Universidad Nacional Autónoma de México

Facultad de estudios Superiores Aragón

Ingeniería en Computación

Área: Ciencias de la Computación

Materia: Estructura de Datos

Profesor: Roberto Blanco Bautista

Título: Matriz Dispersa (Sparse Matrix)

Alumno: Hernández Gorgonio Henry

Grupo: 2307

Fecha: 10/04/2022

Sparse Matrix

"In numerical análisis and scientific computing, a sparse matrix or sparse array is a matrix in which most of the elements are zero".

$$\begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 6 & 0 & 0 \end{pmatrix}$$

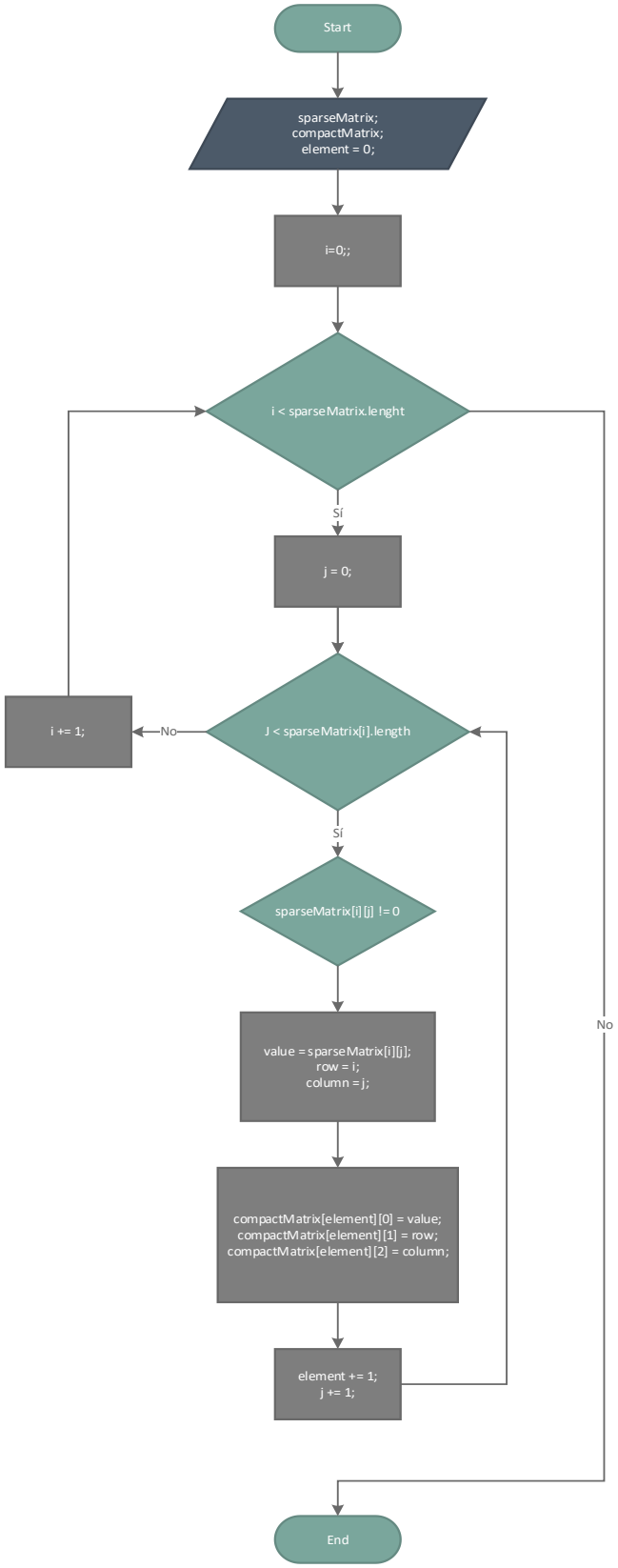
The matrix above is a sparse matrix where most of its elements are 0.

In computer science or programming, we usually represent matrices using bidimensional arrays (a[i][j], where i = row and j=column), in most of the cases that approach works fine, but in the particular case of sparse matrices we can use a different approach saving memory resources.

