ՀԱՅԿԱԿԱՆ ՍՍՀ ԳԻՏՈՒԹՅՈՒՆՆԵՐԻ ԱԿԱԴԵՄԻԱՅԻ ԶԵԿՈՒՅՑՆԵՐ
ДОКЛАДЫ АКАДЕМИИ НАУК АРМЯНСКОЙ ССР

LXI 1975 5

ПРИКЛАДНАЯ МАТЕМАТИКА

УДК 518.5 : 681.3

А. А. Азатян, Р. А. Тамразян

Об одном методе перечисления элементов списка всех перестановок из n различных символов

(Представлено чл.-корр. АН Армянской ССР Ф. Т. Саркисяном 18/VI 1975)

Среди известных алгоритмов для перечисления элементов списка всех перестановок из n различных символов ([1–8]) одним из наиболее эффективных является алгоритм Джонсона ([4]). Сущность метода Джонсона состоит в последовательном генерировании каждой $(T+1)$—ой n— перестановки из T— ой путем вычисления на каждой стадии позиций $S(T)$ и $S(T)+1$ переставляемых элементов перестановки. Рассматривается равенство

$$T=n!\left|\frac{d_2}{2!}+\frac{d_3}{3!}+\cdot\cdot\cdot+\frac{d_n}{n!}\right|\equiv(d_2,d_3,\cdot\cdot\cdot,d_n), \tag{1}$$

где: $d_k=0$ или 1, или 2, $\cdot\cdot\cdot$, или $k-1$; $d_0=d_1=0$.
Поведение функций d_k задается последовательно конструируемой на каждом T-ом шаге соответствующей n-кой чисел, составляемой с помощью системы счисления с переменным основанием (n-ое число возрастает до величины $n-1$, $(n-1)$-ое — до величины $n-2$, и т. д.). Для того, чтобы получить $(T+1)$-ую перестановку из T-ой, необходимо переставлять символы в позициях $S(T)$ и $S(T)+1$; при этом используются равенства:

$$S(T)=a_T(k)+b_T(k), \tag{2}$$

где: $a_T(k)=\begin{cases}k-d_k, & \text{если } |d_{k-1}+(k-1)d_{k-2}| \text{ — четное;}\\ d_k & \text{в противоположном случае;}\end{cases}$

$$b_T(k)=\begin{cases}\left.\begin{matrix}0, \text{ если } d_k \text{ — четное}\\ 2, \text{ если } d_k \text{ — нечетное}\end{matrix}\right\} \text{ и } k \text{ — четное;}\\ \left.\begin{matrix}0, \text{ если } d_k+d_{k-1} \text{ — четное}\\ 1, \text{ если } d_k+d_{k-1} \text{ — нечетное}\end{matrix}\right\} \text{ и } k \text{ — нечетное;}\end{cases}$$

$$b_T(n-1)=\begin{cases}0, \text{ если } |d_{n-1}+(n-1)\,d_{n-2}| \text{ — четное;}\\ 1, \text{ в противоположном случае;}\end{cases}$$

$$b_r(n) = 0.$$

Анализ соотношений (1) и (2) показывает, что для нахождения адресов $S(T)$ и $S(T)+1$ по методу Джонсона требуется время, порядок которого может быть оценен формулой:

$$\theta_{1n} \simeq (10n + 20)\tau, \tag{3}$$

где: n — длина перестановки,

τ — время короткой операции (типа сложения с фиксированной запятой).

В отличие от известных методов, описываемый метод сводится к матричному управлению процессом образования каждой следующей перестановки из предыдущей путем выборки на каждом шаге определенного адреса из матрицы („треугольника адресов") и транспони-

