Построение и анализ алгоритмов

Лекция 4.3

Динамическое программирование

АНАЛОГИИ

АНАЛОГИИ

Решение методом динамического программирования

Задачи:

- Перемножение цепочки матриц
- Оптимальные БДП
- *Задача X и т.п.*

Задачи ***подсчета*** и задачи ***оптимизации***.

Например:

«Число различных расстановок скобок» и «Оптимальная расстановка»

Оценка количества узлов дерева

Из лекции про перемножение матриц

Оценить количество узлов дерева в общем случае можно подсчетом всех возможных ***вариантов расстановок скобок в произведении*** матриц.

Пусть p_n – число вариантов расстановок скобок в произведении n сомножителей (включая самые внешние скобки).

Например, для трех сомножителей abc имеем два варианта $(a(bc))$ и $((ab)c)$, а следовательно, $p_3 = 2$.

В общем случае, считая, что «последнее» по порядку умножение может оказаться на любом из $n-1$ мест, запишем следующее рекуррентное соотношение:

$$p_n = p_1\, p_{n-1} + p_2\, p_{n-2} + \ldots + p_{n-2}\, p_2 + p_{n-1}\, p_1.$$

Из лекции про перемножение матриц

Начальное условие $p_1 = 1$. Далее

$$p_2 = p_1 p_1 = 1,$$

$$p_3 = p_1 p_2 + p_2 p_1 = 2,$$

$$p_4 = p_1 p_3 + p_2 p_2 + p_3 p_1 = 5.$$

Оказывается [7, с. 393], что решением этого рекуррентного уравнения являются так называемые ***числа Каталана***

$p_n = C_{n-1}$, где $C_n = (2k \mid k) / (k+1)$,

а запись $(n \mid m)$ обозначает биномиальный коэффициент

$$(n \mid m) = n!/(m!\,(n - m)!).$$

При больших значениях n справедливо

$$C_n \approx 4^n \big/ n\sqrt{\pi n}$$

т. е. число узлов в дереве перебора есть экспоненциальная функция от n.

Несколько первых чисел Каталана

Из лекции про перемножение матриц

n	0	1	2	3	4	5	6	7	8	9	10
C_n	1	1	2	5	14	42	132	429	1430	4862	16 796

Ср. C_{n-1} и $(n^3 - n)/3$

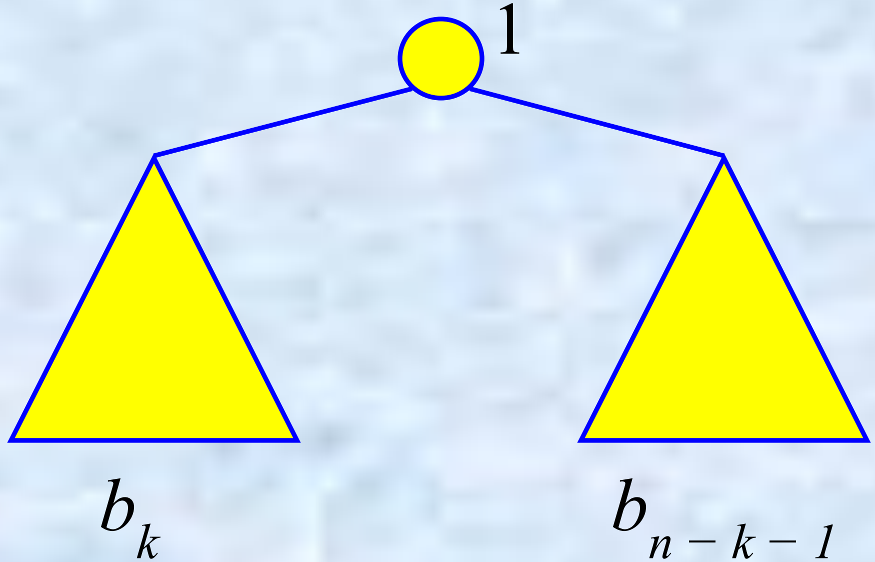
Например, при $n = 10$

n	6	7	8	9	10
C_{n-1}	42	132	429	1430	4862
$(n^3 - n)/3$	70	112	168	240	330

Число b_n структурно различных бинарных деревьев с n узлами

$$b_n = \sum_{k=0}^{n-1} b_k b_{n-k-1} = b_0 b_{n-1} + b_1 b_{n-2} + \ldots + b_{n-2} b_1 + b_{n-1} b_0, \quad b_0 = 1, b_1 = 1.$$

Из лекции про БДП



$k \in 0..(n-1)$

Это рекуррентное уравнение с точностью до обозначений совпадает с рекуррентным уравнением, получающимся при подсчете числа расстановки скобок в произведении n сомножителей (см. лекцию 16, слайд 16).

