Solutions

2023-06-28

Chapter 2

1.

a. Show that the following is an Abelian group

$(\mathbb{R}\backslash\{-1\}, \bigoplus)$

where

$a \bigoplus b := ab + a + b, \quad a, b \in \mathbb{R}\backslash\{-1\}$

1. Closure - closure implies that for any two elements in the set, the operation will applied to those elements will produce an element which also is a member of the set. Ours is the set of real numbers omitting -1. Therefore let

$ab + a + b = -1$

$ab + a + b + 1 = 0$

$(a + 1) * (b + 1) = 0$

In order for this equation to be true, either a or b must = -1 (since anything * 0 = 0). Since -1 is not a member of the set, this is a contradiction confirming that we met the closure property.

2. Associativity

$(a \bigoplus b) \bigoplus c =$
$c \bigoplus (ab + a + b) =$
$(ab + a + b)c + (ab + a + b) + c =$
$abc + ac + bc + c + ab + a + b =$
$abc + ac + ab + bc + bc + c + a + b =$
$a(bc + c + b) + a + (bc + b + c) =$
$a(b \bigoplus c) + a + (b \bigoplus c) =$
$a \bigoplus (b \bigoplus c)$

3. Neutral Element

$(a \bigoplus 0) =$
$a * 0 + a + 0 = a =$
$0 * a + 0 + a = a$

4. Inverse Element

$(y \bigoplus x) = 0$
$xy + x + y = 0$
$y = -x - xy$
$x + xy = -y$
$x(1 + y) = -y$
$x = -y/1 + y$

Since -1 is not a member of the group, there is an inverse element.

5. To be an Abelian group the following must hold:

$y \bigoplus x = x \bigoplus y$

Which is cleary true as

$yx + y + x = xy + x + y$

by the commutative property of addition and multiplication.

2.2

a.

$\mathbb{Z}, \oplus$

where $\oplus$ is defined as $\overline{a} + \overline{b} = \overline{a+b}$ for $\overline{a}, \overline{b} \in \mathbb{Z}n$

$\mathbb{Z}n$ is defined as the set of all congruance classes modulo(n)

According to Euclidean division, this is a finite set containing n elements:

$\mathbb{Z}n = \{\bar{0}, \bar{1}, \ldots, \bar{n} - 1\}$

We know that $\mathbb{Z}n$ is closed under addition for some $x, y \in Z$. Further for some $\bar{R} \in \mathbb{Z}n$, $x + y \in \bar{R}$. Therefore $\bar{x} + \bar{y} \in \bar{R} \in \mathbb{Z}n$ showing that $\mathbb{Z}n$ is closed under addition.

$\bar{0} + \bar{x} = \bar{x}$ for some $x \in \bar{R} \in \mathbb{Z}n$ meaning $\bar{0}$ is the neutral element

$(\bar{x} \oplus \bar{y}) \oplus \bar{z} = (\bar{x} + \bar{y}) + \bar{z} = \overline{(x+y)+z} = \overline{x+(y+z)} = \bar{x} \oplus (\bar{y} + \bar{z})$

$\overline{x} \oplus \overline{-x} = \overline{x - x} = \overline{0}$

b.

$\mathbb{Z}, \oplus$

where $\oplus$ is defined as $\overline{a} * \overline{b} = \overline{a * b}$ for $\overline{a}, \overline{b} \in \mathbb{Z}n$

For $\mathbb{Z}n5$

$modulo(n)$	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$
$\overline{0}$	$\overline{0}$	$\overline{0}$	$\overline{0}$	$\overline{0}$	$\overline{0}$
$\overline{1}$	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$
$\overline{2}$	$\overline{0}$	$\overline{2}$	$\overline{4}$	$\overline{1}$	$\overline{3}$
$\overline{3}$	$\overline{0}$	$\overline{3}$	$\overline{1}$	$\overline{4}$	$\overline{2}$
$\overline{4}$	$\overline{0}$	$\overline{4}$	$\overline{3}$	$\overline{2}$	$\overline{1}$

We can tell by examining the table that the $\mathbb{Z}n5 \backslash \{0\}$ is closed under multiplication. The neutral element is $\overline{1}$. The inverse element is $\overline{x}^{-x}$ for $x \in \mathbb{Z}n5$

c.