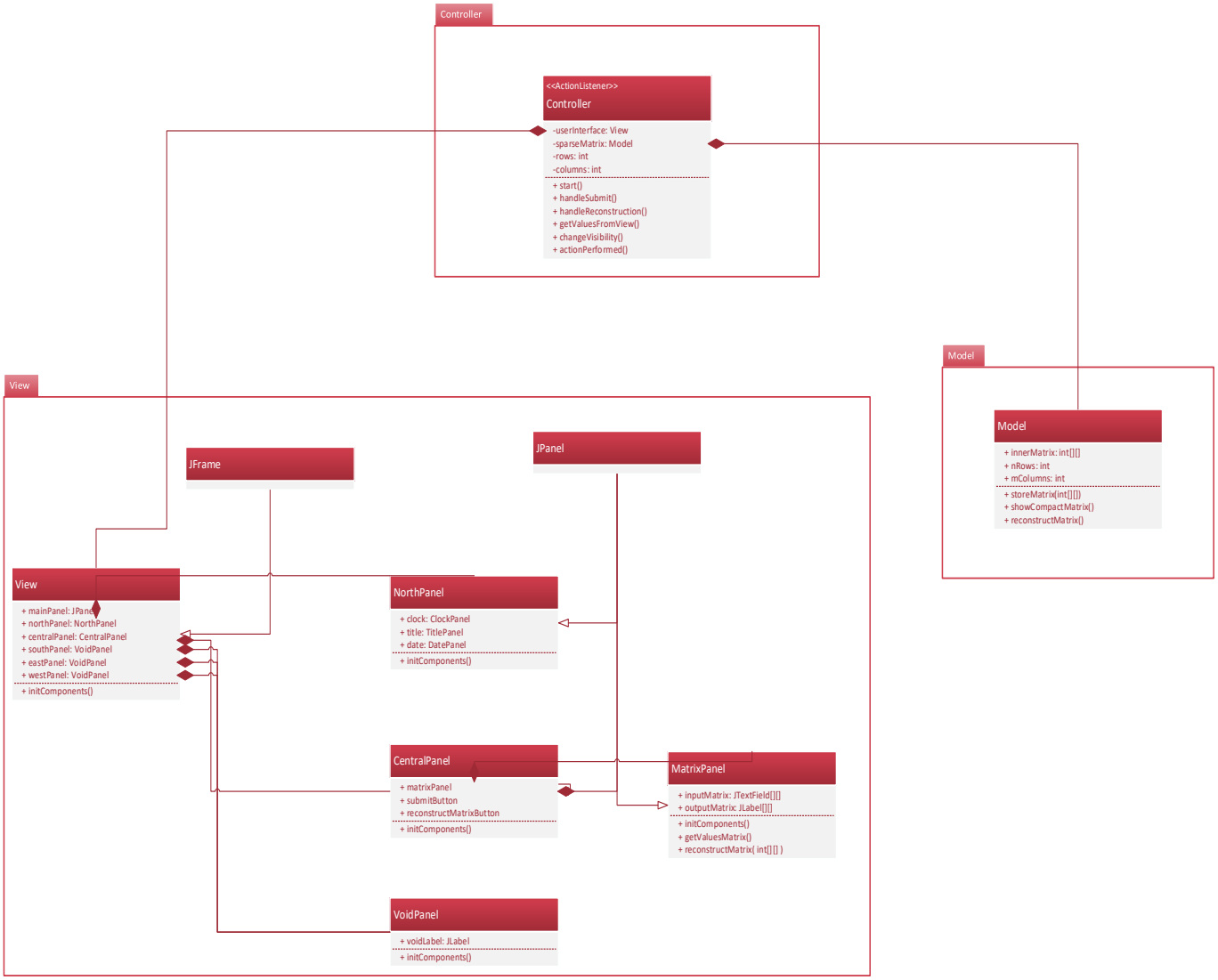
The approach consists in keeping only the non-zero values in the matrix, and the matrix position data. We can describe the process as:

1) Iterate through the matrix counting the number of elements distinct to zero.
2) Generate a bidimensional array [i][3], where i = the number of elements distinct to zero.
3) Iterate through the matrix taking the elements distinct to zero, the column and the row it belongs to, and insert that data in the new array such that:
    a. [i][0] = the value
    b. [i][1] = the row
    c. [i][2] = the column
4) We can reconstruct the original matrix by applying the same process reversed.