Из лекции про БДП

$$b_2 = b_0\, b_1 + b_1\, b_0 = 2,$$
$$b_3 = b_0\, b_2 + b_1\, b_1 + b_2\, b_0 = 5,$$
$$b_4 = b_0\, b_3 + b_1\, b_2 + b_2\, b_1 + b_3\, b_0 =$$
$$= 5 + 2 + 2 + 5 = 14$$

Решением рекуррентного уравнения являются так называемые *числа Каталана* C_n, т. е. $b_n = C_n$.

Ранее (Лекция 13, слайды хх, хх) были приведены: общая формула для чисел Каталана и асимптотическое соотношение

$$\boxed{C_n \approx 4^n \big/ n\sqrt{\pi n}}$$

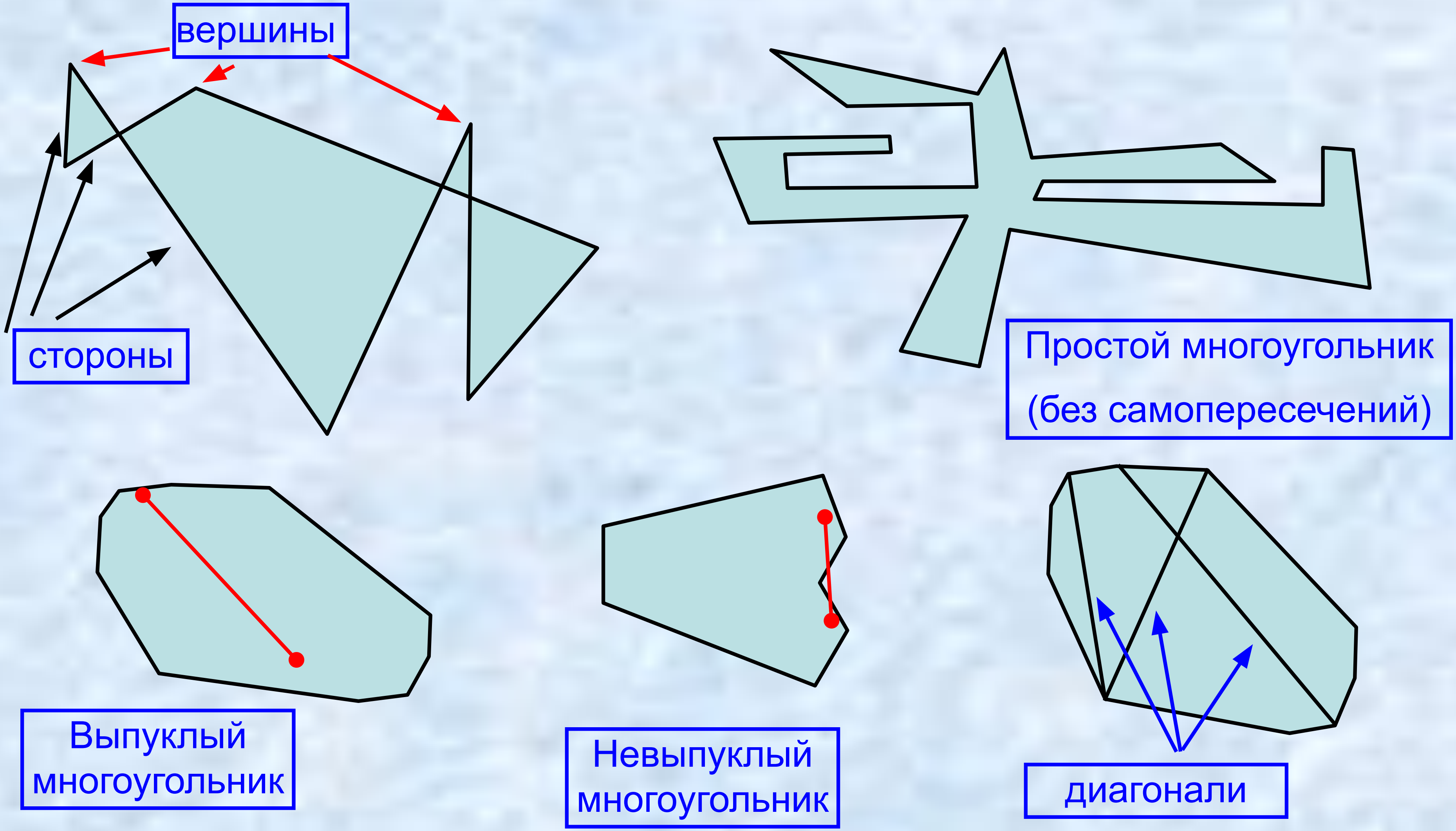
n	0	1	2	3	4	5	6	7	8	9	10
C_n	1	1	2	5	14	42	132	429	1430	4862	16 796