We can show that $\mathbb{Z}n\backslash\{0\}$ is not a group simply by observing that $\overline{8} * \overline{8} = \overline{64} = \overline{0} \notin \mathbb{Z}n8$

d.

2.3

$G = \begin{bmatrix} 1 & x & z \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R} | x, y, z \in \mathbb{R}^{3x3}$

We know that matrix multiplication is not commutative, therefore if G is a group, it is not Abelian.

Given $A, B \in \mathbb{R}^{3x3}$, $AB = C \in \mathbb{R}^{3x3}$ by the definition of matrix multiplication so we have closure. Associativity is given by the definition of matrix multiplication (for square matrices). The identity or neutral element is given by

$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

The inverse is given by

$G^{-1} = \begin{bmatrix} 1 & -x & -y - xz \\ 0 & 1 & -z \\ 0 & 0 & 1 \end{bmatrix}$

1	7	9
0	1	14
0	0	1

1	-7	89
0	1	-14
0	0	1

Therefore it is a group, but not Abelian.

2.4

a. Since the number of columns in the left hand matrix does not equal the number of rows in the right hand matrix, the product is undefined.

b.

4	3	5
10	9	11
16	15	17

c.

5	7	9
11	13	15
8	10	12

14	6
-21	2

e.

12	3	-3	-12
-3	1	2	6
6	5	1	0
13	12	3	2

2.5

a. The following code snippet illustrates that there is no solution to the system since the rank of the coeefficients matrix does not equal the rank of the augmented matrix.

```
x1 = c(1,2,2,5)
x2 = c(1,5,-1,2)
x3 = c(-1,-7,1,-4)
x4 = c(-1,-5,3,2)


m1 = cbind(x1,x2,x3,x4)

knitr::kable(m1, "simple", row.names = F, col.names = NULL)
```

1	1	-1	-1
2	5	-7	-5
2	-1	1	3
5	2	-4	2

```
x5 = c(1,-2,4,6)

m2 = cbind(x5)

knitr::kable(m2, "simple", row.names = F, col.names = NULL)
```

1
-2
4
6

```r
R(m1) == R(cbind(x1,x2,x3,x4,x5))
```

```
## [1] FALSE
```

b.

```r
x1 = c(1,1,2,-1)
x2 = c(-1,1,-1,2)
x3 = c(0,0,0,0)
x4 = c(0,-3,1,-2)
x5 = c(1,0,-1,-1)


m1 = cbind(x1,x2,x3,x4)

knitr::kable(m1, "simple", row.names = F, col.names = NULL)
```

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

```r
x6 = c(3,6,5,-1)

m2 = cbind(x6)

knitr::kable(m2, "simple", row.names = F, col.names = NULL)
```