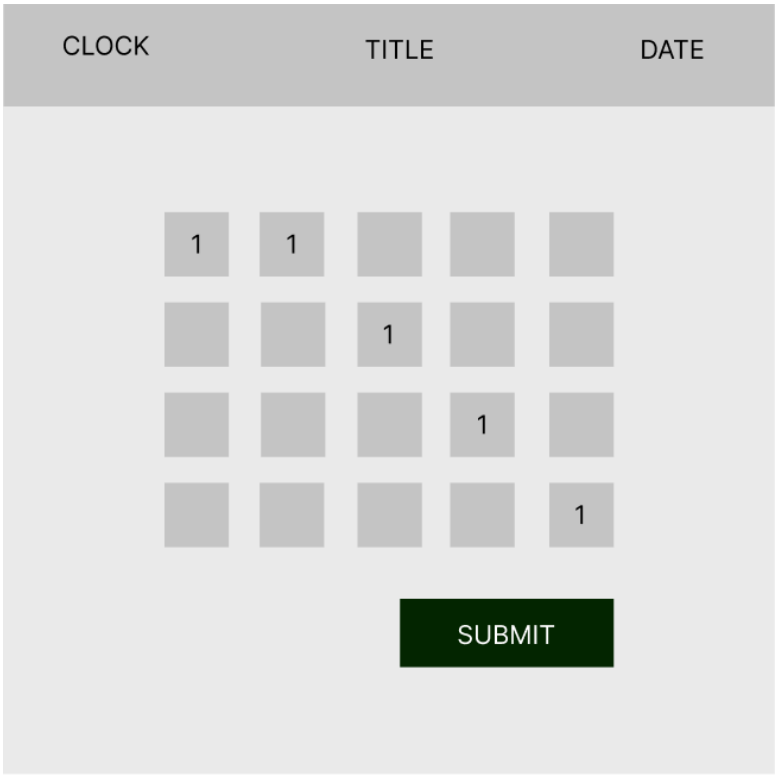
Flowchart


Start
sparseMatrix;
compactMatrix;
element = 0;
i=0;;
i < sparseMatrix.lenght
Sí
j = 0;
J < sparseMatrix[i].length
No
i += 1;
Sí
sparseMatrix[i][j] != 0
value = sparseMatrix[i][j];
row = i;
column = j;
compactMatrix[element][0] = value;
compactMatrix[element][1] = row;
compactMatrix[element][2] = column;
element += 1;
j += 1;
No
End

UML Diagram


Controller
<<ActionListener>>
Controller
-userInterface: View
-sparseMatrix: Model
-rows: int
-columns: int
+ start()
+ handleSubmit()
+ handleReconstruction()
+ getValuesFromView()
+ changeVisibility()
+ actionPerformed()
Model
Model
+ innerMatrix: int[][]
+ nRows: int
+ mColumns: int
+ storeMatrix(int[][])
+ showCompactMatrix()
+ reconstructMatrix()
View
JFrame
JPanel
View
+ mainPanel: JPanel
+ northPanel: NorthPanel
+ centralPanel: CentralPanel
+ southPanel: VoidPanel
+ eastPanel: VoidPanel
+ westPanel: VoidPanel
+ initComponents()
NorthPanel
+ clock: ClockPanel
+ title: TitlePanel
+ date: DatePanel
+ initComponents()
CentralPanel
+ matrixPanel
+ submitButton
+ reconstructMatrixButton
+ initComponents()
MatrixPanel
+ inputMatrix: JTextField[][]
+ outputMatrix: JLabel[][]
+ initComponents()
+ getValuesMatrix()
+ reconstructMatrix(int[][])
VoidPanel
+ voidLabel: JLabel
+ initComponents()

