PSTAT 10 Worksheet 2

Due 6/28/22 11:59pm

Problem 1: Basic vector manipulation

1. Recall from lecture my 2021 monthly gas bill in order was given by:

```
gasbill <- c(46, 33, 39, 37, 46, 30, 48, 32, 49, 35, 30, 48)
```

It turns out the charge for December should have been 49 instead of 48. Update the `gasbill` to reflect the true charge. Try not to "cheat" and just type in all the old values again; use the existing `gasbill` vector.

2. Recreate the following numeric vector. Avoid typing in all of the values manually.

```
## [1] -50 -51 -52 -53 -54 -53 -52 -51 -50
```

3. Create a vector from 1 to 10 with increments of 0.05. What is the length of this vector? Hint: Use `seq` with `by` argument.
4. Create a vector of length 100 from 1 to 10 with uniform increments. What is the increment? Hint: Use `seq` with `length` argument.
5. What happens if you try to use `seq` with both the `length` and `by` arguments specified?

Problem 2

Download the file `ws2.csv` from the course website and import it into R. This data set has two variables named `x` and `y`.

```
ws2_df <- read.csv("ws2.csv")
summary(ws2_df)
```

```
##        x                y         
##  Min.   : 2.00   Min.   :  1.00  
##  1st Qu.:25.75   1st Qu.: 26.00  
##  Median :49.50   Median : 53.50  
##  Mean   :49.11   Mean   : 52.93  
##  3rd Qu.:70.00   3rd Qu.: 78.00  
##  Max.   :99.00   Max.   :100.00
```

Remember the variables in a data frame are accessed by name with the dollar sign (and that the result is a vector).

1. Determine the lengths of `x` and `y`.
2. What is the 40th element of `x` and the 80th element of `y`?

3. What is the average of all the values in the data frame, including both `x` and `y`?
4. How many elements of `x` are greater than 70?

Let's look at the first 4 elements of `x` and `y`:

```
ws2_df$x[1:4]
```

```
## [1] 74 89 78 23
```

```
ws2_df$y[1:4]
```

```
## [1] 58 26 48 80
```

The first three elements of `x` are greater than or equal to their corresponding element in `y`: $74 > 58$, $89 > 26$, $78 > 48$. But the fourth element of `x`, 23, is less than the fourth element of `y`, 80.

5. How many elements of `x` are greater than or equal to the corresponding element in `y`?
6. What is the proportion of elements of `x` that are greater than or equal to the corresponding element in `y`?
7. How many values in `x` differ from their corresponding value in `y` by more than 10 in absolute value? *Hint: there is an* `abs` *function.*

Problem 3

Create a vector of integers from 1 to 12 inclusive.

1. Use the vector to create a 3x4 matrix. Did recycling occur?
2. Use the vector to create a 4x4 matrix. Did recycling occur?

Problem 4

Use the `heights_df` data frame from worksheet 1. The `height` variable is given in centimeters (cm).

1. Write a vectorized function `cm_to_inch` that takes a numeric centimeter and converts it to inches. Apply the function to the height vector. First 10 elements are shown below:

```
head(cm_to_inch(heights_df$height), 10) # the head function gives the first elements
```

```
##  [1] 62.40 67.08 65.52 71.37 68.25 73.71 60.84 65.13 76.05 64.35
```

2. Write a vectorized function `cm_to_ft_inch` that converts numerical values given in cm to a `feet inch` format, rounding to the nearest inch.

For example,

```
cm_to_ft_inch(178)
```

```
## [1] "5 9"
```

You may need the (vectorized) quotient function `%/%` and the remainder function `%%`:

```r
# Quotient: 3 goes into 7 two times
7 %/% 3
```

```
## [1] 2
```

```r
# Remainder: The remainder when 7 is divided by 3 is one
7 %% 3
```

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

Remember you should look things up on StackOverflow if you're stuck with some operations.

Apply the function to the `height` vector.

```r
head(cm_to_ft_inch(heights_df$height), 10)
```

```
##  [1] "5 2"  "5 7"  "5 6"  "5 11" "5 8"  "6 2"  "5 1"  "5 5"  "6 4"  "5 4"
```