Homework 07 CSCI 036 Solutions

Lucas Welch

Due: Friday, 2022-10-28

Instructions

Please box your answers. For numerical answers, this can be done using something like $\boxed{34}$. For text answers, this can be done using something like $\boxed{\text{My answer}}$. The output of a code chunk is automatically boxed, so no need to do more.

Consider the regular expression `34.[a-g]`. Which of the following strings does this pattern match?

a. `"345b"`

b. `"34b5"`

c. `"435b"`

d. `"34gb"`

A and D

Write down a regular expression that matches a digit, followed by any character, followed by the letter `a`, followed by any lowercase letter.

"[0-9].a[a-z]"

Consider the `mtcars` dataset. The `row_names_to_columns` function can be used to make the row names the first columns of data.

```
mtcars |>
  rownames_to_column() |>
  head()
```

```
##             rowname  mpg cyl disp  hp drat    wt  qsec vs am gear carb
## 1         Mazda RX4 21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## 2     Mazda RX4 Wag 21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## 3        Datsun 710 22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## 4    Hornet 4 Drive 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## 5 Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## 6           Valiant 18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

a. Using `str_detect` and `filter`, find all the observations with car names that start with `"Merc"`.

b. Using `str_detect` and `filter`, find all the observations with car names that have at least one digit in them.

a.

```
mtcars |>
  rownames_to_column() |>
  filter(str_detect(rowname, "^Merc"))
```

```
##       rowname  mpg cyl  disp  hp drat   wt qsec vs am gear carb
## 1   Merc 240D 24.4   4 146.7  62 3.69 3.19 20.0  1  0    4    2
## 2    Merc 230 22.8   4 140.8  95 3.92 3.15 22.9  1  0    4    2
## 3    Merc 280 19.2   6 167.6 123 3.92 3.44 18.3  1  0    4    4
## 4   Merc 280C 17.8   6 167.6 123 3.92 3.44 18.9  1  0    4    4
## 5  Merc 450SE 16.4   8 275.8 180 3.07 4.07 17.4  0  0    3    3
## 6  Merc 450SL 17.3   8 275.8 180 3.07 3.73 17.6  0  0    3    3
## 7 Merc 450SLC 15.2   8 275.8 180 3.07 3.78 18.0  0  0    3    3
```

b.

```
mtcars |>
  rownames_to_column() |>
  filter(str_detect(rowname, "[0-9]"))
```

```
##           rowname  mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## 1       Mazda RX4 21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
## 2   Mazda RX4 Wag 21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
## 3      Datsun 710 22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## 4  Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
## 5      Duster 360 14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## 6       Merc 240D 24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
## 7        Merc 230 22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
## 8        Merc 280 19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## 9       Merc 280C 17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## 10     Merc 450SE 16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## 11     Merc 450SL 17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## 12    Merc 450SLC 15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## 13       Fiat 128 32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## 14     Camaro Z28 13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## 15      Fiat X1-9 27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
## 16  Porsche 914-2 26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## 17     Volvo 142E 21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

Consider the `USArrests` dataset.

a. Use `str_detect` and `filter` to find all the observations from states that begin with the letter `A`.

b. Use `str_detect` and `filter` to find all the observations from states that end with the letter `a`.

