Part1) Strings (60 points)

Use the stringr functions for the following:

Initialize the vector of words from MLK's speech with the following code:

file <- "https://people.bu.edu/kalathur/datasets/mlk.txt"

words <- scan(file, what=character())

```
> library(stringr)#for part 1

载入程辑包：'stringr'

The following object is masked _by_ '.GlobalEnv':

    words

>
> #part1
> file <- "https://people.bu.edu/kalathur/datasets/mlk.txt"
```

a) Detect and show all the words that have a punctuation symbol.

```
> #a
> words <- scan(file, what=character())
Read 288 items
> punct_words <- words[str_detect(words, "[[:punct:]]")]
> punct_words
 [1] "today,"         "friends,"       "moment,"        "dream."         "dream."         "creed:"         "self-evident:"
 [8] "equal."         "slave-owners"   "brotherhood."   "Mississippi,"   "state,"         "oppression,"    "justice."
[15] "character."     "today."         "Alabama,"       "governor's"     "nullification," "brothers."     "today."
[22] "exalted,"       "low,"           "plain,"         "straight,"      "revealed,"      "together."
```

b) Replace all the punctuation symbols in the words dataset with an empty string. Convert all the resulting words to lower case.

Make this the new_words dataset.

```
> #b
> new words <- str replace all(words, "[[:punct:]]", "") %>% tolower()
```

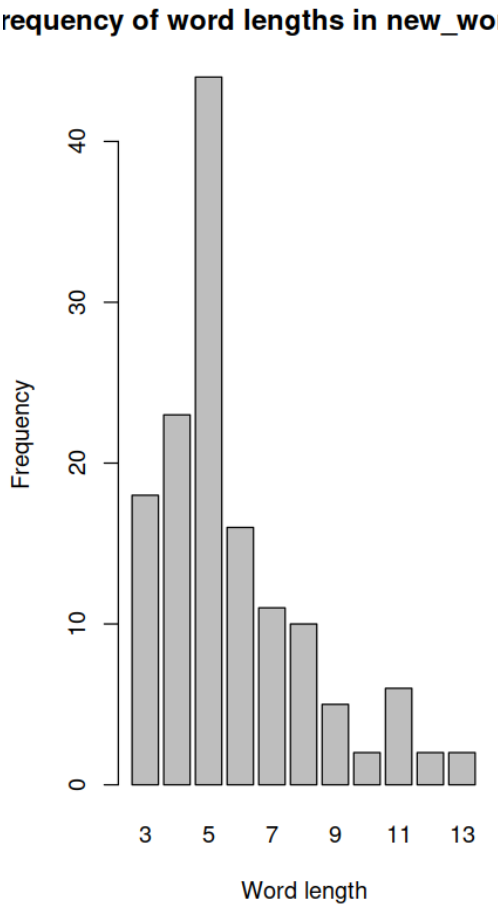
```
> new_words
  [1] "i"             "say"           "to"            "you"           "today"         "my"            "friends"       "that"
  [9] "in"            "spite"         "of"            "the"           "difficulties"  "and"           "frustrations"  "of"
 [17] "the"           "moment"        "i"             "still"         "have"          "a"             "dream"         "it"
 [25] "is"            "a"             "dream"         "deeply"        "rooted"        "in"            "the"           "american"
 [33] "dream"         "i"             "have"          "a"             "dream"         "that"          "one"           "day"
 [41] "this"          "nation"        "will"          "rise"          "up"            "and"           "live"          "out"
 [49] "the"           "true"          "meaning"       "of"            "its"           "creed"         "we"            "hold"
 [57] "these"         "truths"        "to"            "be"            "selfevident"   "that"          "all"           "men"
 [65] "are"           "created"       "equal"         "i"             "have"          "a"             "dream"         "that"
 [73] "one"           "day"           "on"            "the"           "red"           "hills"         "of"            "georgia"
 [81] "the"           "sons"          "of"            "former"        "slaves"        "and"           "the"           "sons"
 [89] "of"            "former"        "slaveowners"   "will"          "be"            "able"          "to"            "sit"
 [97] "down"          "together"      "at"            "a"             "table"         "of"            "brotherhood"   "i"
[105] "have"          "a"             "dream"         "that"          "one"           "day"           "even"          "the"
[113] "state"         "of"            "mississippi"   "a"             "desert"        "state"         "sweltering"    "with"
[121] "the"           "heat"          "of"            "injustice"     "and"           "oppression"    "will"          "be"
[129] "transformed"   "into"          "an"            "oasis"         "of"            "freedom"       "and"           "justice"
[137] "i"             "have"          "a"             "dream"         "that"          "my"            "four"          "children"
[145] "will"          "one"           "day"           "live"          "in"            "a"             "nation"        "where"
[153] "they"          "will"          "not"           "be"            "judged"        "by"            "the"           "color"
[161] "of"            "their"         "skin"          "but"           "by"            "the"           "content"       "of"
[169] "their"         "character"     "i"             "have"          "a"             "dream"         "today"         "i"
[177] "have"          "a"             "dream"         "that"          "one"           "day"           "the"           "state"
[185] "of"            "alabama"       "whose"         "governors"     "lips"          "are"           "presently"     "dripping"
[193] "with"          "the"           "words"         "of"            "interposition" "and"           "nullification" "will"
[201] "be"            "transformed"   "into"          "a"             "situation"     "where"         "little"        "black"
[209] "boys"          "and"           "black"         "girls"         "will"          "be"            "able"          "to"
[217] "join"          "hands"         "with"          "little"        "white"         "boys"          "and"           "white"
[225] "girls"         "and"           "walk"          "together"      "as"            "sisters"       "and"           "brothers"
[233] "i"             "have"          "a"             "dream"         "today"         "i"             "have"          "a"
[241] "dream"         "that"          "one"           "day"           "every"         "valley"        "shall"         "be"
[249] "exalted"       "every"         "hill"          "and"           "mountain"      "shall"         "be"            "made"
[257] "low"           "the"           "rough"         "places"        "will"          "be"            "made"          "plain"
[265] "and"           "the"           "crooked"       "places"        "will"          "be"            "made"          "straight"
[273] "and"           "the"           "glory"         "of"            "the"           "lord"          "shall"         "be"
[281] "revealed"      "and"           "all"           "flesh"         "shall"         "see"           "it"            "together"
>
```

