Assignment #3

Gabe Abreu

2/15/2020

Assignment 3

Question 1

Using the 173 majors listed in fivethirtyeight.com's College Majors dataset [https://fivethirtyeight.com/features/the-economic-guide-to-picking-a-college-major/], provide code that identifies the majors that contain either "DATA" or "STATISTICS"

```
major_list <- "https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/majors-list

Majors <- read.csv(major_list)

data_majors <- as_data_frame(Majors)
```

```
## Warning: `as_data_frame()` is deprecated, use `as_tibble()` (but mind the new semantics).
## This warning is displayed once per session.
```

```
SelectedMajors <- data_majors$Major[grep("DATA|STATISTICS", data_majors$Major)]

View(SelectedMajors)

Majors %>% filter(str_detect(Major, ("DATA|STATISTICS")))
```

```
##   FOD1P                                     Major          Major_Category
## 1  6212 MANAGEMENT INFORMATION SYSTEMS AND STATISTICS                Business
## 2  2101      COMPUTER PROGRAMMING AND DATA PROCESSING Computers & Mathematics
## 3  3702               STATISTICS AND DECISION SCIENCE Computers & Mathematics
```

Question 2

#2 Write code that transforms the data below:

[1] "bell pepper" "bilberry" "blackberry" "blood orange"

[5] "blueberry" "cantaloupe" "chili pepper" "cloudberry"

[9] "elderberry" "lime" "lychee" "mulberry"

[13] "olive" "salal berry"

Into a format like this:

c("bell pepper", "bilberry", "blackberry", "blood orange", "blueberry", "cantaloupe", "chili pepper", "cloudberry", "elderberry", "lime", "lychee", "mulberry", "olive", "salal berry"

```
fruits <- c("bell pepper", "bilberry", "blackberry", "blood orange", "blueberry", "cantaloupe", "chili p

dput(as.character(fruits))
```

```
## c("bell pepper", "bilberry", "blackberry", "blood orange", "blueberry",
## "cantaloupe", "chili pepper", "cloudberry", "elderberry", "lime",
## "lychee", "mulberry", "olive", "salal berry")
```

Question 3

Describe, in words, what these expressions will match:

(.)\1\1 Answer: same character appears 3 times in a row

"(.)(.)\2\1" Answer: 2 characters attached to the same 2 characters in reverse order

(..)\1 Answer:Any 2 characters repeated

"(.).\1.\1" Answer:1 character repeated three times with a different character in between every original character occurrence "rtrsr"

"(.)(.)(.).*\3\2\1" Answer:3 characters followed by zero or more characters followed by the original 3 characters in reverse order.

Question 4

Construct regular expressions to match words that:

Start and end with the same character. Regular Expression: "^(.)((.*\1)|
1?)"

Contain a repeated pair of letters (e.g. "church" contains "ch" repeated twice.) Regular Expression: "[A-Za-z][A-Za-z]).*\1"

Contain one letter repeated in at least three places (e.g. "eleven" contains three "e"s.) Regular Expression:"([a-z].*1.*\1")

```
rando_words <- c("bell","apple","dog","eye","bob","test","mom","sense","church","banana","pepperoni","r

#A
four_a <- str_subset(rando_words, "^(.)((.*\\1$)|\\1?$)")
four_a
```

```
## [1] "eye"  "bob"  "test" "mom"  "oso"
```

```
#B
four_b <- str_subset(rando_words, "([A-Za-z][A-Za-z]).*\\1")
four_b
```

```
## [1] "sense"       "church"      "banana"      "pepperoni"   "soso"
## [6] "bandana"     "Mississippi"
```

```
#C
four_c <- str_subset(rando_words, "([A-Za-z]).*\\1.*\\1")
four_c
```

```
## [1] "banana"      "pepperoni"   "eleven"      "bandana"     "Mississippi"
## [6] "conscience"
```