

A collection of historical artifacts is arranged on a light-colored surface. In the top left, a portion of a wooden chessboard with a checkered pattern and several chess pieces is visible. Below the chessboard, there are two medals: one with a red ribbon and a white star, and another with a blue ribbon and a white star. A pair of round-rimmed glasses with thin frames lies diagonally across the center. In the bottom left corner, a small, round, vintage-style compass is partially visible.

ТЕМА 9

ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ



- ◆ Динамическое программирование – это метод, который может быть использован для решения различных оптимизационных проблем, которые предполагают возможность разворачивания их в пространстве и времени



- ◆ *Во многих приложениях динамическое программирование приносит результат, развивая процесс решения от конца к началу, а также разбивая большую громоздкую проблему на серию более мелких, легко поддающихся решению задач*



Классические задачи

- ◆ *распределение капитальных вложений,*
- ◆ *замена оборудования,*
- ◆ *набор высоты самолетом,*
- ◆ *нахождение кратчайшего расстояния на сети,*
- ◆ *определение оптимальной структуры мероприятий по повышению эффективности производства*
- ◆ *и некоторые другие*



Задача

- ◆ У нас две емкости для жидкости: одна 9-литровая, а другая 4-литровая. Необходимо решить, как получить ровно 6 литров жидкости, используя имеющиеся в наличии емкости?

Решение

Количество литров в 9-литровой емкости	Количество литров в 4-литровой емкости
6	0
6	4
9	1
0	1
1	0
1	4
5	0
5	4
9	0
0	0



Петр живет во Львове. Однако ему там не везет, и он решил предпринять путешествие в Донецк в поисках судьбы и удачи. Он планирует проехать все расстояние на автомашине. Причем, учитывая свое финансовое положение, Петр решает так разработать свой маршрут, чтобы в конце каждого дня пути останавливаться на ночлег у друзей, которые живут в Ровно, Виннице, Черновцах, Киеве, Кировограде, Николаеве, Запорожье и Полтаве.



Петр знает, что после 1-го дня пути он может добраться либо до Ровно, либо до Винницы, либо до Черновцов. После 2-х дней пути он может достичь Киева, Кировограда или Николаева. К концу 3-го дня он может быть в Запорожье или Полтаве. И наконец, после 4-х дней пути, он может достичь Донецка. Где должен Петр ночевать в конце каждого дня пути, чтобы минимизировать общее расстояние?

В таблице приведены действительные расстояния между городами



<i>Города</i>	<i>Расстояние</i>
1-2	215
1-3	369
1-4	278
2-5	324
2-6	618
2-7	779
3-5	266
3-6	317
3-7	466
4-5	538
4-6	637
4-7	692
5-8	343
5-9	568
6-8	251
6-9	314
7-8	488
7-9	352
8-10	391
9-10	243

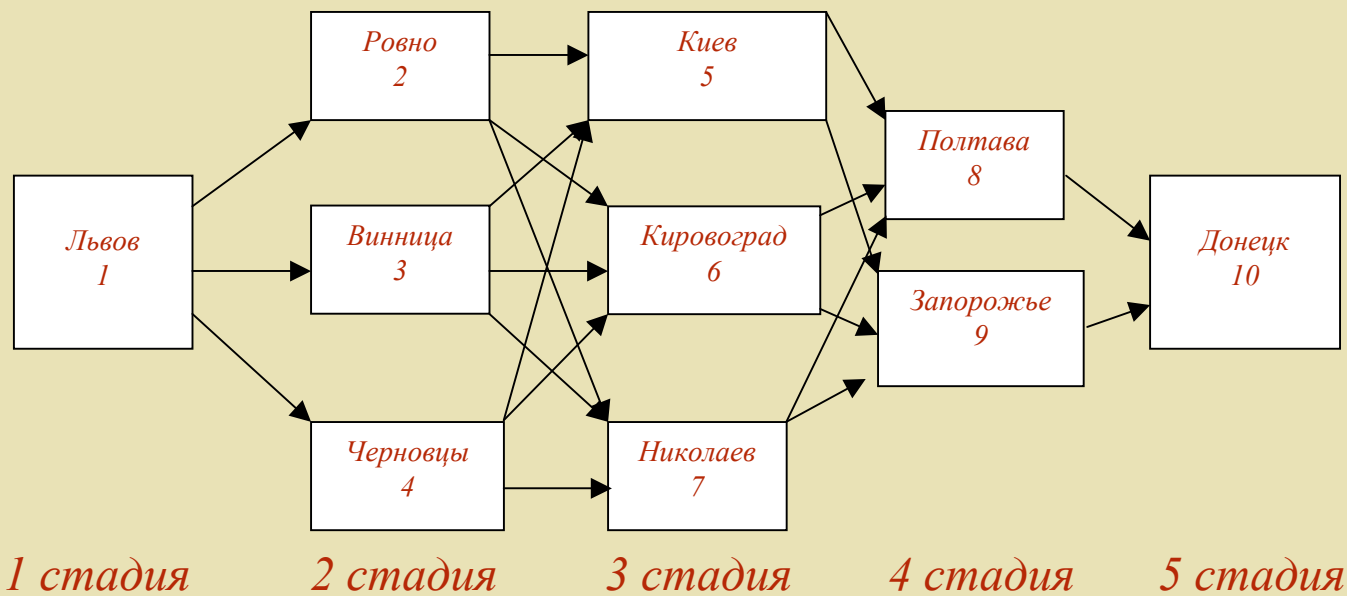


Схема маршрутов движения между городами



C_{ij} – расстояние в км от города i до города j

$f_t(i)$ – кратчайшее расстояние от города i до Донецка, принимая во внимание, что город i принадлежит t -й стадии



4-я стадия расчетов

$f_4(8) = 391$ — кратчайший путь от Полтавы

$f_4(9) = 243$ — кратчайший путь от Запорожья



3-я стадия расчетов

Для 5-го города возможны 2 пути:

1-й путь. Поехать из 5-го в 8-й город и затем выбрать кратчайший путь из 8-го города в 10-й.