c) What are the top 5 frequent words in the new_words dataset?

```
> #c
> stopfile <- "https://people.bu.edu/kalathur/datasets/stopwords.txt"
> stopwords <- scan(stopfile, what=character())
Read 176 items
> new_words_clean <- new_words[!(new_words %in% stopwords)]
> freq_words_clean <- table(new_words_clean)
> top_words_clean <- sort(freq_words_clean, decreasing = TRUE)[1:5]
> top_words_clean
new_words_clean
dream   day   one shall  made
   11     6     6     4     3
```

d) Show the frequencies of the word lengths in the new_words dataset. Plot the distribution of these frequencies.

```
> #d
> word_lengths_clean <- str_length(new_words_clean)
> freq_lengths_clean <- table(word_lengths_clean)
> barplot(freq_lengths_clean, main = "Frequency of word lengths in new_words_clean", xlab = "Word length", ylab = "Frequency")
>
```



e) What are the words in the new_words dataset with the longest length?

```
> #e
> longest_words <- new_words[which.max(nchar(new_words))]
> longest_words
[1] "interposition"
```

f) Show all the words in the new_words dataset that start with the letter c.

```
> #f
> c_words <- str_subset(new_words, "^c")
> c_words
[1] "creed"     "created"   "children"  "color"     "content"   "character" "crooked"
```

g) Show all the words in the new_words dataset that end with the letter r.

```
> #g
> r_words <- str_subset(new_words, "r$")
> r_words
 [1] "former"    "former"    "together"  "four"      "color"     "their"     "their"     "character" "together"  "together"
```

h) Show all the words in the new_words dataset that start with the letter c and end with the letter r

```
> #h
> cr_words <- str_subset(new_words, "^c.*r$")
> cr_words
[1] "color"     "character"
```

Part2) Data Wrangling (40 points)

Use the tidyverse library for the following:
Download the following csv file,
https://people.bu.edu/kalathur/usa_daily_avg_temps.csv
locally first and use read.csv to load the data into a data frame.