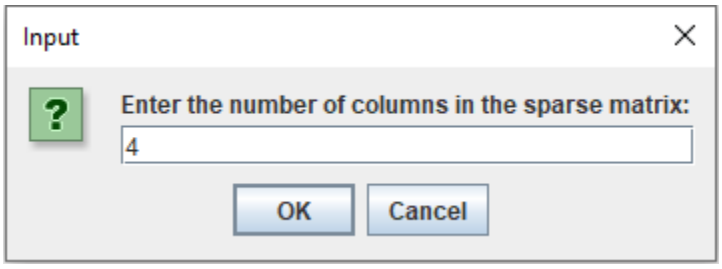
User Interface

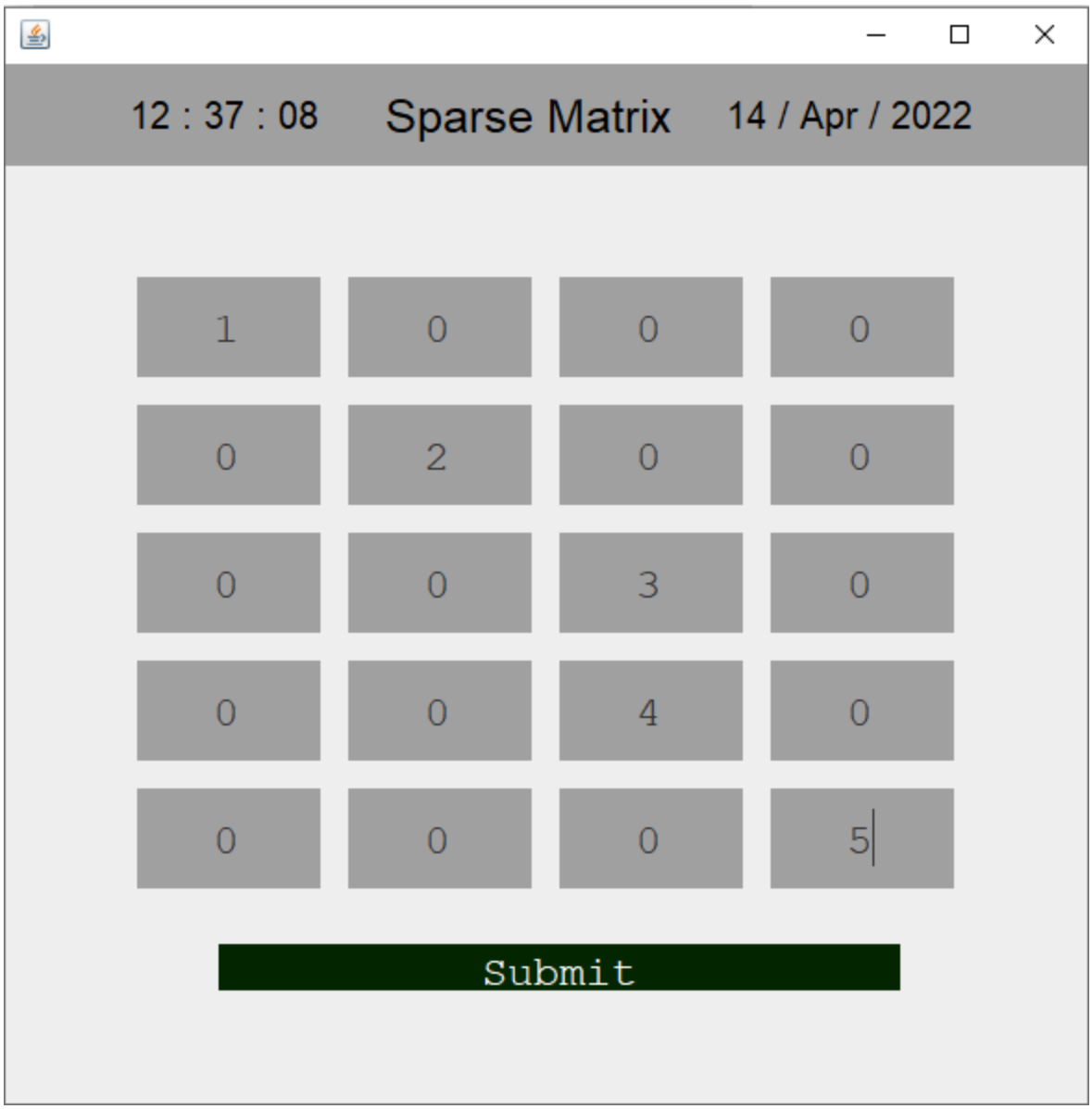


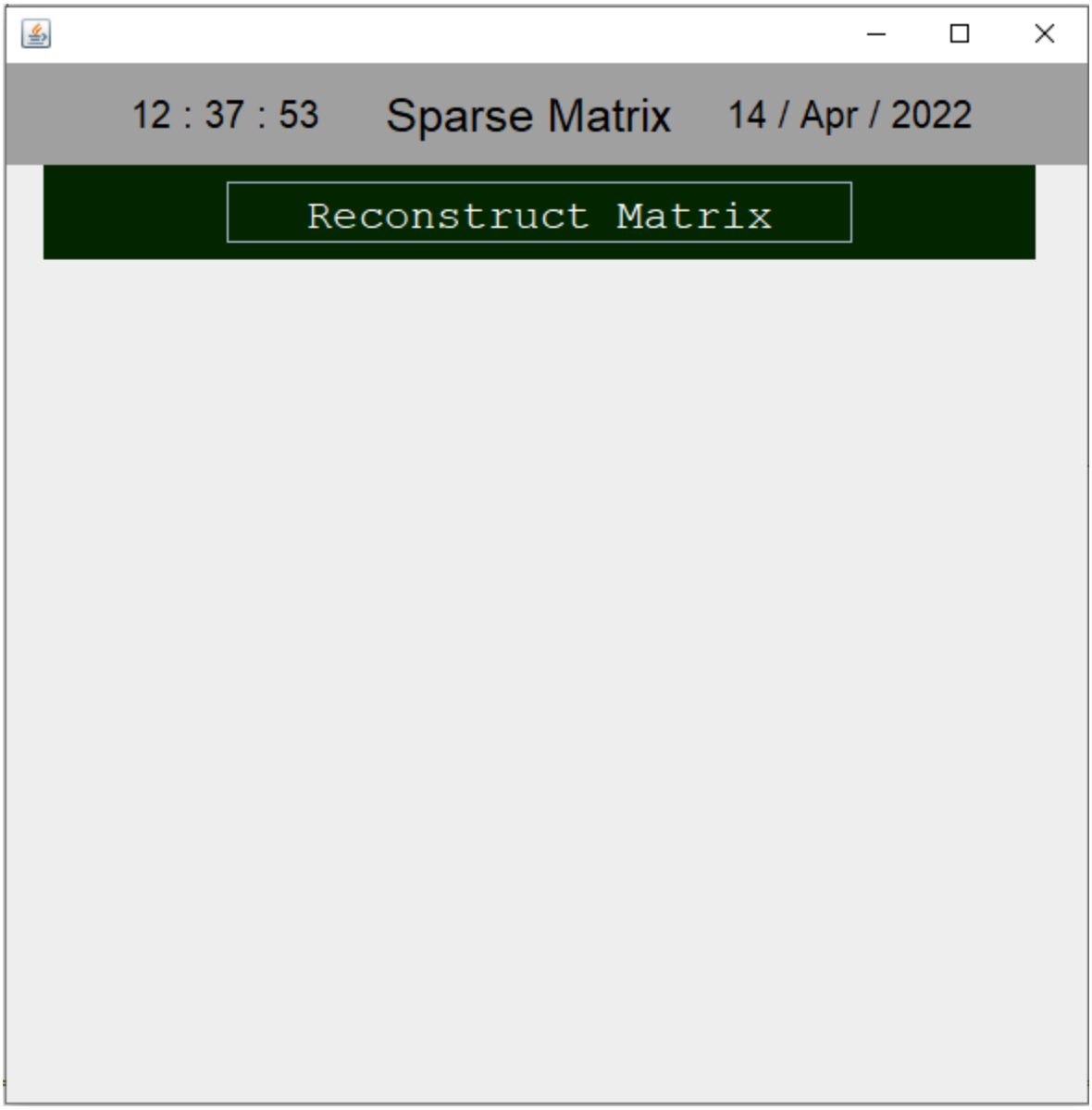
Codebase

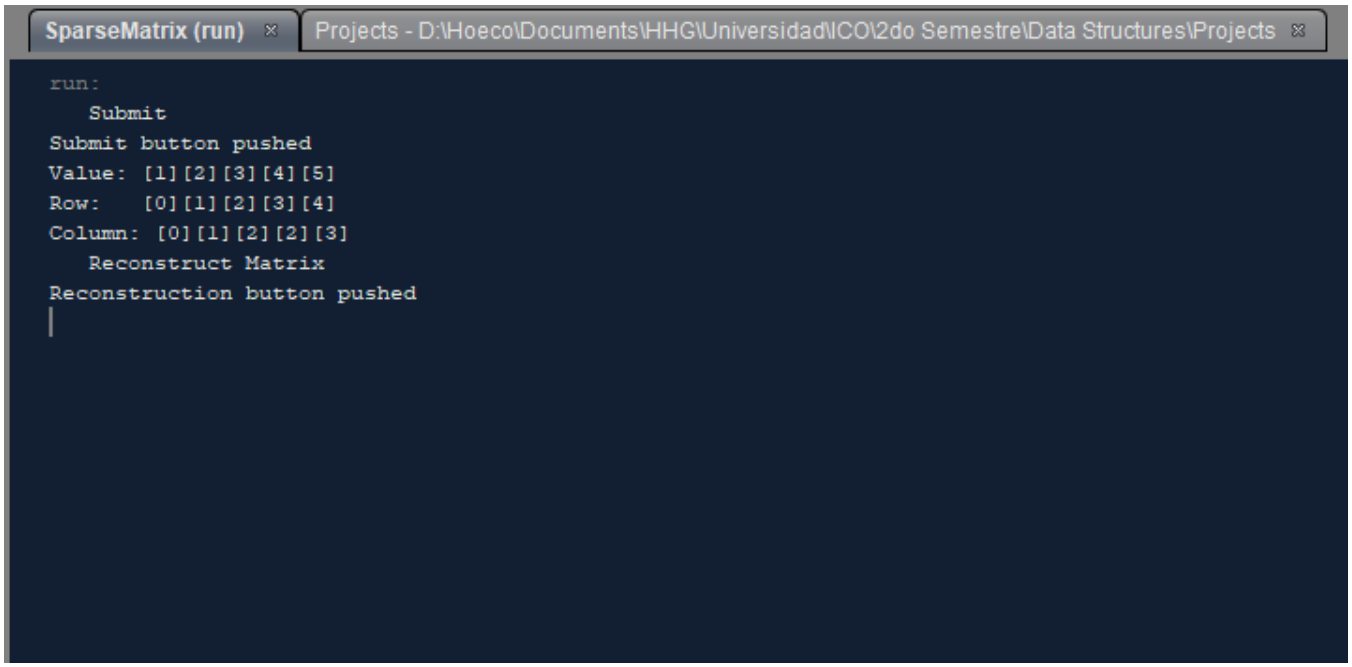
