie jednak, że zwykle jedna strona wychodzi bardziej przysmażona. Rozważa się więc wersję problemu sortowania naleśników tak, aby wszystkie były zwrócone przysmażoną stroną w dół, ale o tym innym razem.



Genetyka w czwartym wymiarze

Dziś będzie o nukleomie. Wiem, sprawa jest niewesoła. Kiedy zacznie się mówić o genetyce, zasób potrzebnych słów może zniechęcić nawet najbardziej wytrwałych słuchaczy. Nazwy cząsteczek (np. DNA, RNA) i konkretnych struktur komórkowych (np. nukleosom, chromosom, jądro komórkowe) płaczą się ze słowami określającymi pojęcia abstrakcyjne mówiące o funkcji (gen, kod genetyczny, transkrypcja, translacja). Czas pędzi, a lista pojęć się wydłuża. Jak za tym nadążyć?

Kiedy zaczęłam uczyć się o DNA, na czarnej tablicy rysowano nam geny. Długa cienka kreska, na niej zaznaczone pudełeczko podpisane jakimś skrótem (np. *DIN7*) oraz (czasami) dodane jeszcze dwa dużo mniejsze, z przodu i z tyłu. Gdzieś na długiej nici DNA organizmu X znajduje się kawałek, nazwany genem *DIN7*, który ma element rozpoznawany przez białka odczytujące informację genetyczną (promotor) i miejsce, gdzie odczyt się kończy (terminator). Schludne to i proste. Ale...

Przychodzi mi na myśl inny obraz: jądro komórkowe jako wielka płatanina cienkich nitek, chaos, bałagan, DNA jak rozwinięte motki włóczek upchane kolanem w szafce z materiałami do szydełkowania. I informacja, że gdyby te wszystkie nitki DNA wyciągnąć z ludzkiej komórki, rozplatać i położyć jedna za drugą, to mierzyłoby to wszystko 2 metry! DWA METRY!?

Pierwszy odczyt sekwencji ludzkiego genomu sprawę dodatkowo zagmatwał. Wbrew oczekiwaniom – genów ludzkich jest jedynie około 30 tysięcy. Zaledwie 2% ludzkiego DNA koduje białka, a olbrzymia jego część to (wtedy tak nazywany) „śmieciowy DNA”.

Aż nadszedł czas, kiedy ludzie zaczęli grzebać w tych „śmieciach”. I trzeba było stworzyć dużo nowych pojęć. A gdzie nukleom?

W 2017 roku konsorcjum kilkunastu instytutów badawczych z USA oraz kilku organizacji z reszty świata rozpoczęło projekt o nazwie „Nucleome 4D”. Przedsięwzięcie potężne, bo wymagało nie tylko zebrania i analizy olbrzymiej