Saipidin kyzy Upol
ihlg15@inf.elte.hu
IHLG15
Group 7

1. assignment/1. task

6 th of April, 2022

Task

Implement the chessboard matrix type which contains integers. In these matrices, every second entry is zero. The entries that can be nonzero are located like the same-colored squares on a chessboard, with indices (1, 1), (1, 3), (1, 5), ..., (2, 2), (2, 4), ... The zero entries are on the indices (1, 2), (1, 4), ..., (2, 1), (2, 3), ... Store only the entries that can be nonzero in row-major order in a sequence. Don't store the zero entries. Implement as methods: getting the entry located at index (i, j), adding and multiplying two matrices, and printing the matrix (in a shape of n by n).

ChessMatrix matrix type

Set of values

ChessMatrix(n) = { a ∈ $\mathbb{Z}$ n x n | ∀ i,j ∈ [1..n] : i + j % 2 ≠ 0 → a[i, j] = 0}

Operations

1. Getting an entry.

Getting the entry of the **i**th column and **j**th row *(i, j∈[0..n-1]): e:=a[i,j].*

A : ChessMatrix(n x n) x $\mathbb{Z}$ x $\mathbb{Z}$ x $\mathbb{Z}$

a i j e

Pre = (a=a' ∧ i=i' ∧ j=j' ∧ i, j∈*[0..n-1]*))

Post = (Pre ∧ e=a[i,j])

This operation needs any action only if 2 l (i+j), otherwise the output is zero.

2. Setting an entry

Getting the entry of the **i**th column and **j**th row *(i,j∈[0..n-1: a[i,j]:=e. Entries, outside the ChessMatrix cannot be modified (2 ł (i+j)).*

A : ChessMatrix(n x n) x $\mathbb{Z}$ x $\mathbb{Z}$ x $\mathbb{Z}$

a i j e

Pre = (a=a' ∧ i=i' ∧ j=j' ∧ i, j∈*[0..n-1]*))

Post = (Pre ∧ e=a[i,j])

This operation needs any action only if 2 l (i+j), otherwise it gives an error if we want to modify a zero entry.

3. Sum

Sum of two matrices: c:=a+b. The matrices have the same size.

A = (ChessMatrix(n x n) × ChessMatrix(n x n) × ChessMatrix(n x n))
a *b* *c*

Pre = (a=a' ∧ b=b')

Post = (Pre ∧ ∀*i,j*∈[0..n-1]: c[i,j] = a[i,j] + b[i,j])

4. Multiplication

Multiplication of two matrices: c:=a+b. The matrices have the same size.

A = (ChessMatrix(n x n) × ChessMatrix(n x n) × ChessMatrix(n x n))
a *b* *c*

Pre = (a=a' ∧ b=b')

Post = (Pre ∧ ∀*i,j*∈[0..n-1]: c[i,j] = $\sum_{k=0..n-1}$ a[i,k] * b[k,j])

Representaion

Only entries at indices of same property (even-even; odd-odd) have to be stored.

$$a\ (4x4) = \begin{matrix} a_{00} & 0 & a_{02} & 0 \\ 0 & a_{11} & 0 & a_{13} \\ a_{20} & 0 & a_{22} & 0 \\ 0 & a_{31} & 0 & a_{33} \end{matrix} \quad \leftrightarrow v = < a_{00}\ a_{01}\ a_{11}\ a_{13}\ a_{20}\ a_{22}\ a_{31}\ a_{33} >$$

