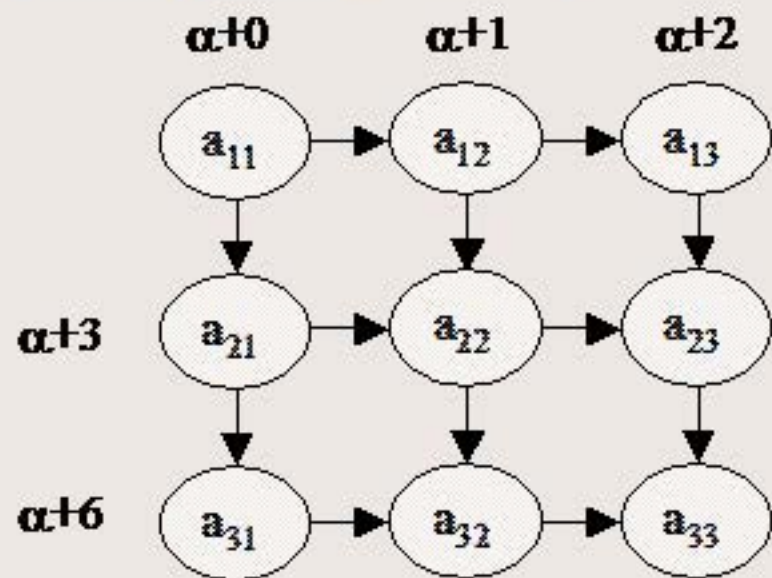


Пример 1.3. Матрица $A=(a_{ij})$



Адрес

$$(a_{ij}) = \alpha + 3 * (i-1) + (j-1)$$

Каждый элемент матрицы участвует в двух отношениях (по строке и по столбцу):

- Следующий по строке – адрес + 1
- Следующий по столбцу – адрес + 3

1. Ленточные матрицы

$$\begin{bmatrix} a_{11} & a_{12} & & & \\ a_{21} & a_{22} & a_{23} & & \\ & a_{32} & a_{33} & a_{34} & \\ & & a_{43} & a_{44} & a_{45} \\ & & & a_{54} & a_{55} \end{bmatrix}$$

- ☑ Для хранения элементов можно выделить непрерывный вектор памяти размера $3 \cdot n - 2$
- ☑ Адрес $(a_{ij}) = \alpha + 3 \cdot (i - 1) + (j - i)$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ & a_{22} & a_{23} & \dots & a_{2n} \\ & & a_{33} & \dots & a_{3n} \\ & & & \dots & \\ & & & & a_{nn} \end{bmatrix}$$

Подход 1

Матрицы подобного вида можно представить как матрицы общего вида и использовать для хранения двухиндексные массивы

- используется память $V_{\text{исп}} = n^2$
- необходимая память $V_{\text{необ}} = n(n+1)/2$

⇒ Эффективность использования памяти $E_{\text{mem}} \approx 0.5$

Подход 2 Исключение элементов ниже главной диагонали (ускорение доступа)

Матрица

Отображение

Вектор памяти

Вектор
указателей
pRow

$$\text{Адрес } (a_{ij}) = \text{pRow}[i] + (j-i), \quad 0 \leq i, j \leq n-1$$

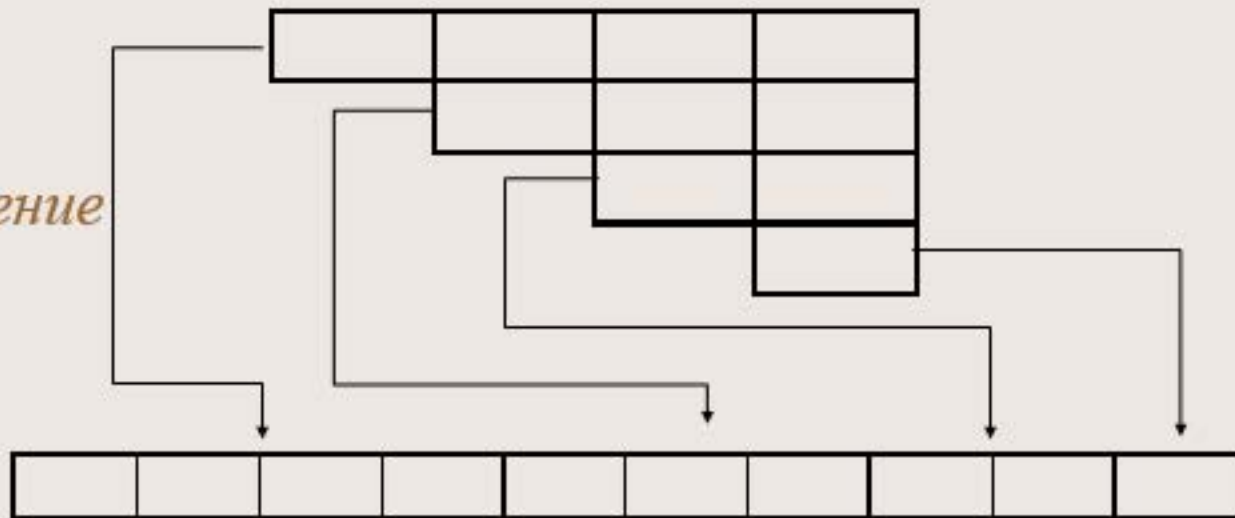


Подход 2 Исключение хранения элементов ниже главной диагонали

Матрица

Отображение

Вектор памяти



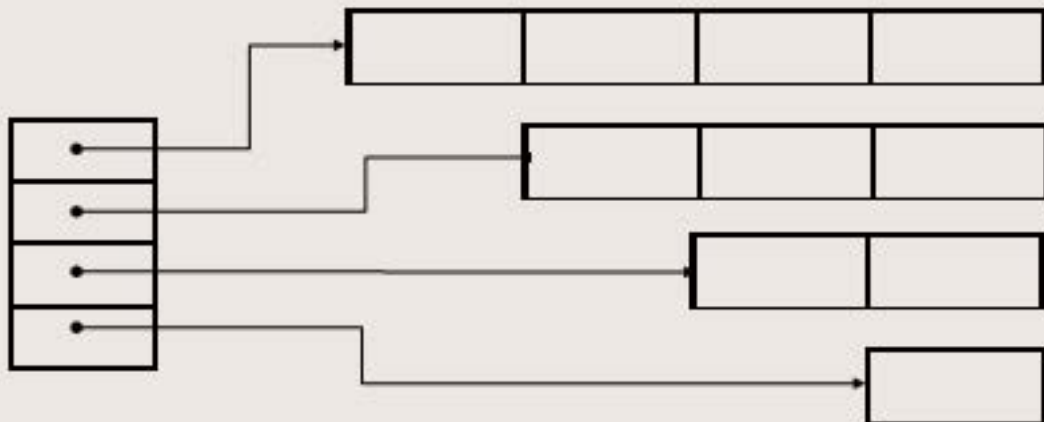
$$\text{Адрес } (a_{ij}) = \alpha + i*n - i*(i-1)/2 + (j-i), \quad 0 \leq i, j \leq n-1$$

Подход 3 Представление матрицы в виде набора векторов

Вектор



Матрица



Подход 4 Матрица как вектор векторных элементов (шаблоны)

Вектор



$1..*$

1

Матрица

