Zajęcia C-6: “Sortowanie szybkie”

Cel zajęć i efekty uczenia

Główne cele zajęć / materiał do opanowania:

- Algorytm sortowania szybkiego (QuickSort)
- Technika dwóch wskaźników (“gąsienicy”) na przykładzie problemu 2-SUM

Dodatkowe cele:

- Łagodne zapoznanie uczniów z algorytmami randomizowanymi (tzw. Las Vegas)

Zadania do rozwiązania na sprawdzarce

Randka w ciemno
Dany jest zbiór osób zawierający, dla każdej, imię oraz pewną liczbę. Znaleźć dwie osoby, których liczby sumują się do ustalonej sumy S.

Plan zajęć

Szacunkowy czas trwania: 2 godziny lekcyjne.

1. Idea algorytmu QuickSort
    - *Stosunkowo łatwym sposobem wprowadzenia QuickSorta jest zaproponowanie idei analogicznej do scalania: zamiast najpierw sortować rekurencyjnie, a potem scalić, trzeba podzielić tak, żeby scalanie nie było już potrzebne - po rekursji tablica była już posortowana. Na ogół to wystarczy uczniom do wymyślenia idei “małe elementy na początek, duże na koniec”.*
2. Podział tablicy (*pivoting*)
    - *Po wybraniu elementu dzielącego (pivota) podział tablicy można zrobić na dwa sposoby: metodą Hoare’a (oryginalny pomysł autora QuickSorta), albo metodą Lomuto (nowszą).*
    - *Metoda Lomuto jest prostsza do wymyślenia i łatwiejsza do wytłumaczenia uczniom, ale ma istotną wadę: źle działa na tablicach, w których jest dużo identycznych elementów (na przykład na tablicy, w której wszystkie elementy są równe), dzieląc je nierówno i powodując kwadratowy czas działania całego algorytmu. Należy to koniecznie uświadomić uczniom, jako że test “wszystkie liczby jednakowe” w zasadzie musi być jednym z testów na sprawdzarce. Można zmodyfikować metodę Lomuto tak, aby uwolnić ją od tej wady - na przykład dzieląc tablicę na trzy części (elementy mniejsze od pivota, równe i większe), ale wymaga to dodatkowego wysiłku.*
    - *Metoda Hoare’a działa zawsze i ma bardzo krótki kod, ale jej zachowanie na małych tablicach i w przypadkach brzegowych jest bardzo nieintuicyjne - dwa wskaźniki, których używa (idące z lewego i prawego końca tablicy), czasem*

wskazują na to samo miejsce, czasem wręcz się mijają, a czasem nawet wychodzą poza tablicę. Z mojego doświadczenia próba jakiegokolwiek formalnego udowodnienia uczniom na lekcji, że ta metoda działa, jest skazana na porażkę. Co gorsza, algorytm ma tendencję do błędnego działania przy najmniejszej, nawet zupełnie niewinnej, zmianie w kodzie.
    - *Nie ma w tej sytuacji idealnego rozwiązania. Proponowany sposób na przeprowadzenie lekcji: napisać na tablicy metodę Hoare'a, rozważyć jej ogólne działanie (na dużych tablicach jest dość oczywiste) i zaznaczyć to, co jest napisane w powyższym punkcie - że działanie na przypadkach brzegowych wymaga bardzo dokładnego przepisania algorytmu.*
3. Złożoność algorytmu
    - *Rozważamy dwa skrajne przypadki: algorytm zawsze losuje medianę zbioru (czyli element środkowy pod względem wielkości - wychodzi złożoność jak w sortowaniu przez scalanie, O(n log n)), oraz algorytm zawsze losuje najmniejszy element (złożoność kwadratowa). Dla ilustracji można jeszcze (choć wymaga to więcej czasu i dobrej intuicji odnośnie logarytmów) wytłumaczyć przypadek "zawsze losujemy element, który jest w ⅓ pod względem wielkości"), i przez analogię argumentować, że nawet wybór elementu w 1/10 będzie dobry.*
    - *Wynika z tego, że aby algorytm działał wolno (kwadratowo), wybór musi być wyjątkowo pechowy, czyli dane wyjątkowo złośliwie dobrane. Warto w tym momencie poświęcić chwilę na układanie z uczniami złośliwych przykładów na różne wybory elementu dzielącego (np. na wybieranie go zawsze z początku tablicy/zawsze z końca tablicy/zawsze ze środka tablicy).*
    - *A zatem losowanie elementu dzielącego praktycznie gwarantuje dobrą złożoność - żadne spreparowane dane nie są w stanie "wyłożyć" algorytmu z losowaniem. Tutaj warto wytłumaczyć uczniom, że algorytm, który używa losowania (randomizowany) uznajemy za dobry, jeśli najczęściej działa dobrze. "Najczęściej"* ***nie oznacza*** *"na prawie każdych danych", tylko "przy prawie każdym losowaniu": poprawność zależy "od nas i naszego losowania", a nie od "tego, który układał dane".*
4. Zadanie "Randka w ciemno" i metoda dwóch wskaźników
    - *Pierwszy sposób rozwiązania zadania: sortujemy tablicę. Dla każdego elementu bierzemy jego wartość x, i sprawdzamy, czy w tablicy istnieje liczba S-x, za pomocą wyszukiwania binarnego.*
    - *Alternatywna, polecana metoda: niech A będzie rozważaną tablicą elementów, ustawiamy zmienną i na 0 (początek tablicy), j na ostatni element tablicy. Jeśli suma elementów stojących na miejscach i oraz j jest mniejsza niż S, to znaczy że element A[i] jest zbyt mały (nie może dać sumy S z żadnym innym elementem). Zatem możemy zapomnieć o elemencie A[i] do końca algorytmu, zwiększając wskaźnik i o 1. Analogicznie, jeśli A[i]+A[j]>S, to element A[j] nie będzie już potrzebny, wykonujmy więc j = j-1. Powtarzamy operację do momentu znalezienia sumy S, albo spotkania się wskaźników i oraz j.*

- *Technika ta zwana jest czasem “gąsienicą”, ma kilka wariantów i wiele zastosowań “olimpijskich”. Można rozważyć poświęcenie jej osobnych zajęć.*