Exercises 3. Simple Functions

1. **(a)** Write functions `tmpFn1` and `tmpFn2` such that if `xVec` is the vector $(x_1, x_2, \ldots, x_n)$, then `tmpFn1(xVec)` returns the vector $\left(x_1, x_2^2, \ldots, x_n^n\right)$ and `tmpFn2(xVec)` returns the vector $\left(x_1, \frac{x_2^2}{2}, \ldots, \frac{x_n^n}{n}\right)$.

(b) Now write a function `tmpFn3` which takes 2 arguments `x` and `n` where `x` is a single number and `n` is a strictly positive integer. The function should return the value of

$$1 + \frac{x}{1} + \frac{x^2}{2} + \frac{x^3}{3} + \cdots + \frac{x^n}{n}$$

2. Write a function `tmpFn(xVec)` such that if `xVec` is the vector $\mathbf{x} = (x_1, \ldots, x_n)$ then `tmpFn(xVec)` returns the vector of moving averages:

$$\frac{x_1 + x_2 + x_3}{3}, \quad \frac{x_2 + x_3 + x_4}{3}, \quad \ldots, \quad \frac{x_{n-2} + x_{n-1} + x_n}{3}$$

Try out your function; for example, try `tmpFn( c(1:5,6:1) )`.

3. Consider the continuous function

$$f(x) = \begin{cases} x^2 + 2x + 3 & \text{if } x < 0 \\ x + 3 & \text{if } 0 \le x < 2 \\ x^2 + 4x - 7 & \text{if } 2 \le x. \end{cases}$$

Write a function `tmpFn` which takes a single argument `xVec`. The function should return the vector of values of the function $f(x)$ evaluated at the values in `xVec`.
Hence plot the function $f(x)$ for $-3 < x < 3$.

4. Write a function which takes a single argument which is a matrix. The function should return a matrix which is the same as the function argument but every odd number is doubled.

Hence the result of using the function on the matrix

$$\begin{bmatrix} 1 & 1 & 3 \\ 5 & 2 & 6 \\ -2 & -1 & -3 \end{bmatrix}$$

should be:

$$\begin{bmatrix} 2 & 2 & 6 \\ 10 & 2 & 6 \\ -2 & -2 & -6 \end{bmatrix}$$

Hint: First try this for a specific matrix on the Command Line.

5. Write a function which takes 2 arguments `n` and `k` which are positive integers. It should return the $n \times n$ matrix:

$$\begin{bmatrix} k & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & k & 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & k & 1 & \cdots & 0 & 0 \\ 0 & 0 & 1 & k & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & k & 1 \\ 0 & 0 & 0 & 0 & \cdots & 1 & k \end{bmatrix}$$

Hint: First try to do it for a specific case such as $n = 5$ and $k = 2$ on the Command Line.

6. Suppose an angle α is given as a positive real number of degrees.

If $0 \le \alpha < 90$ then it is quadrant 1. If $90 \le \alpha < 180$ then it is quadrant 2.
If $180 \le \alpha < 270$ then it is quadrant 3. If $270 \le \alpha < 360$ then it is quadrant 4.
If $360 \le \alpha < 450$ then it is quadrant 1. And so on.

Write a function `quadrant(alpha)` which returns the quadrant of the angle α.

7. **(a)** Zeller's congruence is the formula:

$$f = \big([2.6m - 0.2] + k + y + [y/4] + [c/4] - 2c\big) \bmod 7$$

where $[x]$ denotes the integer part of x; for example [7.5] = 7.

Zeller's congruence returns the day of the week f given:

k = the day of the month,
y = the year in the century
c = the first 2 digits of the year (the century number)
m = the month number (where January is month 11 of the preceding year, February is month 12 of the preceding year, March is month 1, etc.)

For example, the date 21/07/1963 has $m = 5$, $k = 21$, $c = 19$, $y = 63$; whilst the date 21/2/1963 has $m = 12$, $k = 21$, $c = 19$ and $y = 62$.

Write a function `weekday(day,month,year)` which returns the day of the week when given the numerical inputs of the day, month and year.

Note that the value of 1 for f denotes Sunday, 2 denotes Monday, etc.

(b) Does your function work if the input parameters `day`, `month` and `year` are vectors with the same length and with valid entries?

8. **(a)** Suppose $x_0 = 1$ and $x_1 = 2$ and

$$x_j = x_{j-1} + \frac{2}{x_{j-1}} \qquad \text{for } j = 1, 2, \ldots.$$

Write a function `testLoop` which takes the single argument n and returns the first $n - 1$ values of the sequence $\{x_j\}_{j\geq 0}$: that means the values of $x_0, x_1, x_2, \ldots, x_{n-2}$.

(b) Now write a function `testLoop2` which takes a single argument `yVec` which is a vector. The function should return

$$\sum_{j=1}^{n} e^j$$

where n is the length of `yVec`.

9. *Solution of the difference equation $x_n = rx_{n-1}(1 - x_{n-1})$, with starting value x_1.*

(a) Write a function `quadmap( start, rho, niter )` which returns the vector $(x_1, \ldots, x_n)$ where $x_k = rx_{k-1}(1 - x_{k-1})$ and

`niter` denotes n,
`start` denotes x_1, and
`rho` denotes r.

Try out the function you have written:

- for $r = 2$ and $0 < x_1 < 1$ you should get $x_n \to 0.5$ as $n \to \infty$.
- try `tmp <- quadmap(start=0.95, rho=2.99, niter=500)`

Now switch back to the Commands window and type:

```
plot(tmp, type="l")
```

Also try the plot `plot(tmp[300:500], type="l")`

(b) Now write a function which determines the number of iterations needed to get $| x_n - x_{n-1} | < 0.02$. So this function has only 2 arguments: `start` and `rho`. (For `start=0.95` and `rho=2.99`, the answer is 84.)

10. **(a)** Given a vector $(x_1, \ldots, x_n)$, the sample autocorrelation of lag k is defined to be

$$r_k = \frac{\sum_{i=k+1}^{n} (x_i - \bar{x})(x_{i-k} - \bar{x})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Thus

$$r_1 = \frac{\sum_{i=2}^{n} (x_i - \bar{x})(x_{i-1} - \bar{x})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} = \frac{(x_2 - \bar{x})(x_1 - \bar{x}) + \cdots + (x_n - \bar{x})(x_{n-1} - \bar{x})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Write a function `tmpFn(xVec)` which takes a single argument `xVec` which is a vector and returns a

`list` of two values: r_1 and r_2.
In particular, find r_1 and r_2 for the vector $(2, 5, 8, \ldots, 53, 56)$.

(b) (Harder.) Generalise the function so that it takes two arguments: the vector `xVec` and an integer `k` which lies between 1 and $n - 1$ where n is the length of `xVec`.
The function should return a vector of the values $(r_0 = 1, r_1, \ldots, r_k)$.

If you used a loop to answer part (b), then you need to be aware that much, much better solutions are possible—see exercises 4. (Hint: `sapply`.)