

**Федеральное государственное автономное образовательное учреждение
высшего образования
"Национальный исследовательский университет "Высшая школа экономики"**

Московский институт электроники и математики им. А.Н. Тихонова НИУ ВШЭ
Департамент компьютерной инженерии

Курс: Алгоритмизация и программирование

Раздел	Макс. оценка	Итоговая оценка
Постановка	0.5	
Метод	1	
Спецификация	0.5	
Алгоритм	1.5	
Работа программы	1	
Листинг	0.5	
Тесты	1	
Вопросы	2	
Доп. задание	2	

Отчет по лабораторной работе № 2

Студент: Чапайкин Арсений Георгиевич

Группа: БИВ242

Вариант: № 171 (1, 1)

Руководитель: Елисеенко А.М.

Оценка:

Дата сдачи:

Содержание

Задание	2
Постановка задачи	3
Метод решения	4
Внешняя спецификация	5
Описание алгоритма на псевдокоде	6
Листинг программы	9
Распечатка тестов к программе и результатов	12

Задание

1. Даны целочисленная матрица $Z[1:n \ 1:m]$ и целочисленный массив $X[1:k]$. В каждой строке матрицы Z обнулить максимальный по абсолютной величине элемент, если его нет в массиве X .
2. Дан массив $X[1:k]$ целых положительных чисел. Сформировать новый массив, содержащий суммы цифр каждого элемента исходного массива.

Постановка задачи

Дано:

1. $n, m, k \in \mathbb{N}$. $Z[i][j], X[i] \in \mathbb{Z}$.
2. $k \in \mathbb{N}$. $X[i] \in \mathbb{Z}$.

Результат:

1. Матрица $Z[1:n \ 1:m]$, $Z[i][j] \in \mathbb{Z}$.
2. Массив $X[1:k]$, $X[i] \in \mathbb{Z}^+$.

При:

1. $n, m, k \in \mathbb{N}$. $Z[i][j], X[i] \in \mathbb{Z}$. $n, m, k \leq \text{MaxSize}$.
2. $k \in \mathbb{N}$. $X[i] \in \mathbb{Z}$. $k \leq \text{MaxSize}$.

Связь:

1. $\forall i = \overline{1 \dots n}$ требуется $\forall j = \overline{1 \dots m} \ \forall k = \overline{1 \dots m} : |Z[i][j]| \geq |Z[i][k]|$ присвоить $Z[i][j] := 0$.
2. $\forall i = \overline{1 \dots n}$ требуется присвоить $X[i] := \sum_{k=0}^{\infty} \left(\left[\frac{X[i]}{10^k} \right] \% 10 \right)$.

Метод решения

$$1. \left\{ \begin{array}{l} \text{Для } i = \overline{1 \dots n} \\ idx = 0 \\ \left\{ \begin{array}{l} \text{Для } j = \overline{1 \dots m} \\ (idx = j), \text{ если } |Z[i][j]| > |Z[i][idx]| \end{array} \right. \\ \left\{ \begin{array}{l} \text{Для } j = \overline{1 \dots m} \\ (Z[i][j] = 0), \text{ если } |Z[i][j]| = |Z[i][idx]| \end{array} \right. \end{array} \right.$$

$$2. \left\{ \begin{array}{l} \text{Для } i = \overline{1 \dots n} \\ number = X[i] \\ sum = 0 \\ \left\{ \begin{array}{l} \text{Пока } number > 0 \\ sum = sum + number \% 10 \\ number = number // 10 \end{array} \right. \\ X[i] = sum \end{array} \right.$$

Внешняя спецификация

Задание 1

Input matrix dimensions in the following format:

'number of rows' 'number of columns'

Both variables must be positive integers.