3
6
5
-1

```r
R(m1) == R(cbind(x1,x2,x3,x4,x5,x6))
```

```
## [1] TRUE
```

```r
gaussianElimination(cbind(x1,x2,x3,x4,x5,x6),verbose = TRUE)
```

```
##
## Initial matrix:
##      x1 x2 x3 x4 x5 x6
## [1,]  1 -1  0  0  1  3
## [2,]  1  1  0 -3  0  6
## [3,]  2 -1  0  1 -1  5
## [4,] -1  2  0 -2 -1 -1
##
## row: 1
```

```
##
##  exchange rows 1 and 3
##      x1 x2 x3 x4 x5 x6
## [1,]  2 -1  0  1 -1  5
## [2,]  1  1  0 -3  0  6
## [3,]  1 -1  0  0  1  3
## [4,] -1  2  0 -2 -1 -1
##
##  multiply row 1 by 0.5
##      x1   x2 x3   x4   x5   x6
## [1,]  1 -0.5  0  0.5 -0.5  2.5
## [2,]  1  1.0  0 -3.0  0.0  6.0
## [3,]  1 -1.0  0  0.0  1.0  3.0
## [4,] -1  2.0  0 -2.0 -1.0 -1.0
##
##  subtract row 1 from row 2
##      x1   x2 x3   x4   x5   x6
## [1,]  1 -0.5  0  0.5 -0.5  2.5
## [2,]  0  1.5  0 -3.5  0.5  3.5
## [3,]  1 -1.0  0  0.0  1.0  3.0
## [4,] -1  2.0  0 -2.0 -1.0 -1.0
##
##  subtract row 1 from row 3
##      x1   x2 x3   x4   x5   x6
## [1,]  1 -0.5  0  0.5 -0.5  2.5
## [2,]  0  1.5  0 -3.5  0.5  3.5
## [3,]  0 -0.5  0 -0.5  1.5  0.5
## [4,] -1  2.0  0 -2.0 -1.0 -1.0
##
##  multiply row 1 by 1 and add to row 4
##      x1   x2 x3   x4   x5  x6
## [1,]  1 -0.5  0  0.5 -0.5 2.5
## [2,]  0  1.5  0 -3.5  0.5 3.5
## [3,]  0 -0.5  0 -0.5  1.5 0.5
## [4,]  0  1.5  0 -1.5 -1.5 1.5
##
## row: 2
##
##  multiply row 2 by 0.6666667
##      x1   x2 x3        x4         x5       x6
## [1,]  1 -0.5  0  0.500000 -0.5000000 2.500000
## [2,]  0  1.0  0 -2.333333  0.3333333 2.333333
## [3,]  0 -0.5  0 -0.500000  1.5000000 0.500000
## [4,]  0  1.5  0 -1.500000 -1.5000000 1.500000
##
##  multiply row 2 by 0.5 and add to row 1
##      x1   x2 x3         x4         x5       x6
## [1,]  1  0.0  0 -0.6666667 -0.3333333 3.666667
## [2,]  0  1.0  0 -2.3333333  0.3333333 2.333333
## [3,]  0 -0.5  0 -0.5000000  1.5000000 0.500000
## [4,]  0  1.5  0 -1.5000000 -1.5000000 1.500000
##
##  multiply row 2 by 0.5 and add to row 3
##      x1  x2 x3         x4         x5       x6
```

```
## [1,]  1 0.0  0 -0.6666667 -0.3333333 3.666667
## [2,]  0 1.0  0 -2.3333333  0.3333333 2.333333
## [3,]  0 0.0  0 -1.6666667  1.6666667 1.666667
## [4,]  0 1.5  0 -1.5000000 -1.5000000 1.500000
##
##  multiply row 2 by 1.5 and subtract from row 4
##      x1 x2 x3         x4         x5        x6
## [1,]  1  0  0 -0.6666667 -0.3333333  3.666667
## [2,]  0  1  0 -2.3333333  0.3333333  2.333333
## [3,]  0  0  0 -1.6666667  1.6666667  1.666667
## [4,]  0  0  0  2.0000000 -2.0000000 -2.000000
##
## row: 3
##
##  exchange rows 3 and 4
##      x1 x2 x3         x4         x5        x6
## [1,]  1  0  0 -0.6666667 -0.3333333  3.666667
## [2,]  0  1  0 -2.3333333  0.3333333  2.333333
## [3,]  0  0  0  2.0000000 -2.0000000 -2.000000
## [4,]  0  0  0 -1.6666667  1.6666667  1.666667
##
##  multiply row 3 by 0.5
##      x1 x2 x3         x4         x5        x6
## [1,]  1  0  0 -0.6666667 -0.3333333  3.666667
## [2,]  0  1  0 -2.3333333  0.3333333  2.333333
## [3,]  0  0  0  1.0000000 -1.0000000 -1.000000
## [4,]  0  0  0 -1.6666667  1.6666667  1.666667
##
##  multiply row 3 by 0.6666667 and add to row 1
##      x1 x2 x3        x4         x5        x6
## [1,]  1  0  0  0.000000 -1.0000000  3.000000
## [2,]  0  1  0 -2.333333  0.3333333  2.333333
## [3,]  0  0  0  1.000000 -1.0000000 -1.000000
## [4,]  0  0  0 -1.666667  1.6666667  1.666667
##
##  multiply row 3 by 2.333333 and add to row 2
##      x1 x2 x3        x4        x5        x6
## [1,]  1  0  0  0.000000 -1.000000  3.000000
## [2,]  0  1  0  0.000000 -2.000000  0.000000
## [3,]  0  0  0  1.000000 -1.000000 -1.000000
## [4,]  0  0  0 -1.666667  1.666667  1.666667
##
##  multiply row 3 by 1.666667 and add to row 4
##      x1 x2 x3 x4 x5 x6
## [1,]  1  0  0  0 -1  3
## [2,]  0  1  0  0 -2  0
## [3,]  0  0  0  1 -1 -1
## [4,]  0  0  0  0  0  0
##
## row: 4
```