Решение методом динамического программирования

1. Структура оптимального решения
2. Рекуррентное соотношение
3. Вычисление оптимальной стоимости (по рекуррентному соотношению)
4. Построение оптимального решения

Проиллюстрировать на предыдущих примерах

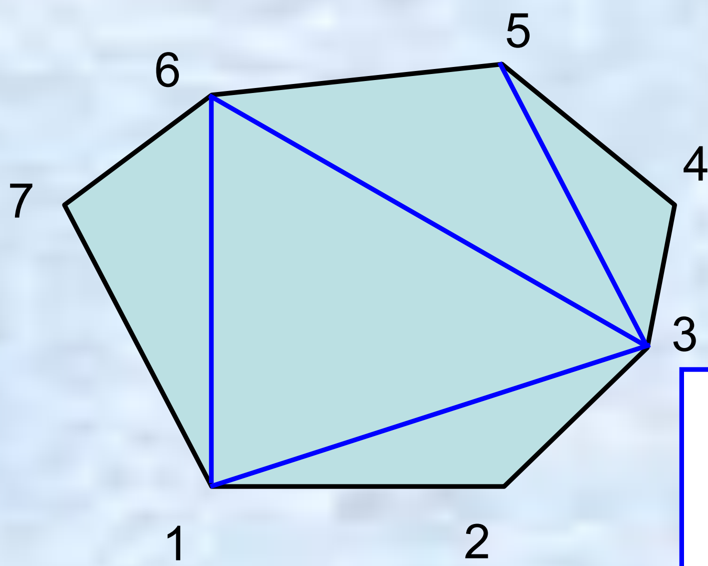
Задача: оптимальная триангуляция выпуклого многоугольника



Задача: оптимальная триангуляция выпуклого многоугольника

Триангуляция

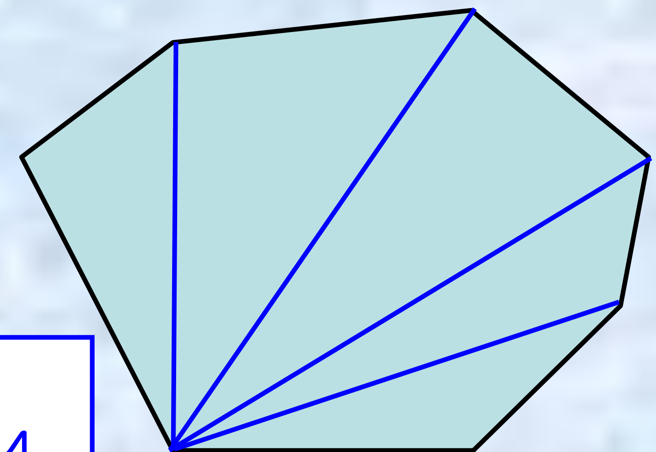
(диагонали не пересекаются внутри многоугольника)