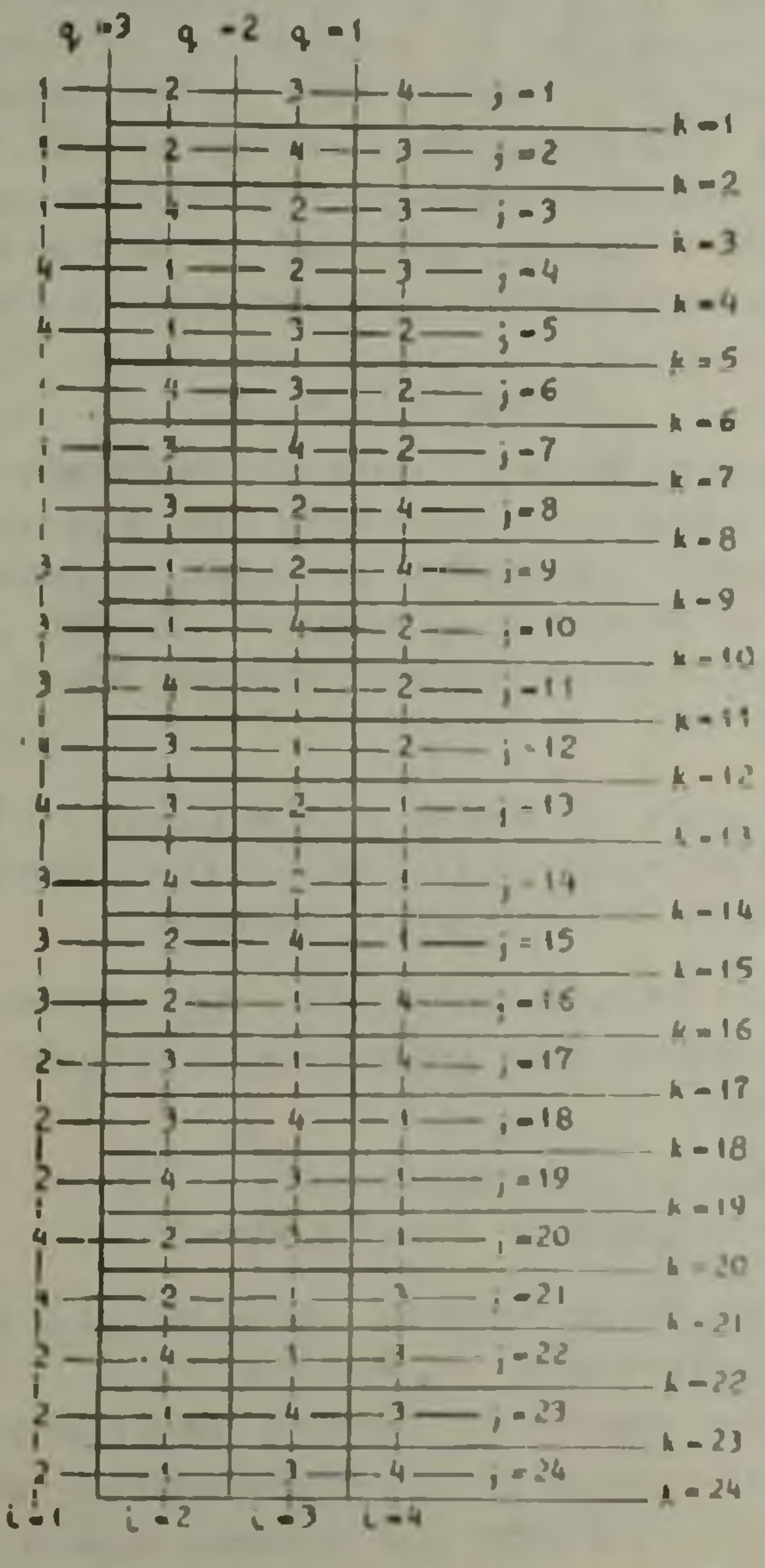

Рис. 1. l, q, j, k — каркас и множество $\{P_n\}$

ровании соответствующей адресу пары переставляемых элементов перестановки. Рассматриваемый метод („метод треугольника адресов") строго обоснован (*), здесь же, ради краткости, ограничимся рассмотрением необходимого набора определений. Для этого сконструируем некоторый i, q, j, k — каркас (рис. 1), представляющий двойную сетку из линий, точки пересечений которых представляют индексы (i, j) $(i \in (1, 2, \cdots, n), j \in (1, 2, \cdots, n!+1))$ и $(q, k)(q \in (1, 2, \cdots, n-1), k \in (1, 2, \cdots, n!)$ и где каждая из пар индексов (q, k) смежна с парами: (i, j), $(i+1, j)$, $(i, j+1)$ и $(i+1, j+1)$; каждую из сеток индексов назовем также „таблицами аргументов" $|i, j|$ и $|q, k|$. Определим числовые функции $\alpha(i, j)(\alpha(i, j) > 0)$ и $\beta(q, k)(0 \leqslant \beta(q, k) \leqslant 1)$ над массивами таблиц $|i, j|$ и $|q, k|$, соответственно, и введем таблицы $|\alpha(i, j)|$ $|\beta(q, k)|$, сопоставляя каждой из пар (i, j) и (q, k) соответствующие значения $\alpha(i, j)$ и $\beta(q, k)$. Каждую из n — последовательностей значений $\alpha(i, j)$ (при j — фиксированном $i = 1, 2, \cdots, n$) (каждую строку таблицы $|\alpha(i, j)|$) назовем „n — перестановкой", если поведение функций $\alpha(i, j)$ таково, что эти n — последовательности всегда одинаковы по составу. Две n — перестановки различны, если они различаются только по порядку расположения элементов. Обозначим через $|P_n|$ и назовем „полным" множество всех различных n — перестановок в том случае, когда число их P_n соответствует равенству