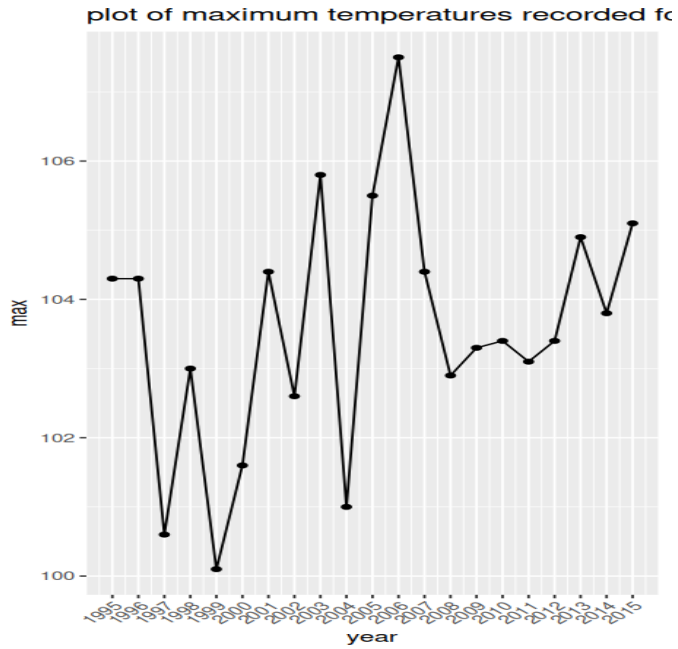
```
> #part2
> library(tidyverse)
── Attaching core tidyverse packages ──────────────────────────────────────────────────────────────────────────────── tidyverse 2.0.0 ──
✔ dplyr     1.1.0     ✔ purrr     1.0.1
✔ forcats   1.0.0     ✔ readr     2.1.4
✔ ggplot2   3.4.1     ✔ tibble    3.1.8
✔ lubridate 1.9.2     ✔ tidyr     1.3.0
── Conflicts ──────────────────────────────────────────────────────────────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package to force all conflicts to become errors
>
>
> df <- read.csv("http://people.bu.edu/kalathur/usa_daily_avg_temps.csv")
>
```

a) Convert the data frame into a tibble and assign it to the variable usaDailyTemps.

```
> #a
> usaDailyTemps <-as_tibble(df)
> usaDailyTemps
# A tibble: 1,174,605 × 6
   state   city       month   day  year avgtemp
   <chr>   <chr>      <int> <int> <int>   <dbl>
 1 Alabama Birmingham     1     1  1995    50.7
 2 Alabama Birmingham     1     1  1996    56.8
 3 Alabama Birmingham     1     1  1997    60.9
 4 Alabama Birmingham     1     1  1998    35.6
 5 Alabama Birmingham     1     1  1999    41
 6 Alabama Birmingham     1     1  2000    59
 7 Alabama Birmingham     1     1  2001    27
 8 Alabama Birmingham     1     1  2002    28.1
 9 Alabama Birmingham     1     1  2003    51.7
10 Alabama Birmingham     1     1  2004    47.9
# … with 1,174,595 more rows
# ℹ Use `print(n = ...)` to see more rows
```