7-угольник

Диагоналей: 4

Треугольников: 5



Выпуклый n-угольник

Число диагоналей: n – 3

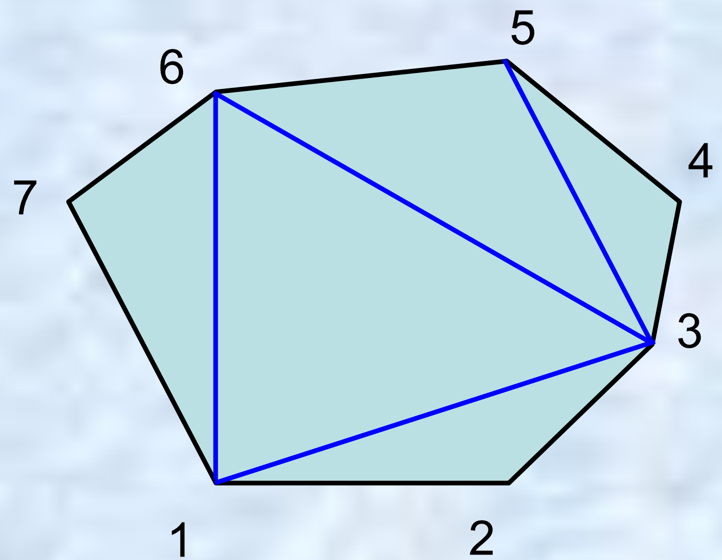
Число треугольников: n – 2

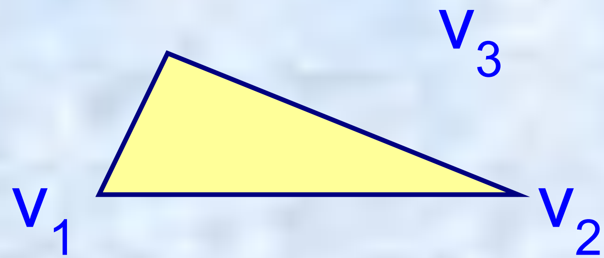
Задача: оптимальная триангуляция многоугольника

На треугольниках определена весовая функция

$w(\Delta v_i v_j v_k)$

Например, $w(\Delta v_1 v_2 v_3) = |v_1 v_2| + |v_2 v_3| + |v_2 v_3|$





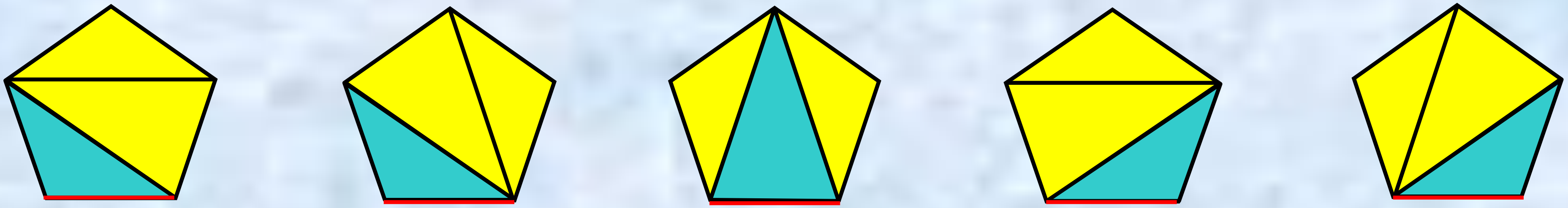
Требуется найти триангуляцию, для которой сумма весов Δ-ков будет *минимальной*

Количество способов триангуляции

Вершин n, диаг. = n – 3 , треуг. = n – 2

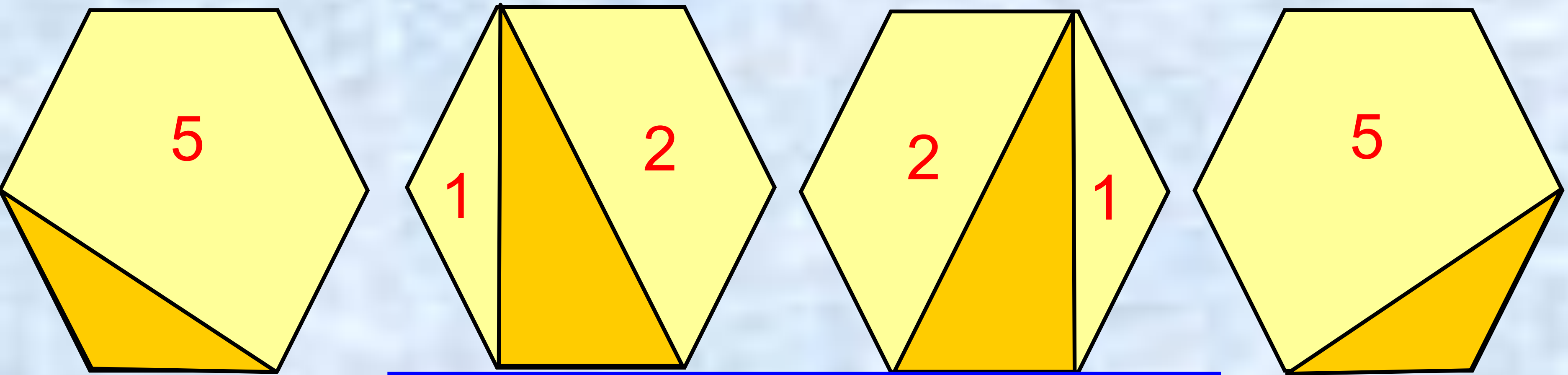
n = 4, диаг. =1, треуг. = 2, вариантов = 2

