Problem 1

Yufei Yin

```
library(tidyverse)
library(lubridate)
load('translink.RData')
```

Question 1

Provide a pdf containing all of the code that you wrote for this project. Begin the code with a comment indicating which project you've done. If your code spans multiple files, put all of the code into a single pdf and use comments to indicate the beginning of the files and the file names. You may use use code from outside sources in your project provided that it is released under an open license and provided that you clearly indicate the code with comments. (5 points)

```
# goal:
# translink() function will extract bus number from translink twitter page
# translink() will return a list include the bus number that start detouring and stop detouring

# translink take year, month, day, and hour as arguments
translink = function(yearin, monthin, dayin, hourin){

  # extract text and date from data
  text <- NULL
  date <- NULL
  for (i in seq_along(data)){
    text[i] = data[[i]]$text
    date[i] = ymd_hms(data[[i]]$created)
  }

  # convert numeric to character
  date = as.POSIXct(date, origin = '1970-01-01', tz = 'UTC')

  year = year(date)
  month = month(date)
  day = day(date)
  hour= hour(date)

  # create a dataframe contains text, date, year, month, day, and hour
  df = data.frame(text = text, date = date, year = year, month = month, day = day, hour = hour)

  # based on the user input, use filter() from tidyverse to get the new tibble
  df2 = filter(df, year == yearin, month == monthin, day == dayin, hour == hourin)

  # text in specified date
  text = df2$text
```

```
  # remove the links (links contains "/" which is also used to seperate bus number)
  noslash = str_remove_all(text,"https://(.*)")

  # convert to lower case
  lower = str_to_lower(noslash)

  # use for loop to extract string contains bus number
  start = NULL
  stop = NULL
  s = NULL


  # twitters start with #rideralert, #rideralert update:

  # (single bus number)
  # followed zero or more r or n and followed by numbers

  #(multiple bus numbers seperated by "/)
  # or followed by "/" and zero or more spaces and followed by r or n and numbers
  pattern =
    "(#rideralert|#rideralert update(:)?) (r?|n?)\\d+|/(\\s+)*(r?|n?)\\d+"
  for (j in seq_along(lower)){
    s[j] = str_extract_all(lower[j], pattern)
  }

  # according to key word "clear" "over" to classify bus number
  for (k in seq_along(s)){
    if (str_detect(lower[k], "clear|over")){
      stop[k] = s[k]
    }else{
      start[k] = s[k]
    }
  }

  # clearly indicate the bus numbers and return the list
  start = unlist(start)
  stop = unlist(stop)
  if(length(start)>0){
    start = unique(start) %>%
      str_remove_all("#rideralert|#rideralert update:|/|\\s+") %>%
      str_to_upper()
  }
  if(length(stop)>0){
    stop = unique(stop) %>%
      str_remove_all("#rideralert|#rideralert update:|/|\\s+") %>%
      str_to_upper()
  }
  return(list(df = df2, start = start, stop = stop))
}
```