$\left\{ \begin{array}{l} \langle n \rangle \langle m \rangle \\ \text{Input format is incorrect. Try again.} \end{array} \right\}^* \quad \text{До } n, m \in \mathbb{N} \wedge 0 < n \leq \text{MaxSize} \wedge 0 < m \leq \text{MaxSize}$

Input matrix in the following format:

Z[1][1] Z[1][2] ... Z[n][m]

$\left\{ \begin{array}{l} \langle Z[1][1] \rangle \langle Z[1][2] \rangle \dots \langle Z[n][m] \rangle \\ \text{Input format is incorrect. Try again.} \end{array} \right\}^* \quad \text{До } \forall i = \overline{1 \dots n} \forall j = \overline{1 \dots m} Z[i][j] \in \mathbb{Z}$

Input array size (positive integer).

$\left\{ \begin{array}{l} \langle k \rangle \\ \text{Input format is incorrect. Try again.} \end{array} \right\}^* \quad \text{До } k \in \mathbb{N} \wedge 0 < k \leq \text{MaxSize}$

Input array in the following format:

X[1] X[2] ... X[k]

$\left\{ \begin{array}{l} \langle X[1] \rangle \langle X[2] \rangle \dots \langle X[k] \rangle \\ \text{Input format is incorrect. Try again.} \end{array} \right\}^* \quad \text{До } \forall i = \overline{1 \dots k} X[i] \in \mathbb{Z}$

Result:

$\begin{array}{ccc} \langle Z[1][1] \rangle & \dots & \langle Z[1][m] \rangle \\ \vdots & \ddots & \vdots \\ \langle Z[n][1] \rangle & \dots & \langle Z[n][m] \rangle \end{array}$

Задача 2

Input array size (positive integer).

$\left\{ \begin{array}{l} \langle k \rangle \\ \text{Input format is incorrect. Try again.} \end{array} \right\}^* \quad \text{До } k \in \mathbb{N} \wedge 0 < k \leq \text{MaxSize}$

Input array in the following format:

X[1] X[2] ... X[k]

$\left\{ \begin{array}{l} \langle X[1] \rangle \langle X[2] \rangle \dots \langle X[k] \rangle \\ \text{Input format is incorrect. Try again.} \end{array} \right\}^* \quad \text{До } \forall i = \overline{1 \dots k} X[i] \in \mathbb{Z}$

Result:

$\langle X[1] \rangle \dots \langle X[k] \rangle$

Описание алгоритма на псевдокоде

Задание 1

Константа $\text{MaxSize} = 100$

Объявить переменные:

rows, cols, sizeOfArray: целое число
matrix: массив размером $\text{MaxSize} * \text{MaxSize}$
array: массив размером MaxSize

функция find(element: целое число):

цикл i := 0 до sizeOfArray - 1:
 если array[i] = element то
 вернуть true
 все
кц
вернуть false

кц

процедура inputArray(size: целое число, array: массив целых чисел):

correctInput: булево
цикл:
 correctInput = true
 цикл i := 0 до size - 1:
 ввод(array[i])
 если array[i] не является целым числом то
 correctInput = false
 все
 кц
 если !correctInput то
 вывод("Неверный формат ввода. Попробуйте снова.")
 все
до correctInput
кц

кц

функция validSize(size: целое число):

вернуть size > 0 и size <= MaxSize

кц

Нач

вывод("Введите размеры матрицы в следующем формате:")
вывод("число строк' 'число столбцов'")
вывод("Оба числа должны быть положительными целыми")

цикл-пока ввод(rows) не целое или ввод(cols) не целое или !validSize(rows) или !validSize(cols):
 вывод("Неверный формат ввода. Попробуйте снова.")