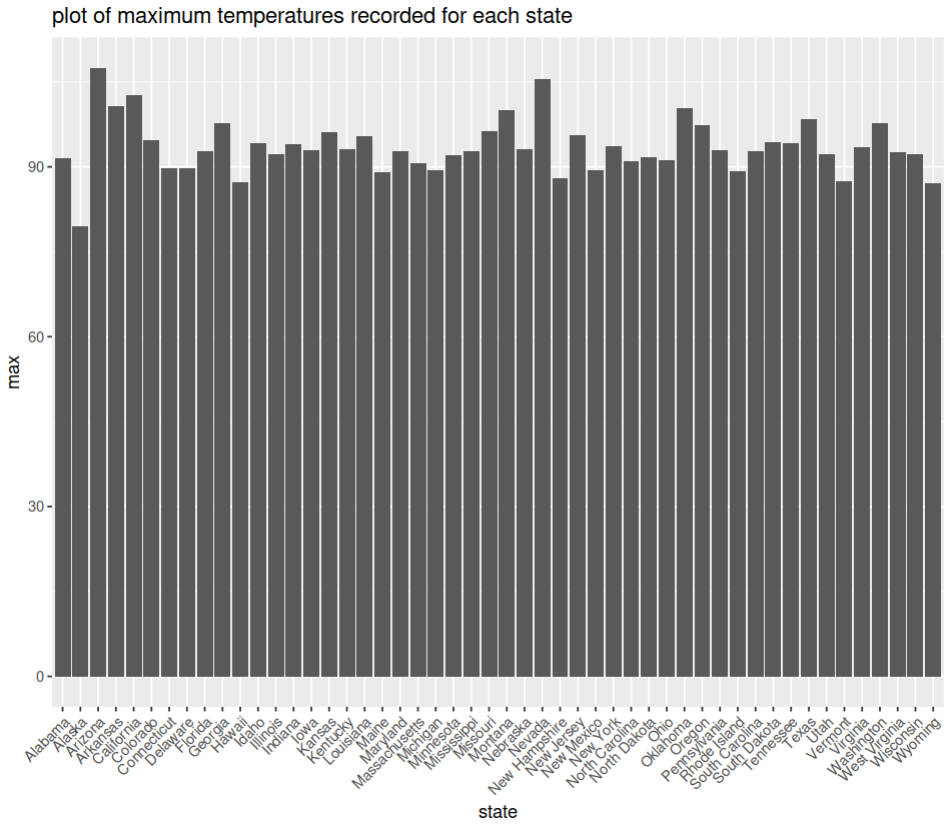
b) What are the maximum temperatures recorded for each year? Show the values and also the appropriate plot for the results.

```
> #b
> max_temp_yr <- usaDailyTemps %>% group_by(year) %>% summarise(max = max(avgtemp))
> max_temp_yr
# A tibble: 21 × 2
    year   max
   <int> <dbl>
 1  1995  104.
 2  1996  104.
 3  1997  101.
 4  1998  103
 5  1999  100.
 6  2000  102.
 7  2001  104.
 8  2002  103.
 9  2003  106.
10  2004  101
# … with 11 more rows
# ℹ Use `print(n = ...)` to see more rows
> ggplot(data = max_temp_yr, aes(x=year, y=max, group=1))+
+   geom_line()+
+   geom_point()+ggtitle("plot of maximum temperatures recorded for each year")+
+   scale_x_continuous(breaks = (seq(min(max_temp_yr$year), max(max_temp_yr$year), by = 1)))+
+   theme(axis.text.x = element_text(angle = 45, hjust = 1))
>
```



c) What are the maximum temperatures recorded for each state? Show the values and also the appropriate plot for the results.

```
> #c
> max_temp_state <- usaDailyTemps %>% group_by(state) %>% summarise(max = max(avgtemp))
> max_temp_state
# A tibble: 50 × 2
   state         max
   <chr>       <dbl>
 1 Alabama      91.5
 2 Alaska       79.5
 3 Arizona     108.
 4 Arkansas    101.
 5 California  103.
 6 Colorado     94.7
 7 Connecticut  89.8
 8 Delaware     89.7
 9 Florida      92.8
10 Georgia      97.7
# … with 40 more rows
# i Use `print(n = ...)` to see more rows
> ggplot(data = max_temp_state, aes(x=state, y=max, group=1))+
+   geom_bar(stat = 'identity')+
+   ggtitle("plot of maximum temperatures recorded for each state")+
+   theme(axis.text.x = element_text(angle = 45, hjust = 1))
> 
```

