











Lista 4, zadanie 8 - Tomasz Woszczyński

Treść: Na każdym polu szachownicy o wymiarach $4 \times n$ znajduje się jedna liczba naturalna. Ułóż algorytm, który umieszcza na szachownicy kamyki w taki sposób, że:

- na każdym polu znajduje się co najwyżej jeden kamień,
- jeśli na polu P znajduje się kamyk, to na polach mających wspólny bok z P nie ma kamyków,
- suma liczb z pól, na których leżą kamiki jest maksymalna.

Rozwiązanie: Rozpatrzmy najpierw jak mogą być ustawione kamiki w poszczególnych kolumnach (numery na dole oznaczają kolejne możliwości, tabelka nie przedstawia też przykładowej z treści zadania):

brak kamyków

							
							
							
							
1	2	3	4	5	6	7	8

Zastosowanie każdej z możliwości zależy od wyboru położenia kamieni w poprzedniej kolumnie, oznacza to, że nie możemy mieć w dwóch kolejnych kolumnach niektórych ułożeń kamyków, gdyż nie zgadzało by się to z drugim warunkiem z treści. Oznaczmy sobie przez $c[i][j]$ sumę pól w i -tej kolumnie, przy wyborze możliwości j . Rozpatrzmy teraz jakie możliwe ułożenia będziemy mieli w kolumnie $i-1$ oraz i :

możliwość w kolumnie i	1	2	3	4	5	6	7	8
możliwości w kolumnie $i-1$	1,2,3,4,5,6,7,8	1,3,4,5,8	1,2,4,5,6,7	1,2,3,5,6,8	1,2,3,4,7	1,3,4	1,3,5,7	1,2,4,7

Zdefiniujemy wtedy funkcję:

$$T[i][j] = \max_{k \in \text{mozliwosci}(j)} (T[i-1][k] + c[i][j])$$

gdzie $\text{mozliwosci}(x)$ zwraca możliwe konfiguracje dla kolumny x . Algorytm ten wykona się w czasie $O(n)$, gdyż rekurencyjnie przechodzimy przez całą tablicę od ostatniej do pierwszej kolumny i wykona się poprawnie, gdyż za każdym razem dodajemy największą znaną wartość z dwóch sąsiednich kolumn. Algorytm działanie zakończy, gdy spełniony będzie warunek $i == 0$.