c. Use `str_detect` and `filter` to find all the observations from states that begins with `A` and ends with `a`.

```
USArrests
```

```
##                Murder Assault UrbanPop Rape
## Alabama          13.2     236       58 21.2
## Alaska           10.0     263       48 44.5
## Arizona           8.1     294       80 31.0
## Arkansas          8.8     190       50 19.5
## California        9.0     276       91 40.6
## Colorado          7.9     204       78 38.7
## Connecticut       3.3     110       77 11.1
## Delaware          5.9     238       72 15.8
## Florida          15.4     335       80 31.9
## Georgia          17.4     211       60 25.8
## Hawaii            5.3      46       83 20.2
## Idaho             2.6     120       54 14.2
## Illinois         10.4     249       83 24.0
## Indiana           7.2     113       65 21.0
## Iowa              2.2      56       57 11.3
## Kansas            6.0     115       66 18.0
## Kentucky          9.7     109       52 16.3
## Louisiana        15.4     249       66 22.2
## Maine             2.1      83       51  7.8
## Maryland         11.3     300       67 27.8
## Massachusetts     4.4     149       85 16.3
## Michigan         12.1     255       74 35.1
## Minnesota         2.7      72       66 14.9
## Mississippi      16.1     259       44 17.1
## Missouri          9.0     178       70 28.2
## Montana           6.0     109       53 16.4
## Nebraska          4.3     102       62 16.5
## Nevada           12.2     252       81 46.0
## New Hampshire     2.1      57       56  9.5
## New Jersey        7.4     159       89 18.8
## New Mexico       11.4     285       70 32.1
## New York         11.1     254       86 26.1
## North Carolina   13.0     337       45 16.1
## North Dakota      0.8      45       44  7.3
## Ohio              7.3     120       75 21.4
## Oklahoma          6.6     151       68 20.0
## Oregon            4.9     159       67 29.3
## Pennsylvania      6.3     106       72 14.9
## Rhode Island      3.4     174       87  8.3
## South Carolina   14.4     279       48 22.5
## South Dakota      3.8      86       45 12.8
## Tennessee        13.2     188       59 26.9
## Texas            12.7     201       80 25.5
## Utah              3.2     120       80 22.9
## Vermont           2.2      48       32 11.2
## Virginia          8.5     156       63 20.7
## Washington        4.0     145       73 26.2
## West Virginia     5.7      81       39  9.3
## Wisconsin         2.6      53       66 10.8
## Wyoming           6.8     161       60 15.6
```

a.

```
USArrests |>
  rownames_to_column() |>
  filter(str_detect(rowname, "^A"))
```

```
##    rowname Murder Assault UrbanPop Rape
## 1  Alabama   13.2     236       58 21.2
## 2   Alaska   10.0     263       48 44.5
## 3  Arizona    8.1     294       80 31.0
## 4 Arkansas    8.8     190       50 19.5
```

b.

```
USArrests |>
  rownames_to_column() |>
  filter(str_detect(rowname, "a$"))
```

```
##           rowname Murder Assault UrbanPop Rape
## 1         Alabama   13.2     236       58 21.2
## 2          Alaska   10.0     263       48 44.5
## 3         Arizona    8.1     294       80 31.0
## 4      California    9.0     276       91 40.6
## 5         Florida   15.4     335       80 31.9
## 6         Georgia   17.4     211       60 25.8
## 7         Indiana    7.2     113       65 21.0
## 8            Iowa    2.2      56       57 11.3
## 9       Louisiana   15.4     249       66 22.2
## 10      Minnesota    2.7      72       66 14.9
## 11        Montana    6.0     109       53 16.4
## 12       Nebraska    4.3     102       62 16.5
## 13         Nevada   12.2     252       81 46.0
## 14 North Carolina   13.0     337       45 16.1
## 15   North Dakota    0.8      45       44  7.3
## 16       Oklahoma    6.6     151       68 20.0
## 17   Pennsylvania    6.3     106       72 14.9
## 18 South Carolina   14.4     279       48 22.5
## 19   South Dakota    3.8      86       45 12.8
## 20       Virginia    8.5     156       63 20.7
## 21  West Virginia    5.7      81       39  9.3
```

c.

```
USArrests |>
  rownames_to_column() |>
  filter(str_detect(rowname, "^A")) |>
  filter(str_detect(rowname, "a$"))
```

```
##   rowname Murder Assault UrbanPop Rape
## 1 Alabama   13.2     236       58 21.2
## 2  Alaska   10.0     263       48 44.5
## 3 Arizona    8.1     294       80 31.0
```

Consider the following code.

```
test <- c("Uranium Fever", "New Moon", "Ain't that a kick in the head")
str_extract(test, "[a-zA-Z]+$")
```

```
## [1] "Fever" "Moon"  "head"
```

```
str_extract(test, "^[a-zA-Z]+")
```

```
## [1] "Uranium" "New"     "Ain"
```

```
str_extract(test, "[^a-zA-Z]+")
```

```
## [1] " " " " " "'"
```

a. Describe in words what the regular expression `[a-zA-Z]+$` is trying to find.

b. Describe in words what the regular expression `^[a-zA-Z]+` is trying to find.

c. Describe in words what the regular expression `[^a-zA-Z]+` is trying to find.

a. \boxed{\text{It's extracting letters(a-z(A-Z)) from code after the first word then adding the second word since the $ }}

b. It's extracting letters(a-zA-Z) from the first words since the ^

c. It's extracting any letters from the whole line because the ^ is in the brackets :::: {.solution data-latex=“”}

::::