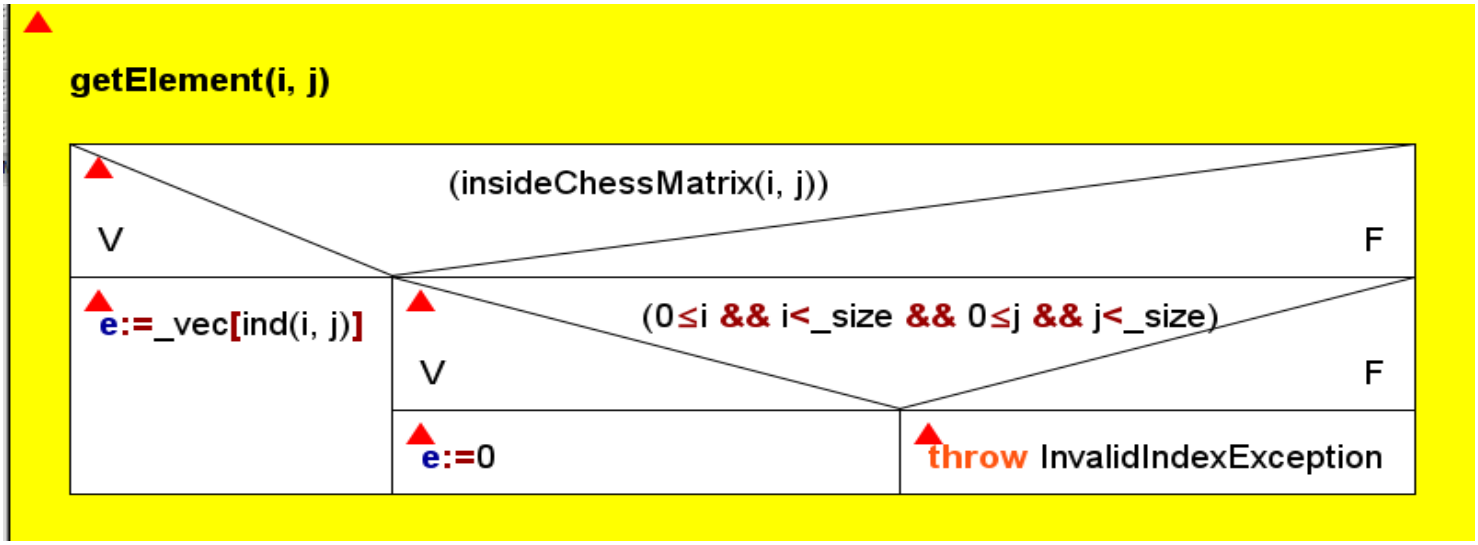
Only one-dimension array (v) is needed, with the help of which any entry of chessboard matrix can be get:

a[i,j] = v[i] if 2 I (i+j)
a[i,j] = 0 if 2 ł (i+j)

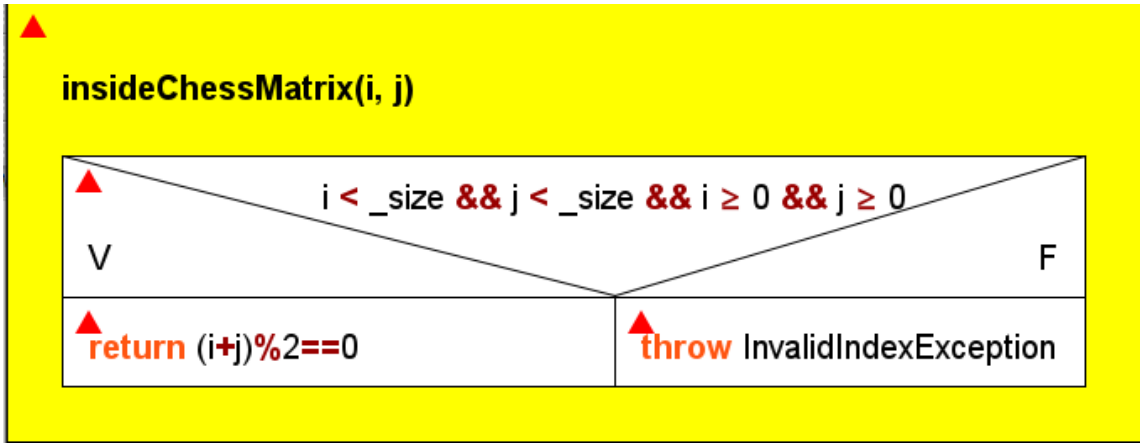
Implementation

1. ***Getting an entry.***

Getting the entry of the ith column and jth row (i,j∈[0..n-1]) e:=a[i,j] where the matrix is represented by _vec, 0≤i≤n-1, and n stands for the size of the matrix can be implemented as