Since the rank of the augmented matrix is the same as the coefficients matrix, we know there is a solution. From the reduced matrix we note the following solution set:

$\{(3 - x5, -2x5, x3, x5 - 1, x5) \in \mathbb{R}\}$

```r
x1 = c(0,0,0)
x2 = c(1,0,1)
x3 = c(0,0,0)
x4 = c(0,1,0)
x5 = c(1,1,0)
x6 = c(0,0,1)
x7 = c(2,-1,1)

m1 = cbind(x1,x2,x3,x4,x5,x6)

m2 = cbind(x1,x2,x3,x4,x5,x6,x7)

gaussianElimination(m2, verbose = TRUE)
```

```
##
## Initial matrix:
##      x1 x2 x3 x4 x5 x6 x7
## [1,]  0  1  0  0  1  0  2
## [2,]  0  0  0  1  1  0 -1
## [3,]  0  1  0  0  0  1  1
##
## row: 1
##
##  subtract row 1 from row 3
##      x1 x2 x3 x4 x5 x6 x7
## [1,]  0  1  0  0  1  0  2
## [2,]  0  0  0  1  1  0 -1
## [3,]  0  0  0  0 -1  1 -1
##
## row: 2
##
## row: 3
##
##  multiply row 3 by -1
##      x1 x2 x3 x4 x5 x6 x7
## [1,]  0  1  0  0  1  0  2
## [2,]  0  0  0  1  1  0 -1
## [3,]  0  0  0  0  1 -1  1
##
##  subtract row 3 from row 1
##      x1 x2 x3 x4 x5 x6 x7
## [1,]  0  1  0  0  0  1  1
## [2,]  0  0  0  1  1  0 -1
## [3,]  0  0  0  0  1 -1  1
##
##  subtract row 3 from row 2
##      x1 x2 x3 x4 x5 x6 x7
## [1,]  0  1  0  0  0  1  1
## [2,]  0  0  0  1  0  1 -2
## [3,]  0  0  0  0  1 -1  1
```

Since the rank of the augmented matrix is the same as the coefficients matrix, we know there is a solution. From the reduced matrix we note the following solution set:

$\{(a, 1-b, c, -2-b, 1+b, b) | a, b, c \in \mathbb{R}\}$

2.7

```r
x1 = c(6,6,0)
x2 = c(4,0,8)
x3 = c(3,9,0)
x4 = c(12,12,12)

m1 = cbind(x1,x2,x3)

m2 = cbind(x1,x2,x3,x4)

R(m1) == R(m2)
```

```
## [1] TRUE
```

```r
gaussianElimination(m2, verbose = TRUE)
```

```
##
## Initial matrix:
##      x1 x2 x3 x4
## [1,]  6  4  3 12
## [2,]  6  0  9 12
## [3,]  0  8  0 12
##
## row: 1
##
##  multiply row 1 by 0.1666667
##      x1        x2  x3 x4
## [1,]  1 0.6666667 0.5  2
## [2,]  6 0.0000000 9.0 12
## [3,]  0 8.0000000 0.0 12
##
##  multiply row 1 by 6 and subtract from row 2
##      x1         x2  x3 x4
## [1,]  1  0.6666667 0.5  2
## [2,]  0 -4.0000000 6.0  0
## [3,]  0  8.0000000 0.0 12
##
## row: 2
##
##  exchange rows 2 and 3
##      x1         x2  x3 x4
## [1,]  1  0.6666667 0.5  2
## [2,]  0  8.0000000 0.0 12
## [3,]  0 -4.0000000 6.0  0
##
##  multiply row 2 by 0.125
##      x1         x2  x3  x4
## [1,]  1  0.6666667 0.5 2.0
```

```
## [2,]   0  1.0000000 0.0 1.5
## [3,]   0 -4.0000000 6.0 0.0
##
##  multiply row 2 by 0.6666667 and subtract from row 1
##      x1 x2  x3  x4
## [1,]  1  0 0.5 1.0
## [2,]  0  1 0.0 1.5
## [3,]  0 -4 6.0 0.0
##
##  multiply row 2 by 4 and add to row 3
##      x1 x2  x3  x4
## [1,]  1  0 0.5 1.0
## [2,]  0  1 0.0 1.5
## [3,]  0  0 6.0 6.0
##
## row: 3
##
##  multiply row 3 by 0.1666667
##      x1 x2  x3  x4
## [1,]  1  0 0.5 1.0
## [2,]  0  1 0.0 1.5
## [3,]  0  0 1.0 1.0
##
##  multiply row 3 by 0.5 and subtract from row 1
##      x1 x2 x3  x4
## [1,]  1  0  0 0.5
## [2,]  0  1  0 1.5
## [3,]  0  0  1 1.0
```

