|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **БИОМЕДИЦИНСКАЯ ТЕХНИКА**

КАФЕДРА **БИОМЕДИЦИНСКИЕ ТЕХНИЧЕСКИЕ СИСТЕМЫ (БМТ-1)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.03 Прикладная информатика (Цифровые биомедицинские системы)**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 3 |

**Название:**

Функциональная и модульная декомпозиция

**Дисциплина:** Алгоритмизация и программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | БМТ1-13Б |  |  | Н.А.Сухов |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Т.А.Ким |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

# Задание.Функции и модули

Дана целочисленная матрица размера B(n,m),(n,m ≤ 10). Преобразовать ее, заменив в каждой

строке диагональный элемент на максимальный элемент этой строки.

# Исходный код

* MainModule.jl

module Main

include("FunctionsModule.jl")

using .MatrixFunctions

using Test

print("Enter n and m: \n")

n,m = parse(Int64, readline()), parse(Int64, readline())

before\_cng = create\_matrix(n, m)

mat\_out(before\_cng)

before\_cng = d\_element\_to\_max(before\_cng)

mat\_out(before\_cng)

@testset "FunctionsTest" begin

  @testset "FindMaxTest" begin

    @test max\_in\_row([[1, 2], [3, 4], [5, 6]], 1) == 5

    @test max\_in\_row([[11, 21, 31, 41], [12, 22, 32, 42], [13, 23, 33, 43]], 4) == 43

  end

  @testset "FinalMatrix" begin

    @test d\_element\_to\_max([[1, 2, 3], [4, 5, 6], [7, 8, 9]]) == [[7, 2, 9], [4, 8, 6], [7, 8, 9]]

    @test d\_element\_to\_max([[11, 21, 31, 41, 51], [12, 22, 32, 42, 52], [13, 23, 33, 43, 53], [14, 24, 34, 44, 54]]) == [[14, 21, 31, 44, 51], [12, 24, 34, 42, 52], [13, 24, 34, 43, 53], [14, 24, 34, 44, 54]]

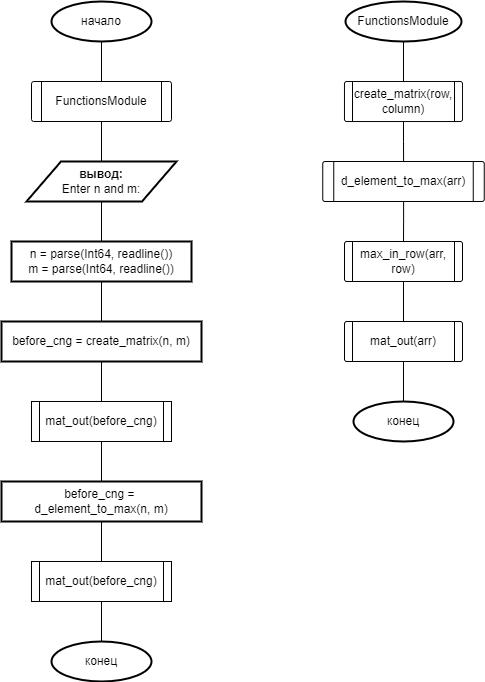
  end

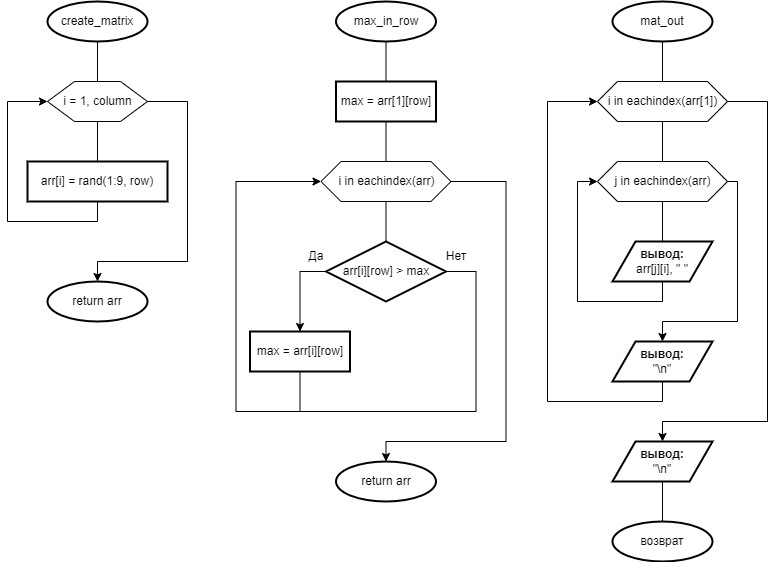
end

end

* FunctionsModule.jl
* module MatrixFunctions
* export create\_matrix, d\_element\_to\_max, max\_in\_row, mat\_out
* function create\_matrix(row::Int, column::Int)
* arr = [rand(1:9, row) for i in 1:column]
* return arr
* end
* function d\_element\_to\_max(arr::Array)
* for i in eachindex(arr[1])
* max = max\_in\_row(arr, i)
* for j in eachindex(arr)
* if i == j || ((i + j - 1) == size(arr)[1])
* arr[j][i] = max
* end
* end
* end
* return arr
* end
* function max\_in\_row(arr::Array, row::Int)
* max = arr[1][row]
* for i in eachindex(arr)
* if arr[i][row] > max
* max = arr[i][row]
* end
* end
* return max
* end
* function mat\_out(arr::Array)
* for i in eachindex(arr[1])
* for j in eachindex(arr)
* print(arr[j][i], " ")
* end
* print("\n")
* end
* print("\n")
* end
* end

# Схема алгоритма





# 

# Тестирование алгоритма

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Наименование проверки** | **Данные на вход** | **Ожидаемый результат** | **Полученный результат** | **Вывод** |
| Проверка внутренней функции нахождения максимального элемента в строке | Массив:  [[1, 2],  [3, 4],  [5, 6]]  Строка: 1 | 5 | 5 | Подфункция отработала в соответствии с ожиданиями. |
| Массив:  [[11, 21, 31, 41],  [12, 22, 32, 42], [13, 23, 33, 43]]  Строка: 4 | 43 | 43 |
| Проверка основной функции, изменяющей массив, используя подфункцию: |  |  |  | Основная функция отработала в соответствии с ожиданиями: диагонали заменялись как в квадратных матрицах, так и прямоугольных матрицах. |
| (квадратная матрица) | Массив:  [[1, 2, 3],  [4, 5, 6],  [7, 8, 9]] | [[7, 2, 9],  [4, 8, 6],  [7, 8, 9]] | [[7, 2, 9],  [4, 8, 6],  [7, 8, 9]] |
| 2)Прямоугольная матрица | Массив:  [[11, 21, 31, 41, 51], [12, 22, 32, 42, 52], [13, 23, 33, 43, 53], [14, 24, 34, 44, 54]] | [[14, 21, 31, 44, 51], [12, 24, 34, 42, 52], [13, 24, 34, 43, 53], [14, 24, 34, 44, 54]] | [[14, 21, 31, 44, 51], [12, 24, 34, 42, 52], [13, 24, 34, 43, 53], [14, 24, 34, 44, 54]] |

# Выводы

В ходе выполнения лабораторной работы я познакомился с пакетами, модулями и научился применять их для структурирования кода и разбиения его на блоки и подблоки, что позволяет использовать его рациональнее, а также внедрил в свой проект систему автоматизированного тестирования для облегчения процесса проверки.