$$P_n = ||P_n|| = n!$$

Две перестановки назовем „взаимно односмежно транспонируемыми", если они могут быть отображены одна в другую путем единственной транспозиции (взаимного обмена местами) некоторых смежных (по индексу i) элементов, а последние назовем „транспонирующими" элементами. Пусть поведение функций $\beta(q, k)$ на i, q, j, k — — каркасе таково, что

$$\beta(q, k) = \begin{cases} 1, \text{ если } \alpha(i, j) = \alpha(i+1, j+1), \ \alpha(i, j+1) = \alpha(i+1, j), \\ \text{а элементы } \alpha(i, j) \text{ и } \alpha(i+1, j) \text{ — транспонирующие;} \\ 0, \text{ во всех остальных случаях.} \end{cases}$$

„Адресом односмежной транспозиции A^n_{qk}" назовем соответствующую этой транспозиции пару аргументов (q, k) таких, чтобы удовлетворялись условия:

$$A^n_{qk} \begin{cases} = (q, k), \text{ если } \beta(q, k) = 1; \\ \text{не существует, если } \beta(q, k) = 0. \end{cases}$$

При необходимости будем говорить также о „q_n — адресе", имея ввиду соответствующий адресу A^n_{qk} аргумент q_n.

„Треугольником адресов A^n_{sq}" назовем совокупность q_n—адресов, расположенных по строкам A^n_{sq}, перенумерованным сверху вниз индексом $s (s = 0, 1, \cdots, n-2)$, при условии, если в каждой s-ой строке расположены q_{n-s} — адреса, удовлетворяющие соотношению:

$$q_{n+1} = \begin{cases} q_n + s & \text{для всех } q_n \text{ — четных,} \\ q_n + t(s-1) & \text{для всех } q_n \text{ — нечетных;} \end{cases} \qquad (4)$$

где:

$$t(s) = \begin{cases} s + (-1)^s \left| \dfrac{1^{s+1} + (-1)^{s-1}}{2} \right| & \text{при } n \text{ — четных.} \\ s + (-1)^{s+1} \left| \dfrac{1^s + (-1)^s}{2} \right| & \text{при } n \text{ — нечетных.} \end{cases}$$

В ([8]) строится такая последовательность адресов A^n_{qs}, которая перечисляет все элементы без повторений из $\{P_n\}$ с помощью взаимно односмежных транспозиций и может быть расположена в треугольник адресов A^n_{sq}, удовлетворяющий соотношению (4).

Таким образом, при каждом фиксированном n треугольнику адресов A^n_{sq} взаимно-однозначно соответствует последовательность адресов A^n_{qs}, перечисляющая множество элементов $\{P_n\}$. В ([8]) также показано, что каждый треугольник A^n_{sq} содержит все треугольники A^{n-2}_{sq}, A^{n-4}_{sq}, . . . , а каждый треугольник A^{n-1}_{sq} — все треугольники A^{n-3}_{sq}, A^{n-5}_{sq}, . . . ; так что, если для некоторого N(N — как угодно большое, конечное) с помощью соотношений (4) построены треугольники A^N_{sq} и A^{N-1}_{sq}, то тем самым заданы все треугольники A^n_{sq}, $n \leq N$. Пусть, например, n — четное и N — нечетное. Тогда, выбирая треугольник A^{N-1}_{sq}, „вырезываем" из него (отсчитывая с прямоугольной вершины по катетам $n-1$ элементов) треугольник A^n_{sq}. Наличие таких треугольников дает возможность перечислить элементы $\{P_n\}$ для всех $n \leq N$ методом матричного управления (методом треугольника). Суть метода описывается следующим алгоритмом (на рис. 1—3 описывается пример при $n=4$):

