// JAVA Code for Maximum size square

// sub-matrix with all 1s

public class MaxSubMatrix

{

// method for Maximum size square sub-matrix with all 1s

static void printMaxSubSquare(int M[][])

{

int i,j;

int R = M.length; //no of rows in M[][]

int C = M[0].length; //no of columns in M[][]

int S[][] = new int[R][C];

int max\_of\_s, max\_i, max\_j;

/\* Set first column of S[][]\*/

for(i = 0; i < R; i++)

S[i][0] = M[i][0];

/\* Set first row of S[][]\*/

for(j = 0; j < C; j++)

S[0][j] = M[0][j];

/\* Construct other entries of S[][]\*/

for(i = 1; i < R; i++)

{

for(j = 1; j < C; j++)

{

if(M[i][j] == 1)

S[i][j] = Math.min(S[i][j-1],

Math.min(S[i-1][j], S[i-1][j-1])) + 1;

else

S[i][j] = 0;

}

}

/\* Find the maximum entry, and indexes of maximum entry

in S[][] \*/

max\_of\_s = S[0][0]; max\_i = 0; max\_j = 0;

for(i = 0; i < R; i++)

{

for(j = 0; j < C; j++)

{

if(max\_of\_s < S[i][j])

{

max\_of\_s = S[i][j];

max\_i = i;

max\_j = j;

}

}

}

System.out.println("Maximum size sub-matrix: ");

for(i = max\_i; i > max\_i - max\_of\_s; i--)

{

for(j = max\_j; j > max\_j - max\_of\_s; j--)

{

System.out.print(M[i][j] + " ");

}

System.out.println();

}

}

// Driver program

public static void main(String[] args)

{

int M[][] = {{0, 1, 1, 0, 1},

{1, 1, 0, 1, 0},

{0, 1, 1, 1, 0},

{1, 1, 1, 1, 0},

{1, 1, 1, 1, 1},

{0, 0, 0, 0, 0}};

printMaxSubSquare(M);

}

}