

Рекомендации к чтению

- Конспект семинара
- Кормен, глава 30
- Тыртышников, лекция 34
- [Статья про wildcard matching](#)

Ключевые понятия: быстрое преобразование Фурье (БПФ, FFT).

Обязательные задачи

Задача 1 (3) Перемножьте многочлены $1 + 2x + 3x^2$ и $3x - 4 + 6x^3$ с помощью ДПФ (используйте $\omega_8 = e^{\frac{2\pi i}{8}}$). Используйте алгоритм БПФ (для вычислений двух ДПФ и одного обратного ДПФ).

Задача 2 (3) Задача 30.1.7 на стр. 948 в Кормене.

Задача 3 (2 + 2 + 1) Прочитайте статью из 4 пункта списка рекомендаций для чтения

1. Покажите, как, используя БПФ, построить $O(n \log n)$ -алгоритм поиска вхождения образца в текст длины n .
2. Покажите, как, используя БПФ, построить $O(n \log n)$ -алгоритм поиска вхождения образца в текст длины n , если в образце разрешены “джокеры”, которые в ТРЯПе обозначались знаком вопроса “?”.
3. Покажите, как понизить сложность предыдущих двух алгоритмов до $O(n \log m)$ (здесь m – длина образца).

Задача 4 (3 + 3) Рассмотрим систему линейных уравнений, записанную в матричной форме:

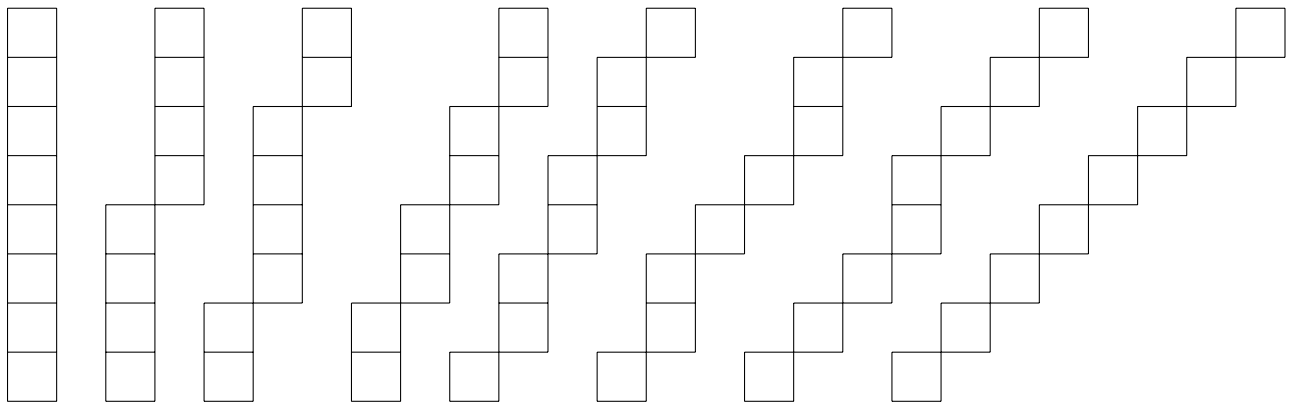
$$\begin{pmatrix} c_0 & c_{n-1} & c_{n-2} & \dots & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & \dots & c_3 & c_2 \\ c_2 & c_1 & c_0 & \dots & \vdots & \vdots \\ \dots & \dots & \dots & \ddots & \vdots & \vdots \\ c_{n-2} & c_{n-3} & \dots & \dots & c_0 & c_{n-1} \\ c_{n-1} & c_{n-2} & \dots & \dots & c_1 & c_0 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-1} \end{pmatrix}$$

Матрица специального вида, использованная здесь, называется *циркулянт*ом.

1. Докажите, что если покомпонентно умножить вектор $DFT(c_0, c_1, \dots, c_{n-1})$ на вектор $DFT(x_0, x_1, \dots, x_{n-1})$, то получится вектор $DFT(b_0, b_1, \dots, b_{n-1})$.
2. Покажите, как эффективно решать линейные системы типа той, что выписана выше, используя БПФ.

Дополнительные задачи (можно сдавать в течение семестра)

Задача 5 (5) В обработке изображений возникает задача распознавания прямых, решаемая при помощи так называемого *преобразования Хафа* (дискретного аналога *преобразования Радона*). Если отбросить все детали и оставить самую суть, то требуется решить следующую задачу. Есть изображение ширины w и высоты h , состоящее из пикселей (давайте считать значение каждого пикселя числом от 0 до 255). Для простоты нам удобно будет считать, что h – степень двойки. Рассматриваются все возможные дискретные прямые, соединяющие нижний край и верхний край изображения, причём угол отклонения прямых от вертикали направо может принимать значения от 0° до 45° . Всего бывает h типов таких прямых: тип прямой однозначно задаётся разностью абсцисс самого пикселя прямой и самого верхнего пикселя прямой, а эта разность может принимать значения от 0 до $h - 1$. На рисунке ниже приведены все типы прямых высоты 8.



Для каждой рассматриваемой прямой (понятно, что их порядка wh) на изображении требуется найти сумму значений пикселей на этой прямой. Это и есть преобразование Хафа.

Придумайте алгоритм (в духе БПФ), который вычислит сумму на всех прямых за $O(wh \log h)$. Обратите внимание, что не было дано чёткого определения дискретной прямой (не было сказано, как точно определяется её пиксельная структура), и вы вольны определить дискретную прямую любым естественным образом, подходящим под специфику вашего алгоритма.