HW4

Tommy Flynn

2/28/2018

GitHub Repository: https://github.com/tommyflynn/N741_Homework/tree/master/Flynn_HW_04

Problem 1.

Suppose you have another vector vText as follows:

```
vText <- c("google", "logo", "dig", "blog", "boogie")
```

You want to match `g`, `og`, `go`, or `ogo` and replace with `.`.

Write the R code that will make that happen.

Answer to Problem 1.

```
pattern <- 'o?go?'
gsub(pattern, '.', vText)
```

```
## [1] "..le"  "l."    "di."   "bl."   "bo.ie"
```

Section 2.

You have 3 strings of text that you wish to merge. One way to do this is to use the `paste` function.

```
x <- "I AM SAM. I AM SAM. SAM I AM."
y <- "THAT SAM-I-AM! THAT SAM-I-AM! I DO NOT LIKE THAT SAM-I-AM!"
z <- "DO YOU LIKE GREEN EGGS AND HAM?"
paste(x, y, z, collapse = NULL)
```

```
## [1] "I AM SAM. I AM SAM. SAM I AM. THAT SAM-I-AM! THAT SAM-I-AM! I DO NOT LIKE THAT SAM-I-AM! DO YOU
```

> Extra credit: What is the difference if you use the `paste0` function instead of the `paste` function above? `past0(x, y, z, collapse = NULL)` omits the space between character vectors.

Problem 2.

Suppose that you now have 4 lines of text as follows:

```
W <- "Hey Diddle Diddle, the cat and the fiddle,"
X <- "The cow jumped over the moon."
Y <- "The little boy laughed to see such a sport,"
Z <- "And the dish ran away with the spoon."
```

Write the R code below to merge these 3 strings.

Answer to Problem 2.

```
paste(W, X, Y, Z, collapse = NULL)
```

```
## [1] "Hey Diddle Diddle, the cat and the fiddle, The cow jumped over the moon. The little boy laughed
```

Problem 3.

We now want to concatenate our 4 vectors and NA. Do this using both methods.

```
W <- "Hey Diddle Diddle, the cat and the fiddle,"
X <- "The cow jumped over the moon."
Y <- "The little boy laughed to see such a sport,"
Z <- "And the dish ran away with the spoon."
```

Answer to Problem 3.

```
paste(W, X, Y, Z, NA, collapse = NULL)
```

```
## [1] "Hey Diddle Diddle, the cat and the fiddle, The cow jumped over the moon. The little boy laughed
```

```
str_c(W, X, Y, Z, " NA", collapse = NULL)
```

```
## [1] "Hey Diddle Diddle, the cat and the fiddle,The cow jumped over the moon.The little boy laughed t
```

Problem 4.

Suppose Melinda Higgins wants to extract the last *6* letters of her name.

```
herName <- "Melinda Higgins"
```

Write the code below to extract the last 6 letters of her name.

Answer to Problem 4.

```
Length <- str_length(herName)
last6letters <- str_sub(herName, Length-5, Length)
last6letters
```

```
## [1] "iggins"
```

Problem 5.

I want to separate the following string into separate words:

```
myNewString <- "Now_is_the_time_for_all_good_men_to_come_to_the_aid_of_their_country"
```

Split this new string into separate words:

Answer to Problem 5.

```
mySeparatedString <- str_split(myNewString, "_")
mySeparatedString
```

```
## [[1]]
##  [1] "Now"     "is"      "the"     "time"    "for"     "all"     "good"
##  [8] "men"     "to"      "come"    "to"      "the"     "aid"     "of"
## [15] "their"   "country"
```

Problem 6.

Suppose we wanted to split off the first "word" from `myNewString`. Again, we have

```
myNewString <- "Now_is_the_time_for_all_good_men_to_come_to_the_aid_of_their_country"
```

Split off the first word but leave the rest intact.

Answer 6.

```
myNewSplitSpring <- str_split(myNewString, "_", n=2)
myNewSplitSpring
```

```
## [[1]]
## [1] "Now"
## [2] "is_the_time_for_all_good_men_to_come_to_the_aid_of_their_country"
```

Problem 7.

```
yourNewString <- "Now is the time for all good men to come to the aid of their country"
```

Use the `stri_count_words` function as above to count the number of distinct words in `yourNewString`.

Answer 7.

```
stri_count_words(yourNewString)
```

```
## [1] 16
```

Problem 8.

Suppose you have string listing famouse nurses and you want to find the duplicates. Here is the list:

```
nurses <-c("Nightingale", "Barton", "Dix", "Sanger", "Barton", "Woodruff", "Lincoln", "Dix", "Peplau")
```

Answer 8.

```
nurses[stri_duplicated(nurses)]
```

```
## [1] "Barton" "Dix"
```

Problem 9.

Suppose we create the object `DIGITS` as follows:

```
DIGITS <- c("0", "1", "2", "3", "4", "5", "6", "7", "8", "9")
```

How can we form the string '0_1-2_3-4_5-6_7-8_9-'?

Answer 9.

```
stri_join(DIGITS, separators = c("_","-"), collapse = "")
```

```
## [1] "0_1-2_3-4_5-6_7-8_9-"
```

Problem 10

Using the original pun, replace `his` with `her` and `he` with `she`:

```
pun<- "A statistician can have his head in an oven and his feet in ice, and he will say that on the ave
```

Answer 10

Note that the word `head` also starts with `he` but we don't want to substitute `she` into there.

```
punModified <- stri_replace_all_fixed(pun, c("his", " he "), c("her", " she "), vectorize_all = FALSE)
punModified
```

```
## [1] "A statistician can have her head in an oven and her feet in ice, and she will say that on the a
```