Zadania domowe. Zestaw 3.2

Maciej Poleski

13 maja 2013

1

$$\begin{bmatrix} A & B & C \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_{n-1} \\ x_{n-2} \\ 1 \end{bmatrix} = \begin{bmatrix} Ax_{n-1} + Bx_{n-2} + C \\ x_{n-1} \\ 1 \end{bmatrix} = \begin{bmatrix} x_n \\ x_{n-1} \\ 1 \end{bmatrix}$$

Korzystając z powyższych równości oraz z łączności mnożenia macierzy możemy ustalić poszukiwany wyraz ciągu podnosząc skrajnie lewą macierz do odpowiedniej (k) potęgi (algorytmem szybkiego potęgowania, nie zapominając o operacji modulo) i mnożąc tak uzyskaną macierz przez wektor (pionowy) $[x_1\ x_0\ 1]$. Pozostaje wybrać z uzyskanego wektora wynikowego środkową komórkę.

Rozmiar macierzy jest stały więc złożoność obliczeniowa jest złożonością algorytmu szybkiego potęgowania ($O(\log k)$) (koszt mnożenia macierzy jest stały).

2

Niech A_k będzie macierzą taką że a_{ij}^k = liczba ścieżek z wierzchołka i do j o długości k. Zauważmy że jeżeli A jest macierzą sąsiedztwa to $A_{k+1} = A_k \times A$. Istotnie $a_{ij}^{k+1} = \sum_{l=1}^{n} a_{il}^k a_{lj}$, gdzie a_{lj} = [istnieje krawędź między l i j] (l jest wierzchołkiem do którego docieramy w k krokach i jeżeli tylko istnieje połączenie z l do j, to składnik a_{il}^k jest liczony do sumy).

Ponownie wykorzystujemy szybkie potęgowanie. Koszt mnożenia macierzy to $O(|V|^3)$ (z definicji). Koszt całego algorytmu to $O(|V|^3 \log k)$.