GITHUB

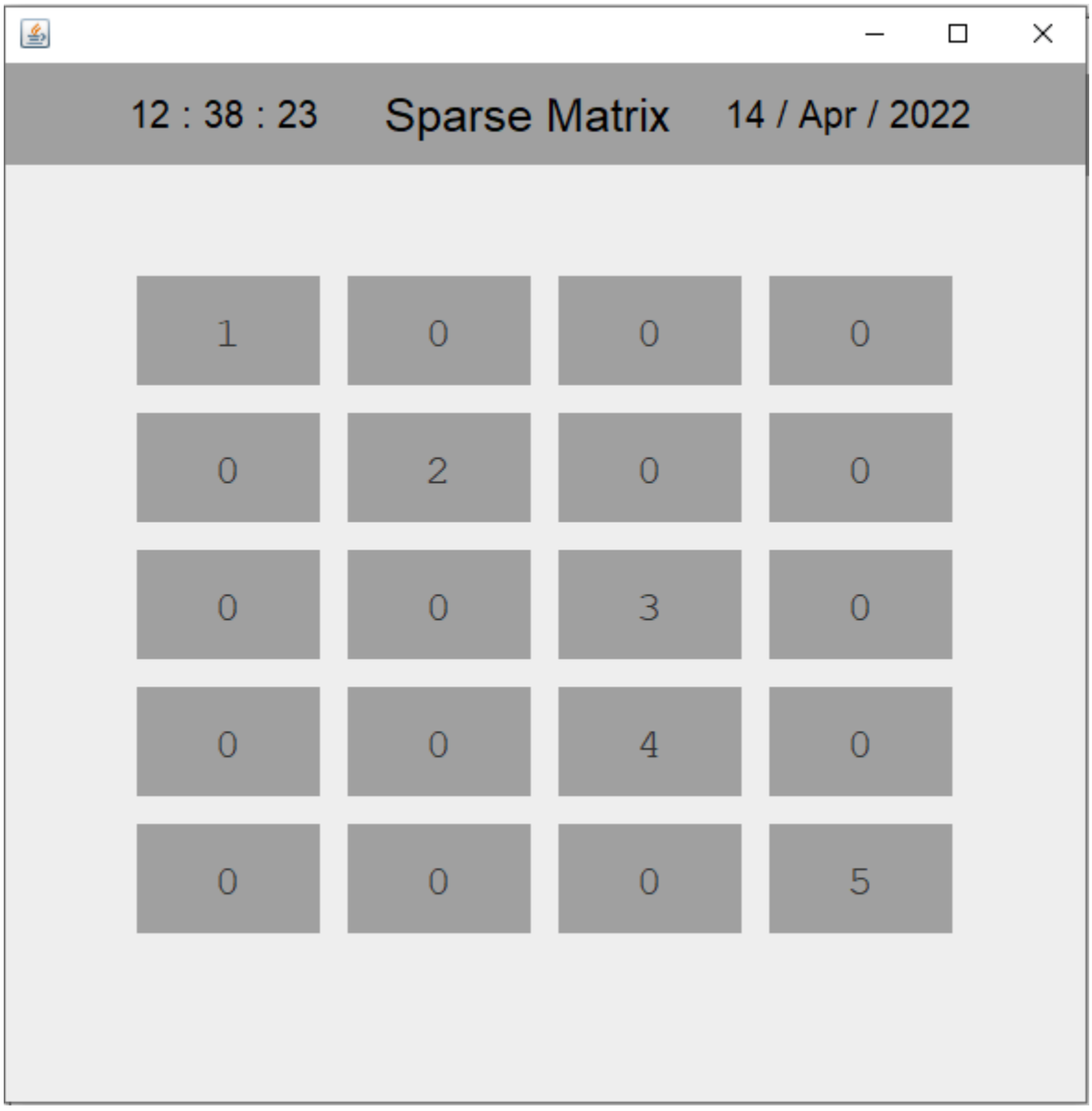
Results




12 : 37 : 08
Sparse Matrix
14 / Apr / 2022
1 0 0 0
0 2 0 0
0 0 3 0
0 0 4 0
0 0 0 5
Submit


12 : 37 : 53
Sparse Matrix
14 / Apr / 2022
Reconstruct Matrix


SparseMatrix (run)
Projects - D:\Hoeco\Documents\HHG\Universidad\ICO\2do Semestre\Data Structures\Projects
run:
Submit
Submit button pushed
Value: [1][2][3][4][5]
Row: [0][1][2][3][4]
Column: [0][1][2][2][3]
Reconstruct Matrix
Reconstruction button pushed


12 : 38 : 23
Sparse Matrix
14 / Apr / 2022
1 0 0 0
0 2 0 0
0 0 3 0
0 0 4 0
0 0 0 5