Since the rank of the augmented matrix is the same as the coefficients matrix, we know there is a solution. From the reduced matrix we note the following solution set:

$\{(a, 1.5a, a) | a \in \mathbb{R}\}$

2.8

a. To calculate the inverse we must find a matrix X such that AX = I. We can do this by using gaussian elimination. First we compare the rank of the coefficients matrix with that of the augmented matrix. Since they are not the same, the inverse does not exist.

```
x1 = c(2,3,4)
x2 = c(3,4,5)
x3 = c(4,5,6)

m1 = cbind(x1, x2, x3)

x1 = c(1,0,0)
x2 = c(0,1,0)
x3 = c(0,0,1)

m2 = cbind(m1, x1, x2, x3)

R(m1) == R(m2)
```

```
## [1] FALSE
```

```r
knitr::kable(m1, "simple", row.names = F, col.names = NULL  )
```

2	3	4
3	4	5
4	5	6

b. To find the inverse of the matrix, we need to find the matrx X such that AX = I where I is the identity matrix for A. We create the augmented matrix and use gaussian elimination to isolate the right hand side of the equation.

```
##
## Initial matrix:
##      x1 x2 x3 x4 x1 x2 x3 x4
## [1,]  1  0  1  0  1  0  0  0
## [2,]  0  1  1  0  0  1  0  0
## [3,]  1  1  0  1  0  0  1  0
## [4,]  1  1  1  0  0  0  0  1
##
## row: 1
##
##  subtract row 1 from row 3
##      x1 x2 x3 x4 x1 x2 x3 x4
## [1,]  1  0  1  0  1  0  0  0
## [2,]  0  1  1  0  0  1  0  0
## [3,]  0  1 -1  1 -1  0  1  0
## [4,]  1  1  1  0  0  0  0  1
##
##  subtract row 1 from row 4
##      x1 x2 x3 x4 x1 x2 x3 x4
## [1,]  1  0  1  0  1  0  0  0
## [2,]  0  1  1  0  0  1  0  0
## [3,]  0  1 -1  1 -1  0  1  0
## [4,]  0  1  0  0 -1  0  0  1
##
## row: 2
##
##  subtract row 2 from row 3
##      x1 x2 x3 x4 x1 x2 x3 x4
## [1,]  1  0  1  0  1  0  0  0
## [2,]  0  1  1  0  0  1  0  0
## [3,]  0  0 -2  1 -1 -1  1  0
## [4,]  0  1  0  0 -1  0  0  1
##
##  subtract row 2 from row 4
##      x1 x2 x3 x4 x1 x2 x3 x4
## [1,]  1  0  1  0  1  0  0  0
## [2,]  0  1  1  0  0  1  0  0
## [3,]  0  0 -2  1 -1 -1  1  0
## [4,]  0  0 -1  0 -1 -1  0  1
##
```

```
## row: 3
##
##  multiply row 3 by -0.5
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  1  0.0  1.0  0.0  0.0  0
## [2,]  0  1  1  0.0  0.0  1.0  0.0  0
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0 -1  0.0 -1.0 -1.0  0.0  1
##
##  subtract row 3 from row 1
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  0  0.5  0.5 -0.5  0.5  0
## [2,]  0  1  1  0.0  0.0  1.0  0.0  0
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0 -1  0.0 -1.0 -1.0  0.0  1
##
##  subtract row 3 from row 2
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  0  0.5  0.5 -0.5  0.5  0
## [2,]  0  1  0  0.5 -0.5  0.5  0.5  0
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0 -1  0.0 -1.0 -1.0  0.0  1
##
##  multiply row 3 by 1 and add to row 4
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  0  0.5  0.5 -0.5  0.5  0
## [2,]  0  1  0  0.5 -0.5  0.5  0.5  0
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0  0 -0.5 -0.5 -0.5 -0.5  1
##
## row: 4
##
##  multiply row 4 by -2
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  0  0.5  0.5 -0.5  0.5  0
## [2,]  0  1  0  0.5 -0.5  0.5  0.5  0
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0  0  1.0  1.0  1.0  1.0 -2
##
##  multiply row 4 by 0.5 and subtract from row 1
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  0  0.0  0.0 -1.0  0.0  1
## [2,]  0  1  0  0.5 -0.5  0.5  0.5  0
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0  0  1.0  1.0  1.0  1.0 -2
##
##  multiply row 4 by 0.5 and subtract from row 2
##      x1 x2 x3   x4   x1   x2   x3 x4
## [1,]  1  0  0  0.0  0.0 -1.0  0.0  1
## [2,]  0  1  0  0.0 -1.0  0.0  0.0  1
## [3,]  0  0  1 -0.5  0.5  0.5 -0.5  0
## [4,]  0  0  0  1.0  1.0  1.0  1.0 -2
##
##  multiply row 4 by 0.5 and add to row 3
```

```
##      x1 x2 x3 x4 x1 x2 x3 x4
## [1,]  1  0  0  0  0 -1  0  1
## [2,]  0  1  0  0 -1  0  0  1
## [3,]  0  0  1  0  1  1  0 -1
## [4,]  0  0  0  1  1  1  1 -2
```

