

Поскольку матрицы являются двумерными или одномерными массивами, вместо слов «матрица», «простая матрица» и «гнездовая матрица» часто используются слова «массив», «простой массив» и «гнездовой массив». При выводе массивы могут представляться или в раскрытом виде, как это показано выше в правых частях присваиваний, или в свернутом виде. В последнем случае виден только верхний уровень вложенности из скалярных или строковых элементов и размеры массивов следующего уровня вложенности в формах $[n \times m]$, где n — число строк и m — число столбцов. Если, например, вывести определенные выше значения переменных $w1$ и $w2$, а затем свернуть их, то получим:

$$w1 = \begin{bmatrix} 1 & -3 & 7 \\ 7 & 6 & [1 \times 2] \end{bmatrix} \quad w2 = \begin{bmatrix} [1 \times 2] & 3 & 7 \\ 4 & 7 & [1 \times 2] \end{bmatrix}$$

Гнездовой массив удобно интерпретировать деревом, корнем которого является сам массив, а от него идут дуги к элементам — скалярам, строкам, простым и гнездовым массивам. От вложенных массивов снова идут дуги к их элементам — скалярам, строкам, простым и гнездовым массивам и т. д. Листьями подобного дерева являются скаляры или строки — конечные элементы, не имеющие последующих ссылок. Исходя из этого, для гнездовых массивов можно использовать многие термины, применяемые для деревьев, например, «листья», «уровни вложенности элементов», «высота». Высотой гнездового массива назовем высоту соответствующего ему дерева, т.е. максимальное из количеств дуг на путях от корня до листьев.

Поскольку гнездовые массивы определяются рекурсивно, то и наиболее простая и эффективная их обработка реализуется рекурсивными алгоритмами [2]. Как объекты хранения данных гнездовые массивы изучены слабо, а встроенных в среду *PTC Mathcad Prime* средств для работы с ними мало. Ряд рекурсивных функций для работы с гнездовыми массивами предложен в работах [3—6]. В [3; 5] речь идет о функциях общего назначения, а в [4; 6] — о функциях, позволяющих проводить «векторные» операции, то есть действия сразу над каждым из элементов массива.

Простой поиск и замещение. В этом разделе сформулированы задачи простого поиска и замещения элементов в гнездовых массивах (*F*, *G*, *H* и *I*). В конце статьи в разделе I приведенного фрагмента *mathcad*-документа предложены рекурсивные функции для решения указанных задач.

Задачи F. Элементы и их позиции

F1. Создать функцию, которая по гнездовой матрице *ma* строит матрицу позиций всех элементов *ma* (кроме *ma*). Строки матрицы-результата должны иметь вид $[ele\ pos]$, где *ele* — элемент *ma*, *pos* — позиция элемента *ele* в *ma*.

F2. Сформировать функцию, обратную к функции, решающей задачу *F1*. То есть требуется написать функцию, которая результат вычислений по функции для *F1* переводит в матрицу *ma*.

Задачи G. Выборки элементов из гнездовой матрицы по их позициям

G1. Построить функцию, возвращающую элемент гнездовой матрицы *ma* из позиции *p*. (*p* — простая матрица с двумя столбцами, в строках которой размещены позиции по уровням вложенности).

G2. Построить функцию, возвращающую вектор тех элементов гнездовой матрицы ma , которые расположены в позициях, указанных компонентами вектора vp .

В следующих задачах под объектом понимается все то, что может являться элементом гнездовой матрицы, то есть любое число, строка, простой или гнездовой массив из чисел и (или) строк.

Задачи H. Вхождение объекта в гнездовую матрицу

H1. Выяснить, сколько раз объект x входит как элемент в гнездовую матрицу ma . При наличии вхождений вернуть матрицу $[x\ n]$, где n — количество вхождений, в противном случае — слово “нет”.

H2. Выяснить, входит ли объект x как элемент в гнездовую матрицу ma . Если да, то вернуть вектор позиций вхождения, если нет — слово “нет”.

Замечание. Если $x = ma$, то, вообще говоря, x не имеет позиции в ma . Будем считать, что в этом случае условно $p = [“x”\ “y”]$.

Задачи I. Замещение объектов в гнездовой матрице

I1. Заместить все вхождения объекта x в гнездовую матрицу ma ($x \neq ma$) объектами y .

I2. Заместить элемент из позиции pos гнездовой матрицы ma объектом y .