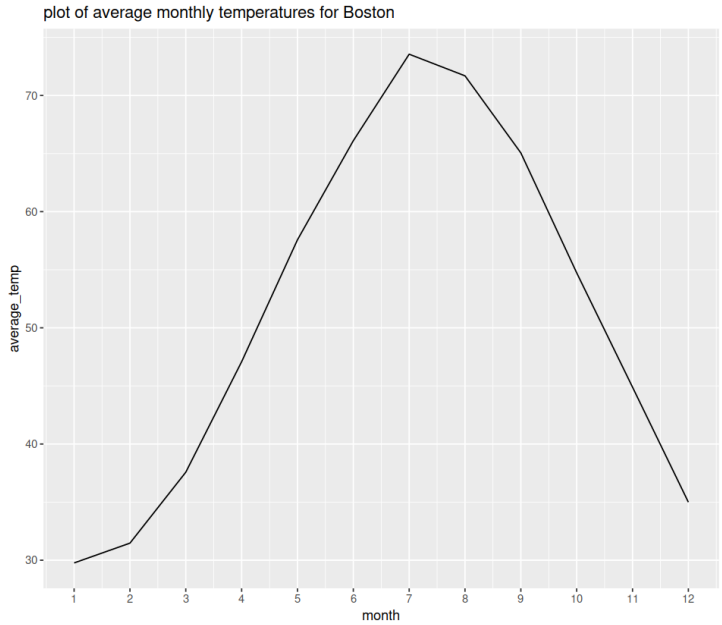


d) Filter the Boston data from usaDailyTemps and assign it to the variable bostonDailyTemps.

```
> #d
> bostonDailyTemps <- filter(usaDailyTemps, city == 'Boston')
> bostonDailyTemps
# A tibble: 7,624 × 6
   state         city   month   day  year avgtemp
   <chr>         <chr>  <int> <int> <int>   <dbl>
 1 Massachusetts Boston     1     1  1995    38.5
 2 Massachusetts Boston     1     1  1996    34.1
 3 Massachusetts Boston     1     1  1997    10
 4 Massachusetts Boston     1     1  1998    14.2
 5 Massachusetts Boston     1     1  1999    21.7
 6 Massachusetts Boston     1     1  2000    34.8
 7 Massachusetts Boston     1     1  2001    27.6
 8 Massachusetts Boston     1     1  2002    28.7
 9 Massachusetts Boston     1     1  2003    40.5
10 Massachusetts Boston     1     1  2004    40.2
# … with 7,614 more rows
# ℹ Use `print(n = ...)` to see more rows
>
```

e) What are the average monthly temperatures for Boston? Show the values and also the appropriate plot for the results. Use the bostonDailyTemps

```
> #e
> avg_temp_mon <- bostonDailyTemps %>% group_by(month) %>% summarise(average_temp = mean(avgtemp))
> avg_temp_mon
# A tibble: 12 × 2
   month average_temp
   <int>        <dbl>
 1     1         29.8
 2     2         31.5
 3     3         37.6
 4     4         47.1
 5     5         57.6
 6     6         66.1
 7     7         73.6
 8     8         71.7
 9     9         65.1
10    10         54.7
11    11         44.9
12    12         35.0
> ggplot(data = avg_temp_mon, aes(x=month, y=average_temp, group=1))+
+   geom_line()+
+   ggtitle("plot of average monthly temperatures for Boston")+
+   scale_x_continuous(breaks = (seq(min(avg_temp_mon$month), max(avg_temp_mon$month), by = 1)))
>
```


plot of average monthly temperatures for Boston
average_temp
70
60
50
40
30
1
2
3
4
5
6
7
8
9
10
11
12
month