n = 5, диаг. =2, треуг. = 3, вариантов = 5



Количество способов триангуляции

n = 6, диаг. =3, треуг. = 4, вариантов = 14

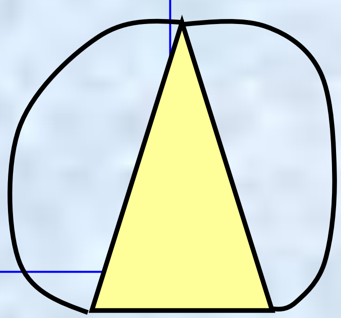


$$d_6 = d_2 d_5 + d_3 d_4 + d_4 d_3 + d_5 d_2$$

$$d_n = \sum_{k=2}^{n-1} d_k d_{n-k+1} = d_2 d_{n-1} + d_3 d_{n-2} + d_4 d_{n-3} + \ldots + d_{n-1} d_2$$

$$d_2 = 1,\ d_3 = 1,\ d_4 = d_2 d_3 + d_3 d_2 = 1 + 1 = 2$$

$$d_n = C_{n-2}$$



Рекуррентная формула для веса оптимальной триангуляции многоугольника (ОТМ)

$$m_{ij} = \min_{i \le k < j}\{m_{ik} + m_{k+1,j} + w(v_{i-1}v_k v_j)\}, \quad i < j,$$

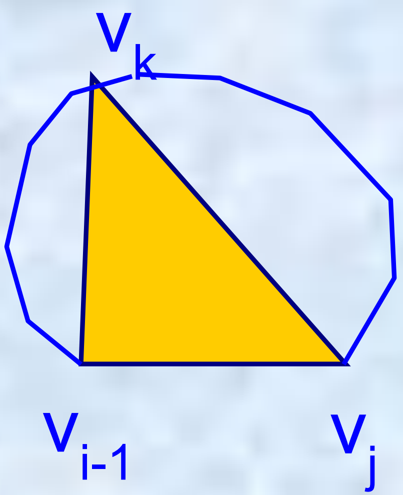
$$m_{ii} = 0$$

m_{ij} – вес ОТМ $(v_{i-1}, v_i, \ldots, v_j)$

m_{1n} – вес ОТМ $(v_0, v_1, \ldots, v_n)$

Для двуугольника $w(v_{i-1}, v_i) = 0$

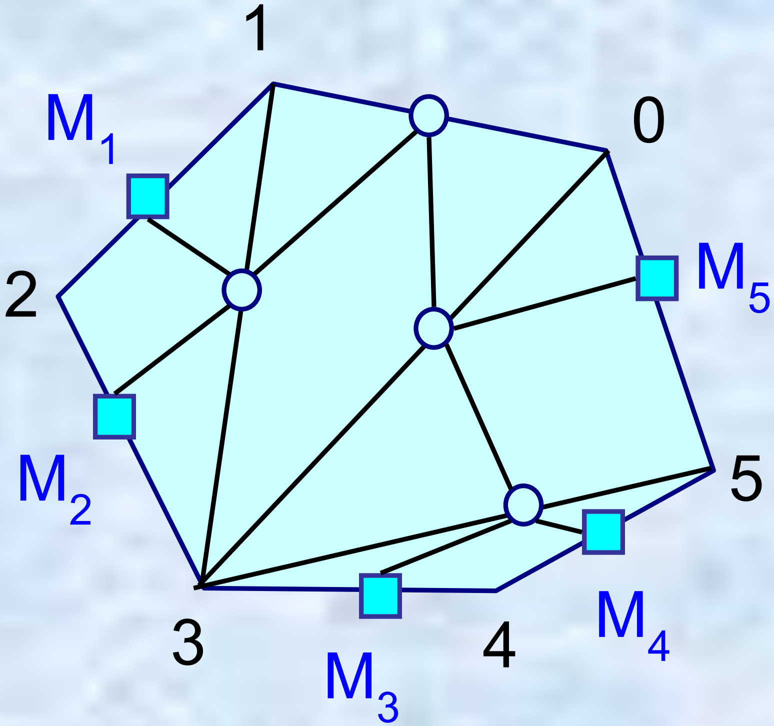
Время $\approx C_1 n^3$, память $\approx C_2 n^2$