I3. Пусть vp — вектор позиций в ma , vy — вектор объектов. Заместить элементы гнездовой матрицы ma в позициях vp соответствующими компонентами vy . Если в vy есть лишние объекты, то они должны игнорироваться. Если в vy недостает объектов, то вместо них необходимо использовать последнюю компоненту vy .

Несколько слов относительно функций решения задачи **F1**. Рекурсивная функция *elepos* имеет два аргумента ma и p , где ma — гнездовая матрица и p — вспомогательный параметр в виде матрицы размером 1×2 для формирования позиций элементов при рекурсивных вызовах. Начальное значение для p роли не играет и его можно взять, например, в виде $[0\ 0]$. По *elepos* формируется матрица ot всех элементов ma и их позиций, включая и саму матрицу ma . Причем ma в ot позиции не имеет и включается в результат ot с любой псевдопозицией, например, $[“x”\ “y”]$. Однако по условиям задачи ma в ot вообще входить не должна. Поэтому и написана головная программа *maelepos*, по которой реализуется обращение к *elepos* и из сформированной матрицы ot удаляется начальная строка с ma . Аргумент *maelepos* принимает лишь исходную гнездовую матрицу ma . Краткие замечания по функциям, решающим другие задачи, даны на самом фрагменте *mathcad*-документа.

Обобщенный поиск и замещение. В этом разделе сформулированы задачи обобщенного поиска и замещения элементов в гнездовых массивах (J , K и L). В конце статьи в разделе II приведенного фрагмента *mathcad*-документа предложены рекурсивные функции для решения указанных задач. Основополагающей из них является функция *equG(ma1, ma2)*, играющая роль оператора «обобщенного логического равенства», т.е. проверяющая, являются ли два гнездовых массива $ma1$ и $ma2$ обобщенно равными.

Прежде чем формулировать решаемые задачи, нам необходимо ввести понятие обобщенного равенства двух гнездовых массивов A и B . К этому и приступим.

Будем считать, что в A или в B могут быть элементы, равные любому другому элементу. Прежде всего, надо условиться об обозначении таких «обобщенных» элементов. Если в A и (или) в B нет специальных элементов NaN (*Not a Number*), то данный скаляр подходит для наших целей. При наличии в A или в B элементов NaN в качестве обобщенного значения может быть взято любое достаточно большое число или какая-либо строка, которые не встречаются ни в A , ни в B . Будем считать, что для обобщенных значений используется именно скаляр NaN .

Определение 1. Два гнездовых массива $ma1$ и $ma2$ обобщенно равны, если у них на верхнем уровне вложенности совпадают количества строк и столбцов, а все их соответствующие друг другу элементы равны, причем считается, что при любых a из $ma1$ и b из $ma2$: $(NaN = b) = 1$, $(a = NaN) = 1$, $(NaN = NaN) = 1$ (заметьте, что в *PTC Mathcad* $(NaN = NaN) = 0!$) [1. С. 465].

Определение 2. Пусть объект obj — скаляр, строка, простой или гнездовой массив, ma — простой или гнездовой массив. Говорят, что obj обобщенно входит в ma , если obj обобщенно равно ma или obj обобщенно равно какому-либо элементу ma на любом его уровне вложенности. Поиск обобщенных вхождений obj в ma будем называть обобщенным поиском [1. С. 465].

Задача J. Обобщенное равенство

Построить функцию, проверяющую, являются ли два объекта (числа, строки, простые или гнездовые массивы) обобщенно равными.

Задачи K. Обобщенные вхождения объектов в гнездовые массивы

K1. Выяснить, сколько раз объект x обобщенно входит в гнездовую матрицу ma . При наличии таких вхождений вернуть матрицу $[x\ n]$, где n — количество вхождений x в ma , при отсутствии вхождений — слово “нет”.

K2. Выяснить, имеются ли обобщенные вхождения объекта в гнездовую матрицу ma . Если да, то вернуть вектор позиций вхождения, если нет — слово “нет”.

K3. Выяснить, имеются ли обобщенные вхождения объекта в гнездовую матрицу ma . Если да, то вернуть матрицу позиций вхождения, строки которой имеют вид $[ele\ pos]$, где ele — образ x в ma , pos — позиция ele в ma . При отсутствии вхождений должно возвращаться слово “нет”.

Задачи L. Замещение объектов в гнездовой матрице

L1. Заменить все обобщенные вхождения объекта x в гнездовой матрице ma ($x \neq ma$) объектами y .

L2. Заменить элемент из позиции pos гнездовой матрицы ma объектом y .