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	15	13	13	11	11	9	9	7	7	5	5	3	3	1
		13	14	11	12	9	10	7	8	5	6	3	4	1
			15	13	13	11	11	9	9	7	7	5	5	3
				13	14	11	12	9	10	7	8	5	6	3
					15	13	13	11	11	9	9	7	7	5
						13	14	11	12	9	10	7	8	5
							15	13	13	11	11	9	9	7
								13	14	11	12	9	10	7
									15	13	13	11	11	9
										13	14	11	12	9
											15	13	13	11
												13	14	11
													15	13
														13

Рис. 2. Треугольник A^{16}_{sq}, „вырезка" треугольника A^4_{sq}

1. Исходя из четности либо нечетности n, выбрать один из треугольников A^{N}_{sq} либо A^{N-1}_{sq} (пусть, например, n — четное, и N — — четное, тогда необходимо выбрать A^{N}_{sq}) (Рис. 2.).

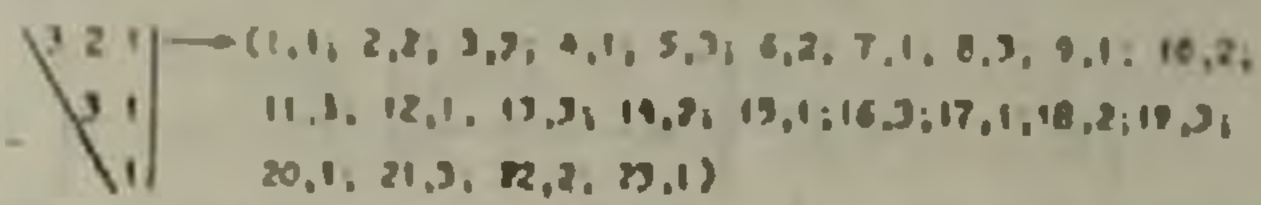

Рис. 3. Разворачивание треугольника A^{4}_{sq} в последовательность A^{4}_{qk}

2. „Вырезать" из треугольника A^{N}_{sq} треугольник A^{n}_{sq} (см. рис 2).

3. „Разворачивать" пошагово треугольник A^{n}_{sq} в последовательность q_n — адресов согласно правилу: выписывать последовательно значения q_{n-s} — адресов (при $s = 0, 1, 2, \cdot \cdot \cdot, n-2$) так, чтобы после каждого полного перечисления подпоследовательности q_{n-s} — адресов выписывался один q_{n-s-1} — адрес, после каждого полного перечисления подпоследовательности q_{n-s-1} — адресов выписывался один q_{n-s-1} — адрес, $\cdot \cdot \cdot$ (рис. 3).

4. Из n различных символов (значений функции $\alpha(i, j)$ ($j = 1$; $i = 1, 2, \cdot \cdot \cdot, n$)) составить некоторую n — последовательность (рис. 1).

5. В соответствии с каждым k — ым шагом п. 3 (рис. 3) осуществлять по одной односмежной транспозиции в каждой k-ой n — перестановке, образовывая тем самым каждую $(k+1)$-ую n — перестановку (рис. 1).

Данный метод отличается от известных тем, что использование треугольника адресов A^{n}_{sq} в качестве управляющей матрицы дает возможность почти полного исключения арифметических операций, резкого сокращения количества команд и существенного уменьшения объема памяти.

При использовании метода „треугольника адресов" для осуществления генерирования каждой следующей перестановки из предыдущей требуется время, порядок которого оценивается формулой:

$$\varepsilon_{2n} \simeq 20\,\tau. \tag{5}$$

Из (3) и (5) имеем:

$$\lambda_n = \frac{\varepsilon_{1n}}{\varepsilon_{2n}} = 0{,}5n + 1.$$

На практике большее значение имеет не величина λ_n, а разница во временах, затрачиваемых на процесс генерирования всех элементов $|P_n|$ при использовании метода Джонсона и метода „треугольника адресов". Очевидно, эта разница существенно возрастает с возрастанием факториальной функции $n!$ и ставит практические ограничения для реализации более жесткие для метода Джонсона.