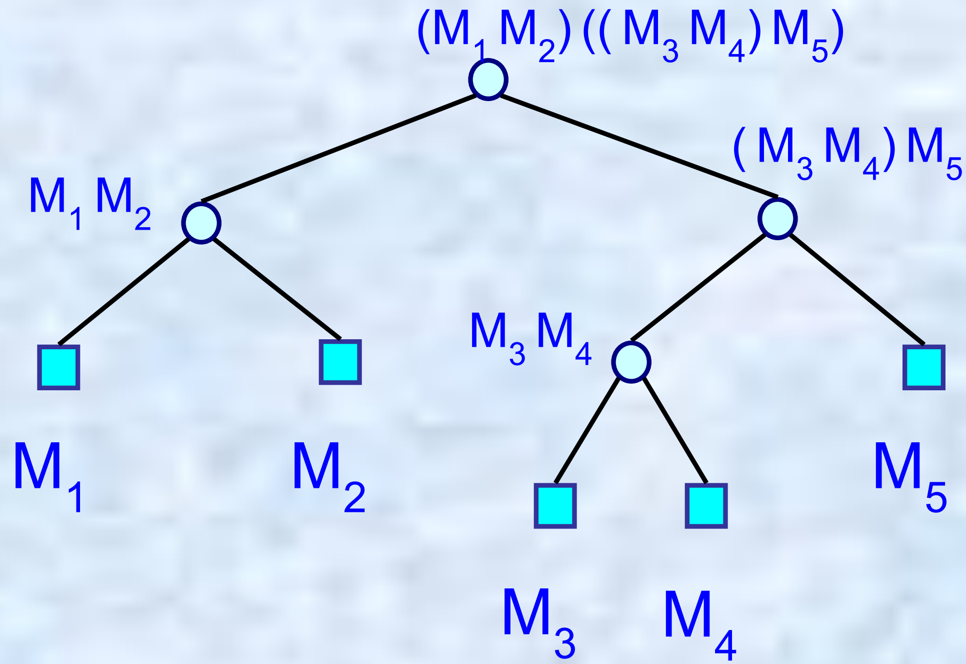


Упражнения

1. Доказать что триангуляция n – угольника содержит n-2 треугольника и n-3 диагоналей.
2. Пусть вес треугольника = его площади. Можно ли упростить алгоритм поиска ОТМ?
3. Весовая функция определена на множестве диагоналей многоугольника. ОТМ – сумма весов диагоналей минимальна. Как свести эту задачу к рассмотренной?