L3. Пусть $vpos$ — вектор позиций в ma , vy — вектор объектов. Заменить элементы гнездовой матрицы ma в позициях $vpos$ соответствующими компонентами vy . Если в vy есть лишние объекты, то они должны игнорироваться, если в vy недостает объектов, то вместо них необходимо использовать последнюю компоненту vy .

Замечание. Задачи **L2** и **L3** связаны с замещением элементов в гнездовых массивах на конкретных позициях. Поскольку такое замещение не опирается на наличие или отсутствие в массиве элементов NaN , для решения **L2** и **L3** нет необходимости строить новые функции. Решать их можно ранее созданными функциями *repo* и *reprov*.



Поиск и замещение в гнездовых массивах

Создаваемые функции решения сформулированных задач будем проверять на гнездовых массивах из соответствующих разделов или на гнездовом массиве:

$$a0 := \left[\begin{array}{cc} 5 & 7 \\ 2 & [2 \ 19] \end{array} \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & 1 \end{array} \right] \left[\begin{array}{cc} [2 \ 19] \\ 7 & \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & 1 \end{array} \right] \end{array} \right] \right]$$

I. Простой поиск и замещение элементов в гнездовых массивах

Решение задач F

F1. Для решения задачи $F1$ предлагается рекурсивная функция $elepos(ma, p)$, где ma - гнездовая матрица и p - вспомогательный параметр для формирования позиций элементов при рекурсивных вызовах. По $elepos$ формируется матрица ot всех элементов ma и их позиций, включая и саму матрицу ma .

```

elepos(ma, p) := || ot ← [ ma p ]
                  || for j ∈ 0..cols(ma) - 1
                  ||   || for i ∈ 0..rows(ma) - 1
                  ||   ||   || [ b p ] ← [ mai,j [ i j ] ]
                  ||   ||   || if IsArray(b)
                  ||   ||   ||   || r ← elepos(b, p)
                  ||   ||   ||   || if rows(r) > 1
                  ||   ||   ||   ||   || for k ∈ 1..rows(r) - 1
                  ||   ||   ||   ||   ||   || rk,1 ← stack(p, rk,1)
                  ||   ||   ||   || else
                  ||   ||   ||   ||   || r ← [ b p ]
                  ||   ||   ||   ||   || ot ← stack(ot, r)
                  ||   ||   ||   ||
                  ||   ||   ||
                  ||   ||
                  || ot
    
```

```

maelepos(ma) := || ot ← elepos(ma, [0 0])
                 || submatrix(ot, 1, rows(ot) - 1, 0, 1)
    
```

По $maelepos$ из матрицы-результата ot , сформированного рекурсивной функцией $elepos$ удаляется первая строка

```

t := maelepos(a0)           t1 := submatrix(t, 0, 11, 0, 1)
t2 := submatrix(t, 12, 22, 0, 1)   t3 := submatrix(t, 23, rows(t) - 1, 0, 1)
    
```


$$pick \left(a0, \begin{bmatrix} 1 & 3 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \right) = [19 \ 3] \quad \left((a0_{1,3})_{0,1} \right)_{0,1} = [19 \ 3] \quad (a0_{1,3})_{0,1} = \begin{bmatrix} 2 & [19 \ 3] \\ 4 & 1 \end{bmatrix}$$

G2. Для решения задачи G2 предлагаются функция *pickv* или рекурсивная функция *pickvr*.

$$pickv(ma, vp) := \left\| \begin{array}{l} \text{for } m \in 0..rows(vp)-1 \\ \quad [ot \ p] \leftarrow [ma \ vp_m] \\ \quad \text{for } k \in 0..rows(p)-1 \\ \quad \quad [i \ j] \leftarrow [p_{k,0} \ p_{k,1}] \\ \quad \quad ot \leftarrow ot_{i,j} \\ \quad \quad wp_m \leftarrow ot \\ \quad wp \end{array} \right\| \quad pickv \left(a0, \begin{bmatrix} 1 & 3 \\ 0 & 1 \\ 0 & 1 \\ 1 & 3 \\ 0 & 1 \end{bmatrix} \right) = \begin{bmatrix} [19 \ 3] \\ 7 \\ [2 \ [19 \ 3]] \\ [4 \ 1] \end{bmatrix}$$

$$pickvr(ma, vp) := \left\| \begin{array}{l} ot \leftarrow [pick(ma, vp_0)] \\ \text{if } last(vp) > 0 \\ \quad \left\| \begin{array}{l} vp \leftarrow submatrix(vp, 1, last(vp), 0, 0) \\ stack(ot, pickvr(ma, vp)) \end{array} \right\| \end{array} \right\| \quad t := pickvr \left(a0, \begin{bmatrix} 0 & 1 \\ 1 & 3 \\ 0 & 1 \end{bmatrix} \right) \\ t = \begin{bmatrix} 7 \\ [2 \ [19 \ 3]] \\ [4 \ 1] \end{bmatrix}$$

