Advent of Code 2020

Zach Bogart

Day 3: Toboggan Trajectory

https://adventofcode.com/2020/day/3

Part 1

- Input, going **right 3 down 1** every row
- Given patterns repeat

```
raw = read_delim("inputs/03-input.txt", delim="\n", col_names = "text")
```

How long is every string?

```
raw %>%
  rowwise() %>%
  mutate(length = nchar(text))
```

```
## # A tibble: 323 x 2
## # Rowwise:
##    text                            length
##    <chr>                            <int>
##  1 ...........#....#...........         31
##  2 ..........##....#......#..#..#       31
##  3 ......#.......#......#.........      31
##  4 ..#.#....#....#............##.       31
##  5 ..#........####....#...#.......      31
##  6 ..##.....#.#.#..#........#....       31
##  7 ...#.#..#..#....#..#..#........      31
##  8 #.......#.........#....##.###..      31
##  9 ......##..#.#...#......#.#....       31
## 10 ...............##.........#.##       31
## # ... with 313 more rows
```

Every string is 31 characters long. There are 323 rows.

- For n rows, need each string to be $3(n-1)+1$ characters long, or $3n-2$
- Since there are 323 rows, n = 323
- So, every string needs to be 967 characters long

Can repeat each string 32 times to allow for enough input

```
repeat_1 = ceiling((3 * nrow(raw) - 2) / 31)
```

```
input = raw %>%
  mutate(text = strrep(text, repeat_1),
         row = row_number(),
```

```
        index_value = 3 * row - 2,
        hit_a_tree = (str_sub(text, index_value, index_value) == "#"))

sum(input$hit_a_tree)
```

```
## [1] 171
```

We hit 171 trees.

Part 2

- add more slopes this time!

- Right 1, down 1.

- Right 3, down 1. (This is the slope you already checked.)

- Right 5, down 1.

- Right 7, down 1.

- Right 1, down 2.

Should all be straight forward except a little filtering for the down 2 example. Will need to repeat the string a few more times.

- 7(n-1)+1 or 7n-6
- Every string needs to be 2255 character long
- Need to repeat each string 73 times (each string 31 chars long)

```
repeat_2 = ceiling((7 * nrow(raw) - 6) / 31)
```

```
slope_1 = raw %>%
  mutate(text = strrep(text, repeat_2),
         row = row_number(),

         right_1 = 1 * (row-1) + 1,
         right_3 = 3 * (row-1) + 1,
         right_5 = 5 * (row-1) + 1,
         right_7 = 7 * (row-1) + 1,

         hit_1 = (str_sub(text, right_1, right_1) == "#"),
         hit_3 = (str_sub(text, right_3, right_3) == "#"),
         hit_5 = (str_sub(text, right_5, right_5) == "#"),
         hit_7 = (str_sub(text, right_7, right_7) == "#"))

# the down 1 slopes
down_1 = slope_1 %>%
  select(hit_1:hit_7) %>%
  summarise_all(sum)

# the down 2 slopes
# have to make sure to filter by index correctly! Don't define row value till after!
slope_2 = raw %>%
  filter(row_number() %% 2 == 1) %>%
  mutate(text = strrep(text, 73),
         row = row_number()) %>%
```

```
  mutate(right_1 = 1 * (row-1) + 1,
         hit_1 = (str_sub(text, right_1, right_1) == "#"))

down_2 = slope_2 %>%
  select(hit_1) %>%
  summarise(hit_1_down_2 = sum(hit_1))

results = cbind(down_1, down_2)

results
```

```
##   hit_1 hit_3 hit_5 hit_7 hit_1_down_2
## 1    70   171    48    60           35
```

Above are how many trees we hit for each slope. So, their product is. . .

```
results %>%
  rowwise() %>%
  prod()
```

```
## [1] 1206576000
```

1206576000

:)