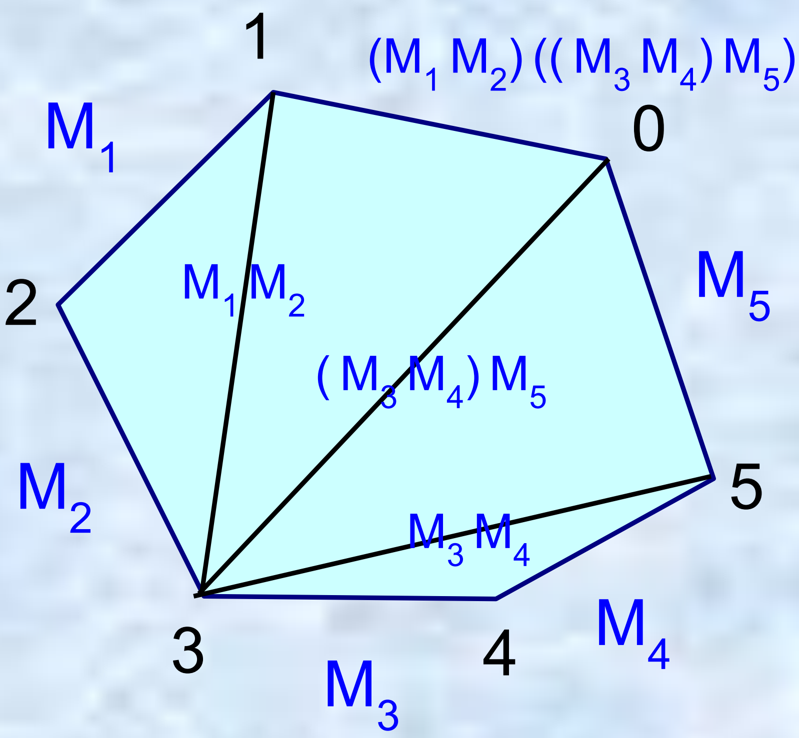
Связь задач





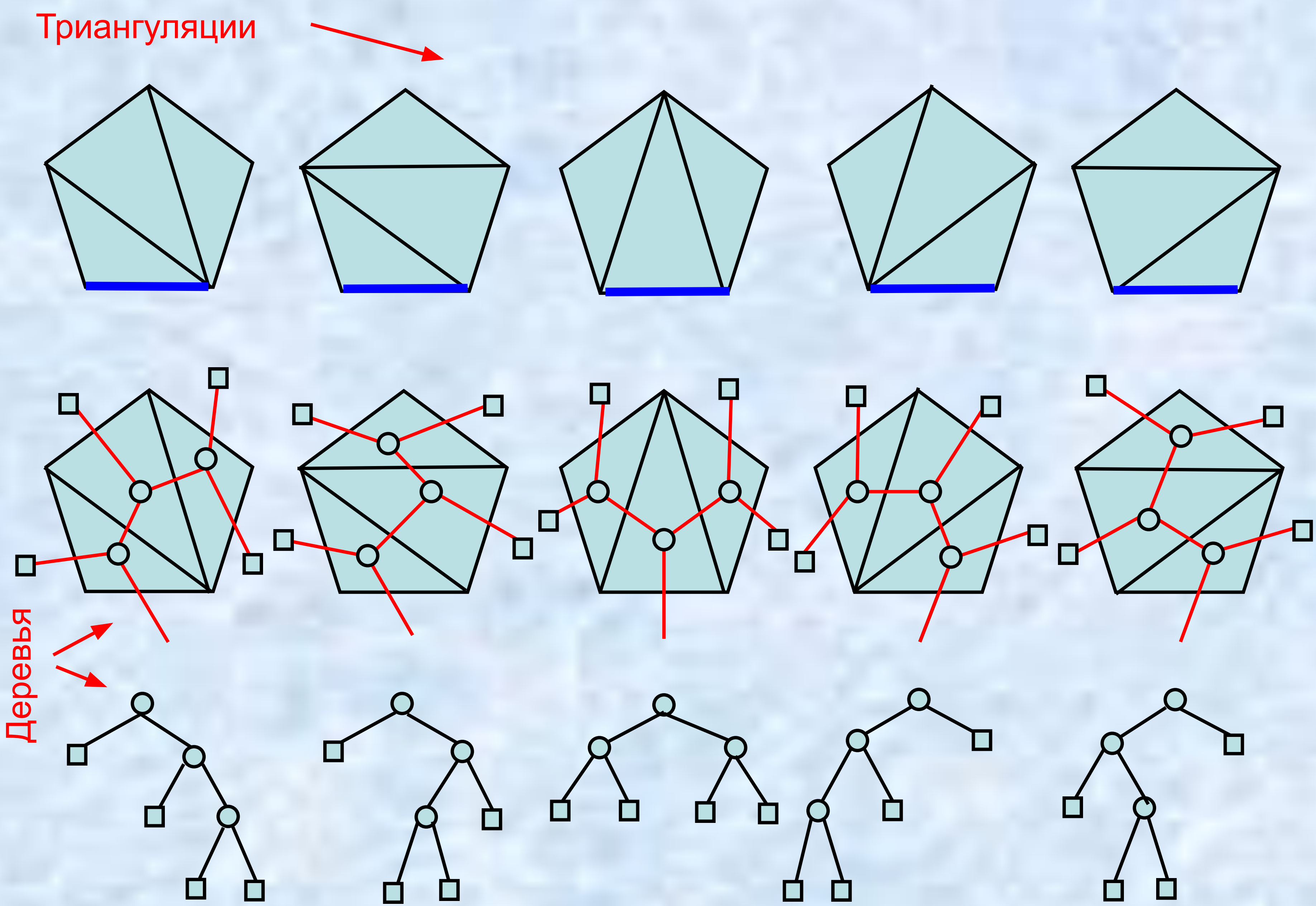
$$(M_1 M_2)((M_3 M_4) M_5)$$

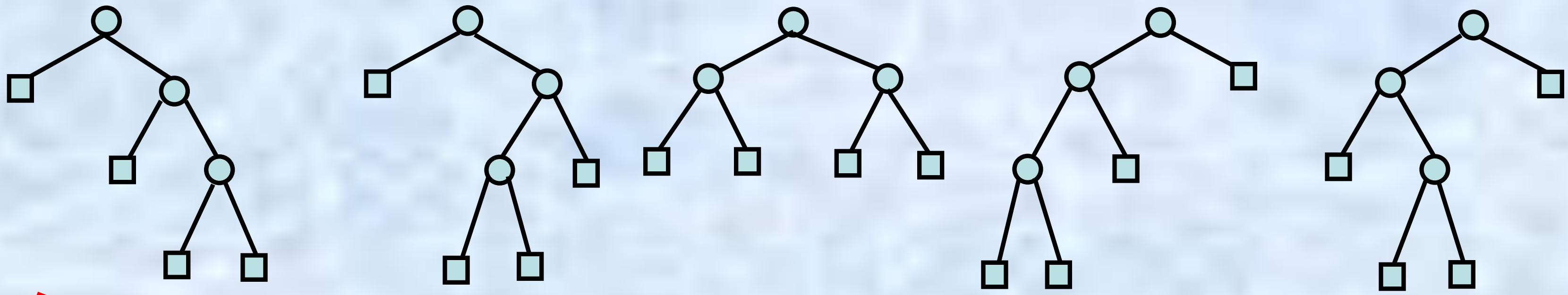
Связь задач



$$(M_1\,M_2)\,((\,M_3\,M_4)\,M_5)$$

$$\boxed{w(\Delta v_i v_j v_k) = r_i\, r_j\, r_k}$$