Решение задач H

H1. Для решения задачи H1 предлагается функция *in*.

$$in(x, ma) := \left\| \begin{array}{l} M \leftarrow elepos(ma, ["x" "x"]) \\ n \leftarrow \sum_{k=0}^{rows(M)-1} (x = M_{k,0}) \\ \text{if } (n = 0, "no", [x \ n]) \end{array} \right\|$$

H2. Для решения задачи H2 предлагается функция *inv*.

$$inv(x, ma) := \left\| \begin{array}{l} [M \ p] \leftarrow [elepos(ma, ["x" "x"]) \ 0] \\ \text{for } k \in 0..rows(M)-1 \\ \quad \left\| \begin{array}{l} \text{if } x = M_{k,0} \\ \quad \left\| \begin{array}{l} [ot \ p] \leftarrow [M_{k,1} \ p+1] \end{array} \right\| \end{array} \right\| \\ \text{if } (p = 0, "no", ot) \end{array} \right\|$$

$$in([2 \ 19], a0) = [[2 \ 19] \ 2] \quad in(77, a0) = "no" \quad in(2, a0) = [2 \ 5]$$

$$inv(a0, a0) = [[\text{"x"} \text{"x"}]] \quad inv([2 \ 19], a0) = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix}$$

$$inv(2, a0)^T = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 0 \\ 0 & 2 \\ 0 & 3 \\ 0 & 0 \\ 1 & 3 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

Решение задач I

Л. Для решения задачи Л1 предлагается рекурсивная функция *re*.

```

re(ma, x, y) :=
  for i ∈ 0..rows(ma)-1
    for j ∈ 0..cols(ma)-1
      if mai,j = x
        mai,j ← y
      else
        if IsArray(mai,j)
          mai,j ← re(mai,j, x, y)
  ma
    
```

$$re(a0, 2, 77) = \begin{bmatrix} 5 & 7 & \begin{bmatrix} 77 & [19 \ 3] \\ 4 & 1 \end{bmatrix} & [77 \ 19] \\ 77 & [77 \ 19] & 5 & \begin{bmatrix} 7 & [77 & [19 \ 3]] \\ 4 & 1 \end{bmatrix} \end{bmatrix}$$

$$re(a0, [2 \ 19], \begin{bmatrix} 2 \\ 19 \end{bmatrix}) = \begin{bmatrix} 5 & 7 & \begin{bmatrix} 2 & [19 \ 3] \\ 4 & 1 \end{bmatrix} & \begin{bmatrix} 2 \\ 19 \end{bmatrix} \\ 2 & \begin{bmatrix} 2 \\ 19 \end{bmatrix} & 5 & \begin{bmatrix} 7 & [2 & [19 \ 3]] \\ 4 & 1 \end{bmatrix} \end{bmatrix}$$

Л2. Данную задачу можно решать функцией *repro(ma, pos, y)*, где: *pos* - это одна позиция в *ma*; *y* - один объект, замещающий элемент *ma* из позиции *pos*.

```

repro(ma, pos, y) :=
  [M n] ← [ma lepos(ma) rows(pos)-1 0 y]
  while Mk,1 ≠ pos
    k ← k+1
  Mk,0 ← y
  while n > 0
    po ← submatrix(pos, 0, n-1, 0, 1)
    while Mk,1 ≠ po
      k ← k-1
    
```


$$\left(\begin{array}{c} \left(M_{k,0} \right)_{pos_{n,0}, pos_{n,1}} \leftarrow y \\ [n \ y] \leftarrow [n-1 \ M_{k,0}] \\ resto(M) \end{array} \right)$$

$$gero(a0, [0 \ 0], 777) = \left[\begin{array}{ccc} 777 & 7 & \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & 1 \end{array} \right] & [2 \ 19] \\ 2 & [2 \ 19] & 5 & \left[\begin{array}{c} 7 \\ \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & 1 \end{array} \right] \end{array} \right] \end{array} \right]$$

$$gero \left(a0, \left[\begin{array}{cc} 1 & 3 \\ 0 & 1 \\ 1 & 1 \end{array} \right], [0] \right) = \left[\begin{array}{ccc} 5 & 7 & \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & 1 \end{array} \right] & [2 \ 19] \\ 2 & [2 \ 19] & 5 & \left[\begin{array}{c} 7 \\ \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & [0] \end{array} \right] \end{array} \right] \end{array} \right]$$