кц

```
вывод("Введите матрицу в следующем формате:")
вывод("matrix[0][0] matrix[0][1] ... matrix[1][1] ... matrix[rows][cols]")
inputArray(rows * cols, matrix)
```

```
вывод("Введите размер массива (положительное целое число).")
```

```
цикл-пока ввод(sizeOfArray) не целое или !validSize(sizeOfArray):
    вывод("Неверный формат ввода. Попробуйте снова.")
```

```
кц
```

```
вывод("Введите массив в следующем формате:")
вывод("array[0] array[1] ... array[sizeOfArray]")
inputArray(sizeOfArray, array)
```

```
цикл i := 0 до rows - 1:
    maximumIndex = 0
    цикл j := 0 до cols - 1:
        если abs((matrix[i * cols + j])) > abs((matrix[i * cols + maximumIndex])) то
            maximumIndex = j
        все
    кц
    цикл j := 0 до cols - 1:
        если abs((matrix[i * cols + j])) = abs((matrix[i * cols + maximumIndex])) то
            matrix[i * cols + j] = 0
        все
    кц
кц
```

```
вывод("Результат:")
цикл i := 0 до rows - 1:
    цикл j := 0 до cols - 1:
        вывод(matrix[i * cols + j])
    кц
кц
```

Кон

Задание 2

Константа MaxSize = 100

Объявить переменные:

sizeOfArray: целое число
array: массив размером MaxSize

функция sumOfDigits(number: целое число):

```
sum = 0
цикл-пока number > 0:
    sum += number % 10
```



```

        number /= 10
    кц
    вернуть sum
кц

процедура inputArray(size: целое число, array: массив целых чисел):
    correctInput: булево
    цикл:
        correctInput = true
        цикл i := 0 до size - 1:
            ввод(array[i])
            если array[i] не является целым числом то
                correctInput = false
            всё
        кц
        если !correctInput то
            вывод("Неверный формат ввода. Попробуйте снова.")
        всё
    до correctInput
    кц
кц

```

```

функция validSize(size: целое число):
    вернуть size > 0 и size <= MaxSize
кц

```

```

Нач:
    вывод("Введите размер массива (положительное целое число).")

    цикл-пока ввод(sizeOfArray) не целое или !validSize(sizeOfArray):
        вывод("Неверный формат ввода. Попробуйте снова.")
    кц

    вывод("Введите элементы массива (целые числа) в следующем формате:")
    вывод("array[0] ... array[sizeOfArray]")
    inputArray(sizeOfArray, array)

    цикл i := 0 до sizeOfArray - 1:
        array[i] = sumOfDigits(array[i])
    кц

    вывод("Результат:")
    цикл i := 0 до sizeOfArray - 1:
        вывод(array[i])
    кц

```

Кон

Листинг программы

Задание 1

```
1 #include <stdio.h>
2 #include <stdbool.h>
3 #include <math.h>
4
5 const int MaxSize = 100;
6
7 size_t rows = 0, cols = 0, size_of_array = 0;
8 int matrix[100 * 100], array[100];
9
10 bool find(int element, int size_of_array, int *array) {
11     for (size_t i = 0; i < size_of_array; i++) {
12         if (array[i] == element) {
13             return true;
14         }
15     }
16     return false;
17 }
18
19 void input_array(size_t size, int* array) {
20     bool correct_input;
21     do {
22         correct_input = true;
23         for (size_t i = 0; i < size; i++) {
24             if (scanf("%d", &array[i]) != 1) {
25                 correct_input = false;
26             }
27         }
28         if (!correct_input) {
29             ("Wrong input format. Try again.");
30         }
31     } while (!correct_input);
32 }
33
34 bool valid_size(size_t size) {
35     return 0 < size && size <= MaxSize;
36 }
37
38 int main() {
39     // Input section
40     puts("Input matrix dimensions in the following format:");
41     puts("'number of rows' 'number of columns'");
42     puts("Both numbers must be positive integers.");
43
44     while (scanf("%zu%zu", &rows, &cols) != 2 || !valid_size(rows) || !valid_size(
45         cols)) {
46         puts("Wrong input format. Try again.");
47     }
48
49     puts("Input matrix in the following format:");
50     puts("matrix[0][0] matrix[0][1] ... matrix[1][1] ... matrix[rows][cols]");
51     input_array(rows * cols, matrix);
52
53     puts("Input array size (positive integer).");
54     while (scanf("%d", &size_of_array) != 1 || !valid_size(size_of_array)) {
55         puts("Wrong input format. Try again.");
56     }
```

```

55     }
56
57     puts("Input array in the following format:");
58     puts("array[0] array[1] ... array[size_of_array]");
59     input_array(size_of_array, array);
60
61     // Main section
62     for (size_t i = 0; i < rows; i++) {
63         size_t maximum_index = 0;
64         for (size_t j = 0; j < cols; j++) {
65             if (abs(matrix[i * cols + j]) > abs(matrix[i * cols + maximum_index]))
66                 {
67                     maximum_index = j;
68                 }
69         }
70         for (size_t j = 0; j < cols; j++) {
71             if (abs(matrix[i * cols + j]) == abs(matrix[i * cols + maximum_index]))
72                 {
73                     matrix[i * cols + j] = 0;
74                 }
75         }
76
77         // Output section
78         puts("Result:");
79         for (size_t i = 0; i < rows; i++) {
80             for (size_t j = 0; j < cols; j++) {
81                 printf("%d ", matrix[i * cols + j]);
82             }
83             printf("\n");
84         }
85
86         return 0;
87     }