Программа, составленная на языке Ассемблер для машин семейства ЕС/ЭВМ с операционной системой ДОС/ЕС, отлажена на машине ЕС

1030. С учетом требования большинства прикладных задач и производительности машины ЕС 1030 число n ограничивается следующим образом: $2 \leqslant n \leqslant N = 16$; хотя нет никаких принципиальных ограничений для использования программы при $N > 16$. Объем программы не зависит от N и требует область оперативной памяти (ОП) в 1492 байта, что обеспечивает ее функционирование в любых машинах семейства ЕС/ЭВМ в исполнении с минимальной памятью. Применение ДОС/ЕС дает возможность загружать программу в память в любой момент рабочего времени. Программа имеет возможность многократного перечисления полных множеств элементов $\{P_n\}$ ($1 \leqslant C \leqslant 99$, где C — кратность перечисления). Информация о режиме работы (n, C) вводится в память с помощью пишущей машинки.

В зависимости от требований прикладной задачи каждую очередную перестановку символов можно либо выводить на периферийные устройства, либо непосредственно использовать для динамического управления или в целях передачи информации.

Блок — схема программы приведена на рис. 4

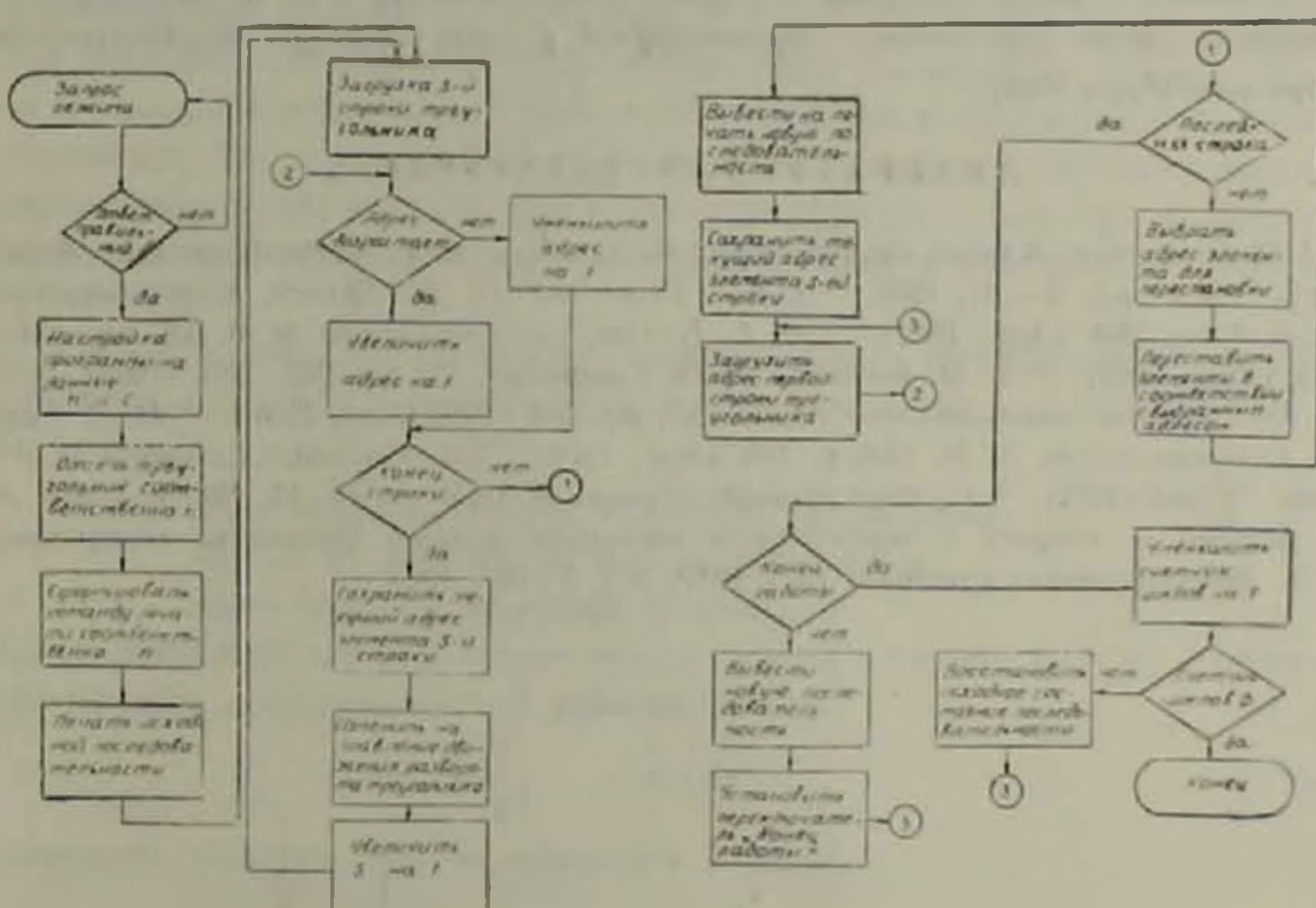

Рис. 4. Блок - схема программы

Контрольными тестовыми задачами являлись задачи по перечислению элементов $\{P_n\}$ при $n = 3, 4, \cdot \cdot \cdot, 9$. Результаты решения задач (списки элементов) проверялись по числу ожидаемых элементов в $\{P_n\}$ и на неповторяемость и дали точное совпадение с теоретически ожидаемыми. Зависимость времени перечисления элементов $\{P_n\}$ от величины n ($T_{б.п.} = f_1(n)$ и $T_{п.} = f_2(n)$, где $T_{б.п.}$ — время перечисления без вывода на печать (в сек) и $T_{п.