В. Эту задачу можно решать функцией $gerov(ma, vpos, vy)$, построенной по аналогии с функцией $gero$, где: $vpos$ - это вектор позиций в ma ; vy - вектор объектов, замещающих элементы ma из позиций $vpos$.

$$gerov(ma, vpos, vy) := \left(\begin{array}{l} \text{for } m \in 0..rows(vpos) - 1 \\ \quad pos \leftarrow vpos_m \\ \quad yr \leftarrow \text{if}(m \leq rows(vy) - 1, vy_m, vy_{last(vy)}) \\ \quad [M \ n \ k] \leftarrow [ma \ elepos(ma) \ rows(pos) - 1 \ 0] \\ \quad \text{while } M_{k,1} \neq pos \\ \quad \quad k \leftarrow k + 1 \\ \quad \quad M_{k,0} \leftarrow yr \\ \quad \quad \text{while } n > 0 \\ \quad \quad \quad po \leftarrow \text{submatrix}(pos, 0, n - 1, 0, 1) \\ \quad \quad \quad \text{while } M_{k,1} \neq po \\ \quad \quad \quad \quad k \leftarrow k - 1 \\ \quad \quad \quad \quad \left(M_{k,0} \right)_{pos_{n,0}, pos_{n,1}} \leftarrow yr \\ \quad \quad \quad \quad [n \ yr] \leftarrow [n - 1 \ M_{k,0}] \\ \quad \quad ma \leftarrow resto(M) \end{array} \right)$$

$$gerov \left(a0, \left(\left[\begin{array}{cc} 1 & 3 \\ 0 & 1 \\ 1 & 1 \end{array} \right], \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \\ 2 & 0 \\ 0 & 2 \end{array} \right] \right) = \left[\begin{array}{ccc} \left[\begin{array}{cc} 2 & 0 \\ 0 & 2 \end{array} \right] & 7 & \left[\begin{array}{cc} 2 & [2 \ 0] \\ 4 & 1 \end{array} \right] & [2 \ 19] \\ 2 & [2 \ 19] & 5 & \left[\begin{array}{c} 7 \\ \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & [1 \ 0] \\ & [0 \ 1] \end{array} \right] \end{array} \right] \end{array} \right]$$

II. Обобщенный поиск и замещение элементов в гнездовых массивах

Решение задач J. Одно из возможных решений этой задачи представлено ниже логической функцией $equG(ma1, ma2)$, возвращающей 1, если массивы $ma1$ и $ma2$ обобщенно равны, и 0 - в противном случае.

```

equG(a, b) := || if IsArray(a) ∧ IsArray(b)
                || L ← (rows(a) = rows(b)) ∧ (cols(a) = cols(b))
                || for i ∈ 0 .. rows(a) - 1
                ||   || for j ∈ 0 .. cols(b) - 1
                ||     || if L = 1
                ||       || [x y] ← [ai,j bi,j]
                ||       || if IsNaN(x) ∨ IsNaN(y) ∨ (IsNaN(x) ∧ IsNaN(y)) ∨ (x = y)
                ||       ||   || L ← 1
                ||       ||   || else if ¬IsArray(x) ∧ ¬IsArray(y)
                ||       ||     || L ← 0
                ||       ||   || else
                ||       ||     || L ← equG(x, y)
                ||   || else
                ||     || L ← IsNaN(a) ∨ IsNaN(b) ∨ (IsNaN(a) ∧ IsNaN(b)) ∨ (a = b)
                ||   || L
    
```

$$x1 := \begin{bmatrix} 7 \\ \begin{bmatrix} 3 \\ 1 \\ NaN \end{bmatrix} \end{bmatrix} \begin{bmatrix} [19 \ 3] \\ NaN \end{bmatrix} \quad y1 := \begin{bmatrix} 7 \\ \begin{bmatrix} 3 \\ 1 \\ NaN \end{bmatrix} \end{bmatrix} \begin{bmatrix} [19 \ NaN] \\ 1 \end{bmatrix} \quad equG(x1, y1) = 1$$

$$x2 := \begin{bmatrix} 7 \\ \begin{bmatrix} 3 \\ 1 \\ NaN \end{bmatrix} \end{bmatrix} \begin{bmatrix} [19 \ 3] \\ NaN \end{bmatrix} \quad y2 := \begin{bmatrix} 6 \\ \begin{bmatrix} 3 \\ 1 \\ NaN \end{bmatrix} \end{bmatrix} \begin{bmatrix} [19 \ NaN] \\ 1 \end{bmatrix} \quad equG(x2, y2) = 0$$

$$\begin{array}{cccc}
 equG(x1, y1) = 1 & equG(x2, y2) = 0 & equG(7, 5) = 0 & equG(7, x1) = 0 \\
 equG(NaN, 5) = 1 & equG(5, NaN) = 1 & equG(7, 5) = 0 & equG(NaN, NaN) = 1
 \end{array}$$

