Exercise 2.3

a. Create and store a sequence of values from 5 to −11 that progresses in steps of 0.3.

```
> myseq <- seq(from=5, to=-11, by=-0.3)
> myseq
 [1]   5.0   4.7   4.4   4.1   3.8   3.5   3.2   2.9   2.6   2.3   2.0   1.7   1.4   1.1
[15]   0.8   0.5   0.2  -0.1  -0.4  -0.7  -1.0  -1.3  -1.6  -1.9  -2.2  -2.5  -2.8  -3.1
[29]  -3.4  -3.7  -4.0  -4.3  -4.6  -4.9  -5.2  -5.5  -5.8  -6.1  -6.4  -6.7  -7.0  -7.3
[43]  -7.6  -7.9  -8.2  -8.5  -8.8  -9.1  -9.4  -9.7 -10.0 -10.3 -10.6 -10.9

>
```

b. Overwrite the object from (a) using the same sequence with the order reversed.

```
> myseq <- seq(from=5, to=-11, by=-0.3)
> myseq
 [1]   5.0   4.7   4.4   4.1   3.8   3.5   3.2   2.9   2.6   2.3   2.0   1.7   1.4   1.1
[15]   0.8   0.5   0.2  -0.1  -0.4  -0.7  -1.0  -1.3  -1.6  -1.9  -2.2  -2.5  -2.8  -3.1
[29]  -3.4  -3.7  -4.0  -4.3  -4.6  -4.9  -5.2  -5.5  -5.8  -6.1  -6.4  -6.7  -7.0  -7.3
[43]  -7.6  -7.9  -8.2  -8.5  -8.8  -9.1  -9.4  -9.7 -10.0 -10.3 -10.6 -10.9
>
> myseq <- seq(from=-11, to=5, by=0.3)
> myseq
 [1] -11.0 -10.7 -10.4 -10.1  -9.8  -9.5  -9.2  -8.9  -8.6  -8.3  -8.0  -7.7  -7.4  -7.1
[15]  -6.8  -6.5  -6.2  -5.9  -5.6  -5.3  -5.0  -4.7  -4.4  -4.1  -3.8  -3.5  -3.2  -2.9
[29]  -2.6  -2.3  -2.0  -1.7  -1.4  -1.1  -0.8  -0.5  -0.2   0.1   0.4   0.7   1.0   1.3
[43]   1.6   1.9   2.2   2.5   2.8   3.1   3.4   3.7   4.0   4.3   4.6   4.9

>
```

c. Repeat the vector c(-1,3,-5,7,-9) twice, with each element repeated 10 times, and store the result. Display the result sorted from largest to smallest.

```
> y <- rep(x=c(-1,3,-5,7,-9), times = 2, each = 10)
> sort(y, decreasing = TRUE)
  [1]  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  3  3  3  3
3  3  3
 [28]  3  3  3  3  3  3  3  3  3  3  3  3  3 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1
 [55] -1 -1 -1 -1 -1 -1 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5
-5 -5 -9
 [82] -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9 -9
```

d. Create and store a vector that contains, in any configuration, the following:

i. A sequence of integers from 6 to 12 (inclusive)

ii. A threefold repetition of the value 5.3

iii. The number −3

iv. A sequence of nine values starting at 102 and ending at the number that is the total length of the vector created in (c)

```
> vecOne <- c(6:12)
>
> vecTwo <- rep(5.3, times = 3)
>
> vecThree <- c(-3)
>
> vecFour <- seq(from = 102, to = 100, length.out = 9)
>
> megaVec = c(vecOne, vecTwo, vecThree, vecFour)
>
> megaVec
 [1]   6.00   7.00   8.00   9.00  10.00  11.00  12.00   5.30   5.30   5.30  -
3.00 102.00
[13] 101.75 101.50 101.25 101.00 100.75 100.50 100.25 100.00
```

e. Confirm that the length of the vector created in (d) is 20.

```
> length(megaVec)
[1] 20
```

Exercise 3.1

a. Construct and store a 4 × 2 matrix that's filled row-wise with the values 4.3, 3.1, 8.2, 8.2, 3.2, 0.9, 1.6, and 6.5, in that order.

```
> A <- matrix(data=c(4.3, 3.1, 8.2, 8.2, 3.2, 0.9, 1.6, 6.5),nrow=4,ncol=2, b
yrow = TRUE)
> A
     [,1] [,2]
[1,]  4.3  3.1
[2,]  8.2  8.2
[3,]  3.2  0.9
[4,]  1.6  6.5
```

b. Confirm the dimensions of the matrix from (a) are 3 × 2 if you remove any one row.

```
> dim(A[-4,])
[1] 3 2
```

c. Overwrite the second column of the matrix from (a) with that same column sorted from smallest to largest.

```
> A[,2] <- sort(x=A[,2])
> A
     [,1] [,2]
[1,]  4.3  0.9
[2,]  8.2  3.1
[3,]  3.2  6.5
[4,]  1.6  8.2
```

d. What does R return if you delete the fourth row and the first column from (c)? Use matrix to ensure the result is a single-column matrix, rather than a vector.

```
> A[-4,]
     [,1] [,2]
```

```
[1,]  4.3  0.9
[2,]  8.2  3.1
[3,]  3.2  6.5
> A[,-1]
[1] 0.9 3.1 6.5 8.2
> matrix(data=A[-4,-1])
     [,1]
[1,]  0.9
[2,]  3.1
[3,]  6.5
```

e. Store the bottom four elements of (c) as a new 2 × 2 matrix.

```
> A[,2] <- sort(x=A[,2])
>
> B <- A[3:4,]
> A
     [,1] [,2]
[1,]  4.3  0.9
[2,]  8.2  3.1
[3,]  3.2  6.5
[4,]  1.6  8.2
> B
     [,1] [,2]
[1,]  3.2  6.5
[2,]  1.6  8.2
```

f. Overwrite, in this order, the elements of (c) at positions (4,2), (1,2), (4,1), and (1,1) with – 1/2 of the two values on the diagonal of (e).

```
> A[c(4,1),2:1] <- -0.5*diag(B)
> A
     [,1] [,2]
[1,] -4.1 -4.1
[2,]  8.2  3.1
[3,]  3.2  6.5
[4,] -1.6 -1.6
```