## Instructions

For this assignment you are asked to write a set of small programs. The code should when entered into the R-console produce the answers to the questions below. Please use the comment operator to annotate and organize your code. Add your code to the accompanying `ProblemSet4.R`. Email me this `.R` file by 3:30pm next Wednesday.

The purpose of this problem set is to review the creation of functions.

### Problem 0

In the accompanying `.R` file carefully annotate lines 8 thru 40.

### Problem 1

Consider the function `min()`. This function returns as its output a vector of length one, containing the smallest element of a numeric input vector. Also, consider the function `sort()`. The `sort()` function simply sorts the elements in a vector by size. It outputs the sorted input.

It is your task to write a function – call it `my_min` – which will return the smallest element of any arbitrary numeric vector. However, you are not allowed to use `min()` function in your code. Assume that inputs do not contain `NAs`.

### Problem 2

Without relying on the `mean()` function produce your own function – `my_mean` – which computes the mean of any arbitrary numeric vector. Input vectors do not contain `NAs`.

### Problem 3

Produce a function called `even()` – which evaluates whether the elements of any arbitrary numeric input vector are even. Input vectors do not contain `NAs`. The output of the function should be a vector of type 'character' and a length equal to the input vector's length. For example for some input `x <- c(-2, 3.5, 3)`, the output of `even(x)` should be a 'character' vector of length three containing `"even"` `"odd"` `"odd"`.

### Problem 4

Without relying on the `abs()` function produce your own function – `my_abs` – which returns the absolute value for any arbitrary numeric vector. Input vectors do not contain `NAs`.

### Problem 5

Without relying on the `median()` function produce your own function – `my_median` – which computes the median of any arbitrary numeric vector. Input vectors do not contain `NAs`.

**Extra Credit**

Write a function that returns various different outputs depending on user preference. The function should work for any arbitray numeric input vector. Input vectors do not contain `NA`s. Depending on user choice, the function should return either

1. All elements of the input vectors that are multiples of 2, 3, 4, ..., or 9. The user should have the ability to specify which multiples she wants to be returned.

   For example, your function should allow the user to specify that they want to extract all multiples of 7 from the input vector `c(3.1, 7, 9, -14)` and return a vector containing 7 and -14 back to the user. The user should also be able to alternatively extract all multiples of 2, 3, 4, 5, 6, 8, or 9 if they so desire.

2. The user should also be able to ask for an alternative output to be returned. This output should be a vector of the same length as the input vector, taking on the value `TRUE` for all elements of the input vector that are multiples of 2, 3, 4, ..., 9 — again depending on user choice — and `FALSE` otherwise.

   For example for the above vector `c(3.1, 7, 9, -14)`, the function should return `FALSE`, `FALSE`, `TRUE`, `FALSE` if the user specifies they'd like to evaluate which elements of the vector are multiple of 9.