Consider the dataset `painters` from the `MASS` library. First, load in the library.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

Then the dataset looks as follows.

```
painters |> head()
```

```
##               Composition Drawing Colour Expression School
## Da Udine               10       8     16          3      A
## Da Vinci               15      16      4         14      A
## Del Piombo              8      13     16          7      A
## Del Sarto              12      16      9          8      A
## Fr. Penni               0      15      8          0      A
## Guilio Romano          15      16      4         14      A
```

a. Find the observations where the painter name has at least one space in it.

b. Using your tibble from part a., create a column `after_space` which lists the part of the painters name after the last space in the name.

```
painters
```

```
##                 Composition Drawing Colour Expression School
## Da Udine                 10       8     16          3      A
## Da Vinci                 15      16      4         14      A
## Del Piombo                8      13     16          7      A
## Del Sarto                12      16      9          8      A
## Fr. Penni                 0      15      8          0      A
## Guilio Romano            15      16      4         14      A
## Michelangelo              8      17      4          8      A
## Perino del Vaga          15      16      7          6      A
## Perugino                  4      12     10          4      A
## Raphael                  17      18     12         18      A
## F. Zucarro               10      13      8          8      B
## Fr. Salviata             13      15      8          8      B
## Parmigiano               10      15      6          6      B
## Primaticcio              15      14      7         10      B
## T. Zucarro               13      14     10          9      B
## Volterra                 12      15      5          8      B
## Barocci                  14      15      6         10      C
## Cortona                  16      14     12          6      C
## Josepin                  10      10      6          2      C
## L. Jordaens              13      12      9          6      C
## Testa                    11      15      0          6      C
## Vanius                   15      15     12         13      C
## Bassano                   6       8     17          0      D
## Bellini                   4       6     14          0      D
## Giorgione                 8       9     18          4      D
## Murillo                   6       8     15          4      D
## Palma Giovane            12       9     14          6      D
## Palma Vecchio             5       6     16          0      D
## Pordenone                 8      14     17          5      D
## Tintoretto               15      14     16          4      D
## Titian                   12      15     18          6      D
## Veronese                 15      10     16          3      D
## Albani                   14      14     10          6      E
## Caravaggio                6       6     16          0      E
## Corregio                 13      13     15         12      E
## Domenichino              15      17      9         17      E
## Guercino                 18      10     10          4      E
## Lanfranco                14      13     10          5      E
## The Carraci              15      17     13         13      E
## Durer                     8      10     10          8      F
## Holbein                   9      10     16         13      F
## Pourbus                   4      15      6          6      F
## Van Leyden                8       6      6          4      F
## Diepenbeck               11      10     14          6      G
## J. Jordaens              10       8     16          6      G
## Otho Venius              13      14     10         10      G
## Rembrandt                15       6     17         12      G
## Rubens                   18      13     17         17      G
## Teniers                  15      12     13          6      G
## Van Dyck                 15      10     17         13      G
## Bourdon                  10       8      8          4      H
```

```
## Le Brun                  16      16       8         16      H
## Le Suer                  15      15       4         15      H
## Poussin                  15      17       6         15      H
```

a.

```
painters |>
  rownames_to_column() |>
  filter(str_detect(rowname, " "))
```

```
##            rowname Composition Drawing Colour Expression School
## 1         Da Udine          10       8     16          3      A
## 2         Da Vinci          15      16      4         14      A
## 3       Del Piombo           8      13     16          7      A
## 4        Del Sarto          12      16      9          8      A
## 5        Fr. Penni           0      15      8          0      A
## 6    Guilio Romano          15      16      4         14      A
## 7  Perino del Vaga          15      16      7          6      A
## 8       F. Zucarro          10      13      8          8      B
## 9     Fr. Salviata          13      15      8          8      B
## 10      T. Zucarro          13      14     10          9      B
## 11     L. Jordaens          13      12      9          6      C
## 12   Palma Giovane          12       9     14          6      D
## 13   Palma Vecchio           5       6     16          0      D
## 14     The Carraci          15      17     13         13      E
## 15      Van Leyden           8       6      6          4      F
## 16     J. Jordaens          10       8     16          6      G
## 17     Otho Venius          13      14     10         10      G
## 18        Van Dyck          15      10     17         13      G
## 19         Le Brun          16      16      8         16      H
## 20         Le Suer          15      15      4         15      H
```