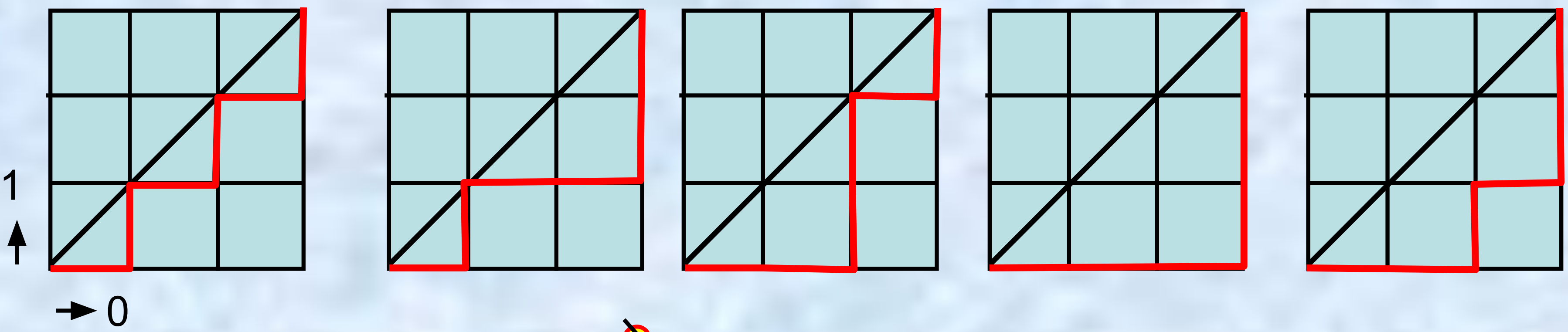

Триангуляции
Деревья

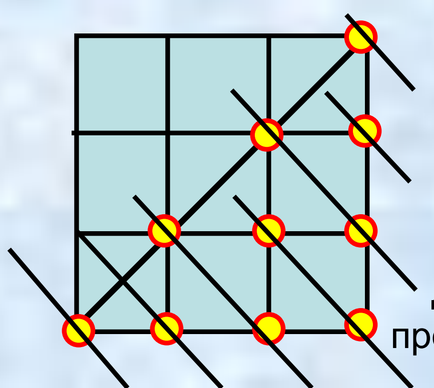


Коды

0 1 0 1 0 1 | 0 1 0 0 1 1 | 0 0 1 1 0 1 | 0 0 0 1 1 1 | 0 0 1 0 1 1

Пути в решетке





Слоистая сеть (спец. вида)

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ

КОНЕЦ   ЛЕКЦИИ