2-й путь. Поехать из 5-го в 9-й город и затем выбрать кратчайший путь из 9-го города в 10-й.


$$f_3(5) = \min \begin{cases} c_{58} + f_4(8) = 343 + 391 = 734 \\ c_{59} + f_4(9) = 568 + 243 = 811. \end{cases}$$

Кратчайший путь: 5–8–10



Аналогично находим $f_3(6)$

$$f_3(6) = \min \begin{cases} c_{68} + f_4(8) = 251 + 391 = 642 \\ c_{69} + f_4(9) = 314 + 243 = 557. \end{cases}$$

Кратчайший путь: 6–9–10



Найдем

$$f_3(7) = \min \begin{cases} c_{78} + f_4(8) = 488 + 391 = 879 \\ c_{79} + f_4(9) = 352 + 243 = 595. \end{cases}$$

Кратчайший путь: 7–9–10

2-я стадия расчетов

Самый короткий путь из города 2 должен пролегать либо через город 5, либо через город 6, либо через город 7:

- ◆ 1-й путь. Едем из города 2 в город 5. Затем следуем кратчайшим путем в город 10
- ◆ 2-й путь. Едем из города 2 в город 6. Затем следуем кратчайшим путем в город 10
- ◆ 3-й путь. Едем из города 2 в город 7. Затем следуем кратчайшим путем в город 10


$$f_2(2) = \min \begin{cases} c_{25} + f_3(5) = 324 + 734 = 1058 \\ c_{26} + f_3(6) = 618 + 557 = 1175 \\ c_{27} + f_3(7) = 779 + 595 = 1374. \end{cases}$$

Кратчайший путь: 2–5–8–10

$$f_2(3) = \min \begin{cases} c_{35} + f_3(5) = 266 + 734 = 1000 \\ c_{36} + f_3(6) = 317 + 557 = 874 \\ c_{37} + f_3(7) = 466 + 595 = 1061. \end{cases}$$

Кратчайший путь: 3–6–9–10


$$f_2(4) = \min \begin{cases} c_{45} + f_3(5) = 538 + 734 = 1272 \\ c_{46} + f_3(6) = 637 + 557 = 1194 \\ c_{47} + f_3(7) = 642 + 595 = 1237. \end{cases}$$

Кратчайший путь: 4–6–9–10



1-я стадия расчетов

Кратчайший путь из Львова в Донецк может быть одним из следующих:

- ◆ 1-й путь. Едем из 1-го города во 2-й и затем выбираем кратчайший путь в 10-й город:
- ◆ 2-й путь. Едем из 1-го города в 3-й и затем выбираем кратчайший путь в 10-й город:
- ◆ 3-й путь. Едем из 1-го города в 4-й и затем выбираем кратчайший путь в 10-й город


$$f_1(1) = \min \begin{cases} c_{12} + f_2(2) = 215 + 1058 = 1273 \\ c_{13} + f_2(3) = 369 + 874 = 1243 \\ c_{14} + f_2(4) = 278 + 1194 = 1472. \end{cases}$$

Кратчайший путь: 1–3–6–9–10

Общая длина пути составит **1243 км**



ОСОБЕННОСТИ МЕТОДА ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

◆ 1-я особенность

Проблема требует разбиения на стадии с необходимостью принятия решения на каждой стадии (хотя это бывает и не всегда)

◆ **2-я особенность**

Каждая стадия имеет определенное число состояний, ассоциирующихся с ней. Под состоянием будем понимать информацию, которая необходима на каждой стадии для принятия оптимального решения

◆ **3-я особенность**

Решение, выбранное на любой стадии, показывает, как состояние на данной стадии трансформируется в состояние на следующей стадии

◆ **4-я особенность**

*Для каждого конкретного состояния оптимальное решение на каждой из оставшихся стадий не должно зависеть от ранее достигнутых состояний или ранее принятых решений. Эта идея известна как **принцип оптимальности***



Принцип оптимальности Ричарда Беллмана

- ◆ *Каково бы ни было начальное состояние и принятое начальное решение, каждое последующее решение должно быть оптимальным относительно состояния, полученного в результате предыдущего решения*

◆ **5-я особенность**

Если состояния проблемы были классифицированы так, что они попали под одну из T стадий, то должно быть выражение, которое соотносит затраты или доход, получаемые на стадиях $t, t + 1, \dots, T - 1$ с затратами или доходом, получаемым на стадиях $t + 1, t + 2, \dots, T$



Рекуррентное соотношение

$$f_t(i) = \min_j (c_{ij} + f_{t+1}(j))$$

Решение, оптимальное для каждого промежуточного шага, носит название **условно оптимального решения**