```
##### Farhad Ramezanghorbani 20131758 HW#7

class Cell:
    "Single cell in a matrix"

    def __init__( self, data ):
        "Cell constructor"

        self.data = data
        self.nextCell = None
        #self.index = index

    def getData( self ):
        "Get cell data"

        return self.data

    def setData( self, data ):
        "Set cell data"

        self.data = data

    def getNextCell( self ):
        "Get reference to next node"

        return self.nextCell

    def setNextCell( self, newCell ):
        "Set reference to next node"

        self.nextCell = newCell
"""
    def getIndex( self ):
        "Get cell index"

        return self.index

    def setIndex( self ):
        "Set cell index"

        self.index = index
"""

class Row:
    "Linked Row"

    def __init__( self ):
        "List constructor"
        self.firstCell = None
        self.lastCell = None

    def __str__( self ):
        "Override print statement"

        if self.isEmpty():
            return "The row is empty"

        currentCell = self.firstCell
        string = ""

        while currentCell is not None:
            string += str( currentCell.getData() ) + " "
            currentCell = currentCell.getNextCell()
        return string


    def insertAtBack( self, value ):
        "Insert cell at back of row"

        newCell = Cell( value )
        if self.isEmpty():
            self.firstCell = self.lastCell = newCell
        else:
            self.lastCell.setNextCell( newCell )
            self.lastCell = newCell


    def isEmpty( self ):
        "Is the list empty?"

        return self.firstCell is None

##############################################
'''import random as rand
print "enter the matrix dimention"
m = int(raw_input("row: "))
n = int(raw_input("column: "))

Matrix = []
for i in range(m):
    List = []
    for j in range(n):
        List.append(rand.randint(0,1))
    Matrix.append(List)'''

Matrix = [[2, 0, 0, 0, 0], [1, 2, 41, 5, 1], [0, 0, 0, 0, 0], [1, 0, 8, 7, 0], [3, 0, 0, 6, 0]]
m = 5
```

```
n = 5
print "your random matrix is:\n" , Matrix
##############################################
""" index representation of nonzeros """
rowList = Row()

for i in range(m):
    rowList = Row()
    for j in range(n):
            if Matrix[i][j] != 0:
                rowList.insertAtBack(j+1)
    print "row #%.d nonzero element indices:" %(i+1), rowList
##############################################
nonzerow = Row()

for i in range(m):
    j=0
    while j<n:
            if Matrix[i][j] != 0:
                nonzerow.insertAtBack(i+1)
                break

            else:
                j+=1
print "index of non zero rows: " , nonzerow
#############################################
sparse = Row()
for i in range(m):
    for j in range(n):
            if Matrix[i][j] != 0:
                sparse.insertAtBack(Matrix[i][j])
print "non zero elements of matrix: " , sparse
#############################################
```