Решение задач K

K1. Для решения задачи K1 предлагается функция inG .

```

inG(x, ma) := || M ← elepos(ma, [0 0])
                || n ← ∑k=0rows(M)-1 equG(x, Mk,0)
                || if(n = 0, "no", [x n])
    
```

К2. Для решения задачи К2 предлагается функция *invG*.

$$invG(x, ma) := \left\| \begin{array}{l} [M \ p] \leftarrow [elepos(ma, ["x" "x"]) \ 0] \\ \text{for } k \in 0 \dots rows(M) - 1 \\ \quad \left\| \begin{array}{l} \text{if } equG(x, M_{k,0}) \\ \quad \left\| \begin{array}{l} [ot \ p] \leftarrow [M_{k,1} \ p+1] \\ \text{if}(p=0, "no", ot) \end{array} \right. \end{array} \right. \end{array} \right\|$$

К3. Для решения задачи К3 предлагается функция *inmG*.

$$inmG(x, ma) := \left\| \begin{array}{l} [M \ p] \leftarrow [elepos(ma, ["x" "y"]) \ 0] \\ \text{for } k \in 0 \dots rows(M) - 1 \\ \quad \left\| \begin{array}{l} \text{if } equG(x, M_{k,0}) \\ \quad \left\| \begin{array}{l} [ot_{p,0} \ ot_{p,1} \ p] \leftarrow [M_{k,0} \ M_{k,1} \ p+1] \\ \text{if}(p=0, "no", ot) \end{array} \right. \end{array} \right. \end{array} \right\| \quad +$$

$$b0 := \left[\begin{array}{cc} 5 & 7 \\ 2 & [2 \ 19] \end{array} \right] \left[\begin{array}{cc} 2 & [19 \ 3] \\ 4 & 1 \end{array} \right] \left[\begin{array}{cc} [NaN \ 19] \\ 7 & [2 \ 19 \ 3] \\ 4 & 1 \end{array} \right] \quad \begin{array}{l} inG([NaN \ 2], b0) = [[NaN \ 2] \ 1] \\ inG([2 \ NaN], b0) = [[2 \ NaN] \ 3] \\ inG([77 \ 78], b0) = [[77 \ 78] \ 1] \end{array}$$

$$inG(2, b0) = [2 \ 5] \quad invG(2, b0)^T = \left[\begin{array}{cc} [1 \ 0] & \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 3 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \end{array} \right]$$

$$inmG(2, b0)^T = \left[\begin{array}{cc} 2 & 2 \ 2 \ NaN \ 2 \\ [1 \ 0] & \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 3 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \end{array} \right]$$

$$inG(77, b0) = [77 \ 1] \quad invG(77, b0) = \left[\begin{array}{cc} [0 \ 3] \\ 0 & 0 \end{array} \right] \quad inmG(77, b0) = \left[\begin{array}{cc} NaN & [0 \ 3] \\ 0 & 0 \end{array} \right]$$

$$inG([NaN \ 19], b0) = [[NaN \ 19] \ 3] \quad invG([NaN \ 19], b0)^T = \left[\begin{array}{cc} [1 \ 1] & [0 \ 3] \\ 0 & 0 \end{array} \right]$$

$$inmG([NaN \ 19], b0)^T = \left[\begin{array}{cc} [2 \ 19] & [NaN \ 19] \ NaN \\ [1 \ 1] & [0 \ 3] \\ 0 & 0 \end{array} \right]$$