b.

```
painters |>
  rownames_to_column() |>
  filter(str_detect(rowname, " ")) |>
  mutate(after_space = str_extract(rowname, "[a-zA-Z]+$"))
```

```
##            rowname Composition Drawing Colour Expression School after_space
## 1         Da Udine          10       8     16          3      A       Udine
## 2         Da Vinci          15      16      4         14      A       Vinci
## 3       Del Piombo           8      13     16          7      A      Piombo
## 4        Del Sarto          12      16      9          8      A       Sarto
## 5        Fr. Penni           0      15      8          0      A       Penni
## 6    Guilio Romano          15      16      4         14      A      Romano
## 7  Perino del Vaga          15      16      7          6      A        Vaga
## 8       F. Zucarro          10      13      8          8      B     Zucarro
## 9     Fr. Salviata          13      15      8          8      B    Salviata
## 10      T. Zucarro          13      14     10          9      B     Zucarro
## 11     L. Jordaens          13      12      9          6      C    Jordaens
## 12   Palma Giovane          12       9     14          6      D     Giovane
## 13   Palma Vecchio           5       6     16          0      D     Vecchio
## 14     The Carraci          15      17     13         13      E     Carraci
## 15      Van Leyden           8       6      6          4      F      Leyden
## 16     J. Jordaens          10       8     16          6      G    Jordaens
## 17     Otho Venius          13      14     10         10      G      Venius
## 18        Van Dyck          15      10     17         13      G        Dyck
## 19         Le Brun          16      16      8         16      H        Brun
## 20         Le Suer          15      15      4         15      H        Suer
```

Suppose you have a vector of strings in `s`.

a. Write code to detect for each string in the vector if it contains the letter `a` followed by a digit at least once.

b. Write code that matches the pattern given in part a, but instead of returning true or false, returns the two characters that match the pattern if it exists in the string.

a.

```
t <- c("aww", "a43", "hfi2")
s_detect <- str_detect(t, "a[0-9]")
s_detect
```

```
## [1] FALSE  TRUE FALSE
```

b.

```
t <- c("jdne", "a898", "vdw3")
s_extract <- str_extract(t, "a[0-9]")
s_extract
```

```
## [1] NA   "a8" NA
```

Consider a tibble:

```
phone_numbers <-
  tribble(
  ~Name, ~Phone,
  "Mr. Plow", "636-555-3226",
  "Big Mean Carla", "323-555-0129",
  "Elmo", "212-555-6666",
  "Phoenix Biogenics", "225-330-7040"
)
```

The *area code* for a U.S. phone number consists of the first three digits.

a. Write code to add a column `area_code` to the tibble which consists of the first three digits of the phone number.

b. Write code to filter `phone_numbers` to only include numbers where the middle three digits are `555`.

a.

```
phone_numbers |>
  mutate(area_code = str_extract(Phone, "^[0-9][0-9][0-9]"))
```

```
## # A tibble: 4 × 3
##   Name              Phone        area_code
##   <chr>             <chr>        <chr>
## 1 Mr. Plow          636-555-3226 636
## 2 Big Mean Carla    323-555-0129 323
## 3 Elmo              212-555-6666 212
## 4 Phoenix Biogenics 225-330-7040 225
```

b.

```
phone_numbers |>
  filter(str_detect(Phone,"-555-"))
```

```
## # A tibble: 3 × 2
##   Name           Phone
##   <chr>          <chr>
## 1 Mr. Plow       636-555-3226
## 2 Big Mean Carla 323-555-0129
## 3 Elmo           212-555-6666
```

a. Give a regular expression that matches one or more digits at the beginning of a string.

b. Give a regular expression that matches zero or more digits at the end of a string.

a. `"^[0-9]+"`

b. `"[0-9]*$"`

Consider the following tibble.

```
data <-
  tibble(
    s = c("blue141car", "red314159truck", "yellow2718airplane")
  )
```

Using the `separate` command, use a regular expression for the `sep` parameter to separate the `s` variable into `before_number` and `after_number`.

```
data
```

```
## # A tibble: 3 × 1
##   s
##   <chr>
## 1 blue141car
## 2 red314159truck
## 3 yellow2718airplane
```

```
data |>
  separate(s, into = c("before_number", "after_number"), sep = ("[0-9]+"))
```

```
## # A tibble: 3 × 2
##   before_number after_number
##   <chr>         <chr>
## 1 blue          car
## 2 red           truck
## 3 yellow        airplane
```