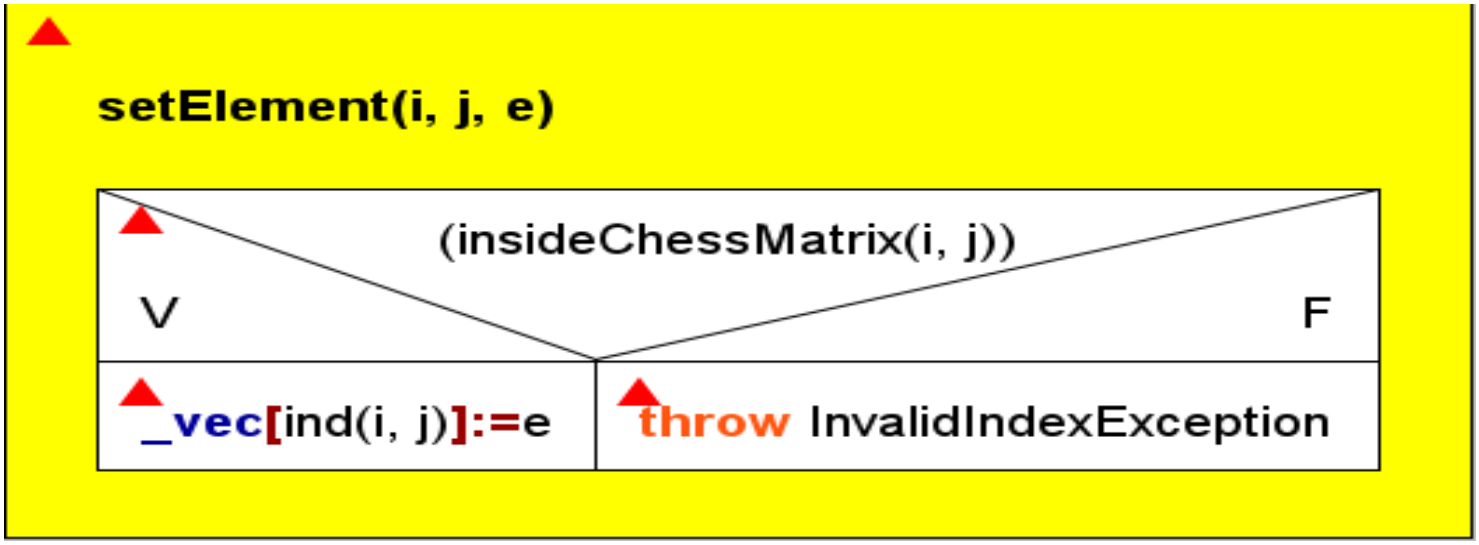


Implementation of insideChessMatrix(i, j)



2. <u>Setting an entry.</u>

Setting the entry of the ith column and jth row (i,j∈[0..n-1]) a[i,j]:=e where the matrix is represented by _vec, 0≤i≤n-1, and n stands for the size of the matrix can be implemented as



3. <u>Sum.</u>

The sum of matrices a and b goes to matrix sum, where all of the arrays have to have the same size.

$$\forall i \in [0..n-1]: \ sum[i] = a[i] + b[i]$$

*4. **<u>Multiplication.</u>***

The product of matrices a and b goes to matrix mul, where all of the arrays have to have the same size.

$$\forall i \in [0..n-1]: \ mul[i] = a[i] * b[i]$$

Testing

Testing the operations

1) Test case 1:

A) Creating the ChessMatrix-matrix:
   A) 4x4 matrix with default values (4);
   B) 5x5 matrix with size 5;
   C) 4x4 matrix with values from the file;
B) Size check;
C) Getting values in ChessMatrix range;
D) Setting the values in ChessMatrix range;
E) Setting vectors for ChessMatrix range;

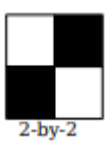
2) Test case 2:

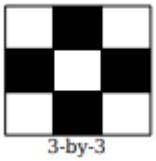
Matrix operations:
A) Sum of two matrices;
B) Multiplication of two matrices;

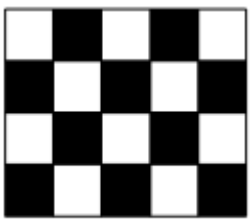
3) Test case 3:

Possible errors to be thrown:
A) INVALID INDEX EXCEPTION - set or get a value out of ChessMatrix matrix;
B) INVALID VECTOR EXCEPTION – set vector out of ChessMatrix matrix;
C) DIFFERENT DIMENSION EXCEPTION - size of two matrices are different on sum and multiplication operations.

Visualization of ChessMatrix


2-by-2


3-by-3


4-by-4