```

Задание 2

```

1  #include <stdio.h>
2  #include <stdbool.h>
3  #include <math.h>
4
5  const int MaxSize = 100;
6
7  size_t size_of_array = 0;
8  int array[100];
9
10 int sum_of_digits(int number) {
11     int sum = 0;
12     while (number > 0) {
13         sum += number % 10;
14         number /= 10;
15     }
16     return sum;
17 }
18
19 void input_array(size_t size, int* array) {
20     bool correct_input;
21     do {

```

```

22     correct_input = true;
23     for (size_t i = 0; i < size; i++) {
24         if (scanf("%d", &array[i]) != 1) {
25             correct_input = false;
26         }
27     }
28     if (!correct_input) {
29         ("Wrong input format. Try again.");
30     }
31 } while (!correct_input);
32 }
33
34 bool valid_size(size_t size) {
35     return 0 < size && size <= MaxSize;
36 }
37
38 int main() {
39     // Input section
40     puts("Input array size (positive integer).");
41
42     while (scanf("%zu", &size_of_array) != 1 || !valid_size(size_of_array)) {
43         puts("Wrong input format. Try again");
44     }
45
46     puts("Input array elements (integers) in the following format:");
47     puts("array[0] ... array[size_of_array]");
48     input_array(size_of_array, array);
49
50     // Main section
51     for (size_t i = 0; i < size_of_array; i++) {
52         array[i] = sum_of_digits(abs(array[i]));
53     }
54
55     // Output section
56     puts("Result:");
57     for (size_t i = 0; i < size_of_array; i++) {
58         printf("%d ", array[i]);
59     }
60     return 0;
61 }

```

Распечатка тестов к программе и результатов

Задание 1

Номер	Исходные данные	Результат
1	n = 3, m = 3, k = 3 Z = 1 2 3 4 5 6 7 8 9 X = 10 11 12	1 2 0 4 5 0 7 8 0
2	n = 3, m = 3, k = 3 Z = 1 2 3 4 5 6 7 8 9 X = 3 6 9	1 2 3 4 5 6 7 8 9
3	n = 3, m = 3, k = 3 Z = -3 2 3 -6 5 6 -9 8 9 X = 10 11 12	0 2 0 0 5 0 0 8 0
4	n = 3, m = 3, k = 3 Z = -3 3 3 -6 6 6 -9 9 9 X = 10 11 12	0 0 0 0 0 0 0 0 0
5	n = 3, m = 3, k = 2 Z = -3 3 3 -6 6 6 -9 9 9 X = -3 9	-3 0 0 0 0 0 0 9 9

Задание 2

Номер	Исходные данные	Результат
1	k = 5 X = 0 0 0 0 0	0 0 0 0 0
2	k = 5 X = 1 2 3 4 5	1 2 3 4 5
3	k = 5 X = -1 -2 -3 -4 -5	1 2 3 4 5
4	k = 5 X = 50 49 17 343 1234	15 13 8 10 10
5	k = 5 X = -1024 -66 -9999 -0 -70	7 12 36 0 7