Advent of Code 2020

Zach Bogart

2020-12-12

Day 10: Adapter Array

Click for Problem Statement

```
testing1 = tibble(x = as.numeric(readLines("inputs/10-test1.txt")))
testing2 = tibble(x = as.numeric(readLines("inputs/10-test2.txt")))
input = tibble(x = as.numeric(read_lines("inputs/10-input.txt")))
```

Part 1

Testing

```
t2_p1 = testing2 %>%
  add_row(x = 0) %>% # charging outlet
  add_row(x = max(.$x, na.rm=TRUE) + 3) %>%  # end adapter
  arrange(x) %>%
  mutate(diff = x - lag(x)) %>%
  mutate(j1 = diff == 1,
         j3 = diff == 3)

t2_p1 %>%
  summarise(total_j1 = sum(j1, na.rm = TRUE),
            total_j3 = sum(j3, na.rm = TRUE),
            answer = total_j1 * total_j3)
```

```
## # A tibble: 1 x 3
##   total_j1 total_j3 answer
##      <int>    <int>  <int>
## 1       22       10    220
```

The Real Thing

```
part1 = input %>%
  add_row(x = 0) %>% # charging outlet
  add_row(x = max(.$x, na.rm=TRUE) + 3) %>%  # end adapter
  arrange(x) %>%
  mutate(diff = x - lag(x)) %>%
  mutate(j1 = diff == 1,
         j3 = diff == 3)

part1 %>%
  summarise(total_j1 = sum(j1, na.rm = TRUE),
            total_j3 = sum(j3, na.rm = TRUE),
            answer = total_j1 * total_j3)
```

```
## # A tibble: 1 x 3
##   total_j1 total_j3 answer
##      <int>    <int>  <int>
## 1       69       30   2070
```

Part 2

Testing

Doing some working out, seems to be that each additional adapter in a string of 1-jolts gives a multiple of three more options.

- need to find strings of ones, count size of string, and multiply results based on rules
- One 1-jolt provides only one option
- Two 1-jolts provide only two options
- For `n >= 3` 1-jolts, they provide `3(n-2) + 1` options

```
t2_p2 = t2_p1 %>%
  filter(!is.na(diff)) %>%
  mutate(one_string = cumsum(diff == 3)) %>%
  filter(diff != 3) %>%
  group_by(one_string) %>%
  summarise(options = ifelse(n() == 1, 1, ifelse(n() == 2, 2, 3*(n()-2) + 1)))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
t2_p2 %>%
  summarise(answer = prod(options))
```

```
## # A tibble: 1 x 1
##   answer
##    <dbl>
## 1  19208
```

The Real Thing

```
options(scipen=30)

part2 = part1 %>%
  filter(!is.na(diff)) %>%
  mutate(one_string = cumsum(diff == 3)) %>%
  filter(diff != 3) %>%
  group_by(one_string) %>%
  summarise(options = ifelse(n() == 1, 1, ifelse(n() == 2, 2, 3*(n()-2) + 1)))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
part2 %>%
  summarise(answer = format(prod(options), big.mark = ","))
```

```
## # A tibble: 1 x 1
##   answer
##   <chr>
## 1 24,179,327,893,504
```