

1 УМНОЖЕНИЕ МАТРИЦ ПО ВИНОГРАДУ

Наивный алгоритм перемножения матриц требует $O(n^3)$ операций.

Алгоритм Штрассена улучшает асимптотику до $O(n^{2.807355})$. Алгоритм

Коммерсмита-Винограда работает за $O(n^{2.375477})$, но на практике не используется из-за очень большой скрытой константы. На лекциях рассказывался небольшой лайфхак, который все равно работает за куб, но немного по другому.

В обычном алгоритме перемножения матриц элемент считается как

$$c_{ij} = \begin{pmatrix} a_{i1} & a_{i2} & \dots & a_{iN} \end{pmatrix} \begin{pmatrix} b_{1j} \\ b_{2j} \\ \dots \\ b_{Nj} \end{pmatrix} = \sum_{k=1}^N a_{ik} b_{kj}, \quad (1.1)$$

То есть всего мы сделаем N^3 умножений и $N^3 - N^2$ сложений.

Однако можно заметить, что

$$c_{ij} = \sum_{k=1}^{N/2} (a_{i,2k-1} + b_{2k,j}) (a_{i,2k} + b_{2k-1,j}) - \sum_{k=1}^{N/2} a_{i,2k-1} a_{i,2k} - \sum_{k=1}^{N/2} b_{2k-1,j} b_{2k,j}. \quad (1.2)$$

Второе и третье слагаемое можно предсчитать и потом много раз использовать. То есть всего умножений и сложений будет

$$\frac{1}{2} N^3 + N^2 \quad (1.3)$$

и

$$\frac{3}{2} N^3 + 2N^2 - 2N \quad (1.4)$$

соответственно (конкретные чиселки зависят от того, что мы считаем сложением, лучше в коде везде раскидать комментарии, сколько где производится операций). Это немного веселее, чем в обычном алгоритме, потому что сложение дешевле, чем умножение. Приведем сам код

```

// предпосчет второго слагаемого
for i = 1:N
    row(i) = A(i, 1) * A(i, 2)
    for k = 2:N/2
        row(i) = row(i) + A(i, 2*k-1) * A(i, 2*k)
    end
end

// предпосчет третьего слагаемого
for j = 1:N
    col(j) = B(1, j) * B(2, j)
    for k = 2:N/2
        col(j) = col(k) + B(2*k-1, j) * B(2*k, j)
    end
end

// само перемножение
for i = 1:N
    for j = 1:N
        C(i, j) = -row(i) - col(j)
        for k = 1:N/2
            C(i, j) = C(i, j) + (A(i, 2 * k - 1) + B(2 * k, j)) *
(A(i, 2 * k) + B(2 * k - 1, j))
        end
    end
end

// если размерность матрицы нечетная, то надо не забыть
последние столбец/строку
if N % 2 == 1 then
    for i = 1:N
        for j = 1:N
            C(i, j) = C(i, j) + A(i, N) * B(N, j)
        end
    end
end
end

```