row	col	k	akk	i	aik	j
3	4	0	-	-	-	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	-	-	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	3	-	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	0	-	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	0	-	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	-	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	3	-

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	3	0

a

1	1	-1	2
3	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	3	0

a

1	1	-1	2
0	5	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	3	1

a

1	1	-1	2
0	2	-7	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	3	2

a

1	1	-1	2
0	2	-4	0
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	1	3	3

a

1	1	-1	2
0	2	-4	-6
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	2	3	3

a

1	1	-1	2
0	2	-4	-6
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	2	2	3

a

1	1	-1	2
0	2	-4	-6
2	-3	1	5

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	0	1	2	2	3

a

1	1	-1	2
0	2	-4	-6
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	1	2	2	3

a

1	1	-1	2
0	2	-4	-6
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	2	2	3

a

1	1	-1	2
0	2	-4	-6
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	3	2	3

a

1	1	-1	2
0	1	-2	-3
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	0	2	3

a

1	1	-1	2
0	1	-2	-3
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	0	1	3

a

1	1	-1	2
0	1	-2	-3
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	0	1	3

a

1	0	1	5
0	1	-2	-3
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	1	1	3

a

1	0	1	5
0	1	-2	-3
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	2	-5	3

a

1	0	1	5
0	1	-2	-3
0	-5	3	1

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	1	2	2	-5	3

a

1	0	1	5
0	1	-2	-3
0	0	-7	-14

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	2	-7	2	-5	3

a

1	0	1	5
0	1	-2	-3
0	0	-7	-14

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	2	-7	3	-5	3

a

1	0	1	5
0	1	-2	-3
0	0	1	2

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```

row	col	k	akk	i	aik	j
3	4	2	-7	2	-5	3

a

1	0	0	3
0	1	0	1
0	0	1	2

```
def gj(a)
  row = a.length()
  col = a [0].length()
  for k in 0..(col-2)
    akk = a[k][k]
    for i in 0..(col-1) # normalize row k
      a[k][i]=a[k][i ]*1.0/akk
    end
    for i in 0..(row-1) # eliminate column k
      if i != k # of all rows but k
        aik = a[i][k]
        for j in k..(col-1)
          a[i][j] = a[i][j] - aik * a[k][j]
        end
      end
    end
  end
  a
end
```