}$ — время перечисления с печатью (в мин)) при $n = 3, 4, 5, 6, 7, 8, 9$ выражается, соответственно: $T_{б.п.} = 9{,}6 \cdot 10^{-3}$; $3{,}84 \cdot 10^{-2}$; $1{,}92 \cdot 10^{-1}$; 1,15; 8,05; 64.40; 580.00 и $T_{п.} = 6{,}86 \cdot 10^{-3}$; $2{,}74 \cdot 10^{-2}$; $1{,}37 \cdot 10^{-1}$; $8{,}23 \cdot 10^{-1}$; 5,76; 46,01; 414,88.

Ереванский НИИ математических машин

Ա. Հ. ԱԶԱՏՅԱՆ, Ռ. Ա. ԹԱՄՐԱԶՅԱՆ

Տարբեր սիմվոլներից կազմված n-տեղափոխությունների լրիվ բազմության թվարկման մի մեթոդի մասին

Ամեն մի ամբողջական n-ի համար տրվում է տարբեր n սիմվոլներից կազմված n-տեղափոխությունների լրիվ բազմության թվարկման մեթոդ, որը ի տարբերություն հայտնի մեթոդներից հիմնված է տեղափոխությունների ստանալու պրոցեսի աղյուսակային ղեկավարման սկզբունքի վրա. Այդ մեթոդի կիրառումը հնարավորություն է տալիս համարյա լրիվ կրճատելու թվաբանական օպերացիաները և բերում է հրամանների թվի ու հիշողության ծավալի խիստ կրճատման: Նկարագրվում է այդ մեթոդի մեքենայական կիրառումներից մեկը:

ЛИТЕРАТУРА — ԳՐԱԿԱՆՈՒԹՅՈՒՆ

[1] *D. H. Lehmer*, Applied Combinatorial Mathematics, E. F. Beckenback, Ed., Wiley New York, p.p. 5—31, 1964. [2] *I. E. L. Peck* and *G. F. Shrack*, Communications ACM, 5,4 p. 208 (Apr., 1962). [3] *H. F. Trotter*, Communications ACM, 5,8, p.p. 434—435, (Aug., 1962). [4] *S. M. Johnson*, Math. Computing, 17, p.p. 282—285. (1963). [5] *G. G. Jr. Langdon*, Communications ACM, 10,5, p.p. 298—299. (May, 1967). [6] *D. A. Pager*, Communications ACM, 13,3, p. 193. (Mar., 1970). [7] *K. Harada*, „Communs ACM", June, 14, №6 (1971). [8] *C. Read Ronald*, „Communs ACM", 1973, 15, №8, p. 775. [9] *А. А. Азатян*, К вопросу о перечислении элементов полного множества ассоциаций, [U]. Депонированная рукопись, 1974, РНР, №3, 3—697. 1975