Решение задачи L

L1. Для решения задачи L1 предлагается рекурсивная функция *reG*.

```

reG(ma, x, y) := || if equG(ma, x)
                  || ma ← y
                  || if IsArray(ma)
                  ||   for i ∈ 0..rows(ma) - 1
                  ||     for j ∈ 0..cols(ma) - 1
                  ||       if equG(mai,j, x)
                  ||         mai,j ← y
                  ||       else
                  ||         if IsArray(mai,j)
                  ||           mai,j ← reG(mai,j, x, y)
                  || ma
    
```

$$A := \left[\begin{array}{c} 7 \\ [2 \ 19] \end{array} \left[\begin{array}{c} [NaN \ 19] \\ 7 \left[\begin{array}{c} 2 \\ 4 \end{array} \right] \left[\begin{array}{c} [19 \ 3] \\ [[2 \ NaN]] \end{array} \right] \end{array} \right] \right] \quad reG(A, [2 \ 19], 77) = \left[\begin{array}{c} 7 \\ 77 \end{array} \left[\begin{array}{c} 77 \\ 7 \left[\begin{array}{c} 2 \\ 4 \end{array} \right] \left[\begin{array}{c} [19 \ 3] \\ [[77]] \end{array} \right] \end{array} \right] \right]$$

$$reG(A, [2 \ NaN], \left[\begin{array}{c} 2 \\ 19 \end{array} \right]) = \left[\begin{array}{c} 7 \\ [2] \\ [19] \end{array} \left[\begin{array}{c} 2 \\ 7 \left[\begin{array}{c} 2 \\ 4 \end{array} \right] \left[\begin{array}{c} [19 \ 3] \\ [[2] \\ [19]] \end{array} \right] \end{array} \right] \right] \quad B := [NaN \ 19] \\ reG(B, [2 \ 19], 77) = 77$$

12. Задачу 12 можно решать ранее построенной функцией $repo(ma, pos, y)$, где: pos - одна позиция в ma ; y - один объект, замещающий элемент ma из позиции pos .

$$repo(A, [0 \ 0], 777) = \left[\begin{array}{c} 777 \\ [2 \ 19] \end{array} \left[\begin{array}{c} [NaN \ 19] \\ 7 \left[\begin{array}{c} 2 \\ 4 \end{array} \right] \left[\begin{array}{c} [19 \ 3] \\ [[2 \ NaN]] \end{array} \right] \end{array} \right] \right]$$

$$repo\left(A, \left[\begin{array}{c} 1 \ 1 \\ 0 \ 1 \\ 1 \ 1 \end{array} \right], [0]\right) = \left[\begin{array}{c} 7 \\ [2 \ 19] \end{array} \left[\begin{array}{c} [NaN \ 19] \\ 7 \left[\begin{array}{c} 2 \\ 4 \end{array} \right] \left[\begin{array}{c} [19 \ 3] \\ [0] \end{array} \right] \end{array} \right] \right]$$

13. Задачу 13 можно решать ранее построенной функцией $repor(ma, vpos, vu)$, где: $vpos$ - вектор позиций в ma ; vu - вектор объектов, замещающих элементы ma из позиций $vpos$.

$$repor\left(a0, \left[\begin{array}{c} [1 \ 3] \\ [0 \ 1] \\ [1 \ 1] \\ [0 \ 0] \\ [0 \ 2] \\ [0 \ 1] \end{array} \right], \left[\begin{array}{c} [1 \ 0] \\ [0 \ 1] \\ [2 \ 0] \\ [0 \ 2] \end{array} \right]\right) = \left[\begin{array}{c} [2 \ 0] \\ [0 \ 2] \end{array} \right] \quad 7 \quad \left[\begin{array}{c} 2 \\ 4 \end{array} \right] \left[\begin{array}{c} [2 \ 0] \\ [0 \ 2] \\ 1 \end{array} \right] \quad [2 \ 19] \\ 2 \quad [2 \ 19] \quad 5 \quad \left[\begin{array}{c} 7 \\ 4 \end{array} \right] \left[\begin{array}{c} 2 \\ [19 \ 3] \\ [1 \ 0] \\ [0 \ 1] \end{array} \right] \end{array} \right]$$

Функции для решения задач F-L. В этом пункте на восьми фрагментах *mathcad*-документа представлены функции для решения перечисленных ранее задач на простой и обобщенный поиск и замещение. Соответствие имен задач и функций таково: *F1 — elepos, maelepos; F2 — resto; G1 — pick, pickv; G2 — pickvr; H1 — in, H2 — inv; I1 — re; I2 — repo, I3 — repov; J — equG, K1 — inG, K2 — invG, K3 — inmG, L1 — reG, L2 — repo, L3 — repov.*

