Part II

Please submit your assignment as an R script file named with your last name, student number, assignment number and with the suffix R. For example, if Joe Smith, student number 87654321 hands in Assignment 4, he would name the file `Smith87654321A2.R`. Within your answer file, include answers with your R code preceded by the `#` sign. For example, to answer the 5th question on an assignment which is "Perform the calculation $2 + 2$", you would type

```
# Question 5
2 + 2 #coding
# 4 (your answer here )
```

1. (a) (2 points) If a right-angle triangle has sides of length a and b, then the hypotenuse has length $h = \sqrt{a^2 + b^2}$. Compose a function called `pythag` which takes vectors a and b as input and returns h.

   (b) (4 points) Use the `rep()` function to create a vector of length 2100 called `VF` which consists of a repeating pattern of three 0's followed by four 1's. The first 35 elements are below:

   ```
   ##  [1] 0 0 0 1 1 1 1 0 0 0 1 1 1 1 0 0 0 1 1 1 1 0 0
   ## [24] 0 1 1 1 1 0 0 0 1 1 1 1
   ```

   Convert it to a factor. Identify the levels of the result, and then change the level labels to obtain the factor (still of length 2100) whose first 35 entries are

   ```
   ##  [1] Male   Male   Male   Female Female Female
   ##  [7] Female Male   Male   Male   Female Female
   ## [13] Female Female Male   Male   Male   Female
   ## [19] Female Female Female Male   Male   Male
   ## [25] Female Female Female Female Male   Male
   ## [31] Male   Female Female Female Female
   ## Levels: Male Female
   ```

   (c) (3 points) $\binom{n}{m} = \frac{n!}{(n-m)!m!}$. Write a function called `BinCal` that takes arguments `m` and `n`, and calculates and returns this quantity. Use your function to calculate $\binom{10}{4}$.

   (d) (3 points) Complete the following function so that it takes a vector `x` and prints the character string `"Error: At least one element of your input is negative."` if the minimum value of `x` is negative, and returns the square roots of the values of `x`, otherwise.

   ```
   sqrtError <- function(x) {
      ....
   }
   ```

   (e) (3 points) Modify the merge sort function (described in the lectures, and written out below) so that it takes a logical argument (called `decrease`) and returns the sorted values in decreasing order when `decrease` is set to `TRUE`.

   ```
   mergesort <- function (x) {
       len <- length(x)
       if (len < 2) result  <- x
       else {
            y <- x[1:(len / 2)]
           z <- x[-(1:(len / 2))]
           y <- mergesort(y)
           z <- mergesort(z)
   ```

```
    result <- c()
    while (min(length(y), length(z)) > 0) {
        if (y[1] < z[1]) {
            result <- c(result, y[1])
            y <- y[-1]
        } else {
            result <- c(result, z[1])
            z <- z[-1]
        }
    }
    if (length(y) > 0)
        result <- c(result, y)
    else
        result <- c(result, z)
  }
  return(result)
}
```

2. (5 points) Define a sequence of numbers $f_1, f_2, \ldots$ as follows:

$$f_1 = 1$$
$$f_2 = 1$$

For $n = 3, 4, \ldots$,

$$f_n = 1.3 f_{n-1} - 0.25 f_{n-2}.$$

Use a `for()` loop to find the value of f_{200}.

3. (5 points) A twin prime is a pair of primes (x, y), such that $y = x + 2$. Write a function called `twinprimes` that takes `n` as input and returns a 2-column matrix whose rows contain the pairs of twin primes whose values are less than `n`. (You may use the sieve of Eratosthenes described in the lectures as a starting point for this.) For example, the output of `twinprimes(13)` is

```
##      [,1] [,2]
## [1,]    3    5
## [2,]    5    7
## [3,]   11   13
```

4. Consider the wine quality data in `table.b11` in the *MPV* package.
   (a) (1 point) List the variables (names of columns) in the `table.b11` data frame.
   (b) (5 points) Obtain a scatter plot of `Quality` versus `Aroma`, and overlay the line of best-fit, after assigning the relevant `lm()` object to `wine.lm`.
   (c) (3 points) Use your fitted line to predict the Quality of wine that has Aroma level 4.
   (d) (4 points) Use the `xyplot()` function in the *lattice* package to plot `Quality` versus `Aroma` for each value of `Region`. Include both the plotted points and a smoothed curve in each panel of this display, using `span=2`.

(e) (2 points) Use the `xyplot()` function in the *lattice* package to plot `Quality` versus `Aroma` for each value of `Region`, and using `(Clarity < 1)` as the `group` variable. Include both the plotted points, coded as "0", if clarity is less than 1, and "1", if clarity equals 1, and a smoothed curve in each panel of this display. The plot should appear as below.

5. (a) (3 points) Use the `rpart()` function in the *rpart* package to obtain the default regression tree which would predict Quality from at least one of the other variables in the `table.b11` data frame. Assign the resulting tree object to `wine.rpart`, and plot the tree, together with the relevant text information which is required so that the tree can be read.

   (b) (3 points) Predict the Quality of wine having the following characteristics:

   ```
   Clarity=1, Aroma=7, Body=4,  Flavor=1.5, Oakiness=3, Region=2
   ```

6. (a) (2 points) Use the `lm()` function default multiple regression model which would predict Quality from all of the other variables in the `table.b11` data frame. Assign the resulting lm object to `wine.lm`.

   (b) (2 points) Predict the Quality of wine having the following characteristics:

   ```
   Clarity=1, Aroma=7, Body=4,  Flavor=1.5, Oakiness=3, Region=2
   ```