1	0	1	0
0	1	1	0
1	1	0	1
1	1	1	0

0	-1	0	1
-1	0	0	1
1	1	0	-1
1	1	1	-2

2.9

a.

b.

c.

d.

2.10

To determine whether the vectors are linearly independent, we gather the vectors in a matrix and use gaussian elimination to reduce it to row-echelon format. We then examine the resulting matrix to find the pivot columns. If every column is a pivot column, then the vectors are linearly independent.

```r
x1 = c(2,-1,3)
x2 = c(1,1,-2)
x3 = c(3,-3,8)

m1 = cbind(x1,x2,x3)

gaussianElimination(m1, verbose = T)
```

a.

```
##
## Initial matrix:
##      x1 x2 x3
## [1,]  2  1  3
## [2,] -1  1 -3
## [3,]  3 -2  8
##
## row: 1
##
##  exchange rows 1 and 3
##      x1 x2 x3
## [1,]  3 -2  8
## [2,] -1  1 -3
## [3,]  2  1  3
##
##  multiply row 1 by 0.3333333
##      x1         x2        x3
## [1,]  1 -0.6666667  2.666667
## [2,] -1  1.0000000 -3.000000
## [3,]  2  1.0000000  3.000000
##
##  multiply row 1 by 1 and add to row 2
##      x1         x2         x3
## [1,]  1 -0.6666667  2.6666667
## [2,]  0  0.3333333 -0.3333333
## [3,]  2  1.0000000  3.0000000
##
##  multiply row 1 by 2 and subtract from row 3
##      x1         x2         x3
## [1,]  1 -0.6666667  2.6666667
## [2,]  0  0.3333333 -0.3333333
## [3,]  0  2.3333333 -2.3333333
##
## row: 2
##
##  exchange rows 2 and 3
##      x1         x2         x3
## [1,]  1 -0.6666667  2.6666667
## [2,]  0  2.3333333 -2.3333333
## [3,]  0  0.3333333 -0.3333333
##
##  multiply row 2 by 0.4285714
##      x1         x2         x3
## [1,]  1 -0.6666667  2.6666667
## [2,]  0  1.0000000 -1.0000000
## [3,]  0  0.3333333 -0.3333333
##
##  multiply row 2 by 0.6666667 and add to row 1
##      x1        x2         x3
## [1,]  1 0.0000000  2.0000000
## [2,]  0 1.0000000 -1.0000000
## [3,]  0 0.3333333 -0.3333333
##
##  multiply row 2 by 0.3333333 and subtract from row 3
```

```
##       x1 x2 x3
## [1,]  1  0  2
## [2,]  0  1 -1
## [3,]  0  0  0
##
## row: 3
```