In [1]:

```
# Define A
A=matrix(QQ,[[0,1,7,8],[1,3,3,8],[-2,-5,1,-8]])
show(A)
```

Out[1]:

$$\begin{pmatrix} 0 & 1 & 7 & 8 \\ 1 & 3 & 3 & 8 \\ -2 & -5 & 1 & -8 \end{pmatrix}$$

In [2]:

```
# Since the first entry of first row first column is zero,
# and in order to get the first leading 1, we swap rows
# we swap R1 and R2
# R1:=R2 and R2:=R1
A.swap_rows(0,1)
show(A)
```

Out[2]:

$$\begin{pmatrix} 1 & 3 & 3 & 8 \\ 0 & 1 & 7 & 8 \\ -2 & -5 & 1 & -8 \end{pmatrix}$$

In [3]:

```
# Make use of the leading 1 in first row first column to reduce
# entry below to zero
# we add twice of R1 into R3
# R3:=R3+2*R1
A.add_multiple_of_row(2, 0, 2)
show(A)
```

Out[3]:

$$\begin{pmatrix} 1 & 3 & 3 & 8 \\ 0 & 1 & 7 & 8 \\ 0 & 1 & 7 & 8 \end{pmatrix}$$

In [4]:

```
# Make use of the leading 1 in second row second column to reduce
# entry below to zero
# we take away (minus) R2 from R3
# R3:=R3-R2
A.add_multiple_of_row(2, 1, -1)
show(A)
```

Out[4]:

$$\begin{pmatrix} 1 & 3 & 3 & 8 \\ 0 & 1 & 7 & 8 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

In [5]:

```
# Make use of the leading 1 in second row second column to reduce
# entry above to zero
# we take away three times of R2 from R1
# R1:=R1-3*R2
A.add_multiple_of_row(0, 1, -3)
show(A)
```

Out[5]:

$$\begin{pmatrix} 1 & 0 & -18 & -16 \\ 0 & 1 & 7 & 8 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

In [6]:

```
# Now it is in reduced row echelon form
# If you do not want to do it step by step,
# we can also directly compute it in CoCalc
A=matrix(QQ,[[0,1,7,8],[1,3,3,8],[-2,-5,1,-8]])
show(A.rref())
```

Out[6]:

$$\begin{pmatrix} 1 & 0 & -18 & -16 \\ 0 & 1 & 7 & 8 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

In [0]:

```
```