ЛИТЕРАТУРА

- [1] Григорьев С.Г., Есаян А.Р. Простой и обобщенный поиск элементов в гнездовых массивах и их замещение // Чебышевский сборник. 2015. Т. XVI. Вып. 3 (55). С. 499—513.
- [2] Есаян А.Р. Обучение алгоритмизации на основе рекурсии: учебное пособие для студентов педвузов. Тула: ТГПУ им. Л.Н. Толстого, 2001. 215 с.
- [3] Есаян А.Р., Добровольский Н.М. Гнездовые массивы и рекурсия // Алгебра, теория чисел и дискретная геометрия. Современные проблемы и приложения: Материалы XXIII международной конференции. Тула, 2015. С. 319—321.
- [4] Есаян А.Р., Якушин А.В. Векторизация и гнездовые массивы // Алгебра, теория чисел и дискретная геометрия. Современные проблемы и приложения: Материалы XXIII международной конференции. Тула, 2015. С. 328—330.
- [5] Есаян А.Р., Добровольский Н.М. Гнездовые массивы и рекурсия // Чебышевский сборник. 2015. Том XVI. Выпуск 3 (55). С. 481—498.
- [6] Есаян А.Р., Якушин А.В. Векторизация и гнездовые массивы // Чебышевский сборник. 2015. Том XVI. Вып. 3 (55). С. 499—513.
- [7] Maxfield B., Essential P.E. PTC Mathcad Prime 3.0. A Guide for New and Current Users. New York, Academic Press is an imprint of Elsevier. 2013. No. 11. P. 563.
- [8] Wesseningh H., Waard H. Calculate & Communicate with Mathcad Prime 3.0, Delft Academic Press, The Netherlands, First edition 2014.

SEARCH AND SUBSTITUTION IN NESTED ARRAYS

S.G. Grigoriev¹, A.R. Esayan²

¹ Moscow city pedagogical university
Sheremetyevskaya str., 29, Moscow, Russia, 127521

² Tula state pedagogical university of L.N. Tolstoy
Prospekt Lenina, 125, Tula, Russia, 300026

In this article simple algorithms of the solution of recursive algorithms for the simple and generalized search and replacement of elements are given in nested matrixes. The new tasks connected with nested matrixes with numerical or line elements are stated. For record of algorithms language of the environment of engineering and scientific calculations PTC Mathcad Prime is used. In the course of their decision at students the logical, algorithmic information thinking develops.

Key words: informatics, nested matrixes, computing tasks, information technologies, training informatization

REFERENCES

- [1] Grigor'ev S.G., Esajan A.R. *Prostoj i obobshhennyj poisk jelementov v gnezdovyh massivah i ih zameshhenie* [Idle time and the generalized search of elements in nested massifs and their replacement]. *Chebyshevskij sbornik* [Chebyshevskij sbornik]. 2015. T. XVI. No 3 (55). Pp. 499—513.
- [2] Esajan A.R. *Obuchenie algoritimizacii na osnove rekursii* [Training of algorithmization on the basis of a recursion]: *uchebnoe posobie dlja studentov pedvuzov*. Tula: TGPU im. L.N. Tolstogo, 2001. 215 p.
- [3] Esajan A.R., Dobrovol'skij N.M. *Gnezdovye massivы i rekursija* [Nested massifs and recursion]. *Algebra, teorija chisel i diskretnaja geometrija. Sovremennye problemy i prilozhenija* [Algebra, theory of numbers and discrete geometry. Modern problems and applications]: *Materialy XXIII mezhdunarodnoj konferencii*. Tula, 2015. Pp. 319—321.
- [4] Esajan A.R., Jakushin A.V. *Vektorizacija i gnezdovye massivы* [Vectorization and nested massifs]. *Algebra, teorija chisel i diskretnaja geometrija. Sovremennye problemy i prilozhenija* [Algebra, theory of numbers and discrete geometry. Modern problems and applications]: *Materialy XXIII mezhdunarodnoj konferencii*. Tula, 2015. Pp. 328—330.
- [5] Esajan A.R., Dobrovol'skij N.M. *Gnezdovye massivы i rekursija* [Nested massifs and recursion]. *Chebyshevskij sbornik* [Chebyshevskij sbornik]. 2015. T. XVI. No 3 (55). Pp. 481—498.
- [6] Esajan A.R., Jakushin A.V. *Vektorizacija i gnezdovye massivы* [Vectorization and nested massifs]. *Chebyshevskij sbornik* [Chebyshevskij sbornik]. Tula, 2015. T. XVI. No 3 (55). Pp. 499—513.
- [7] Maxfield B., Essential P.E. *PTC Mathcad Prime 3.0. A Guide for New and Current Users*. New York, Academic Press is an imprint of Elsevier. 2013. No. 11. P. 563.
- [8] Wessenlingh N., Waard H. *Calculate & Communicate with Mathcad Prime 3.0*, Delft Academic Press, The Netherlands, First edition 2014.