

Advent Of Code 2022

```
library(tidyverse)
```

Day 1

Part 1

Find the Elf carrying the most Calories. How many total Calories is that Elf carrying?

```
raw_data <- read_csv("day1.txt", col_names = FALSE, skip_empty_rows = FALSE)
data <- c()
cumsum <- 0
for (v in raw_data$X1) {
  if (is.na(v)) {
    data <- append(data, cumsum)
    cumsum <- 0
  } else {
    cumsum <- cumsum + v
  }
}

max_cal <- 0
for (v in data) {
  if (v > max_cal) {
    max_cal <- v
  }
}

max_cal
```

```
## [1] 72070
```

Part 2

Find the top three Elves carrying the most Calories. How many Calories are those Elves carrying in total?

```
max_cals <- c(0, 0, 0)
for (v in data) {
  for (i in 1:3) {
    if (v > max_cals[i]) {
      min_index <- 1
      if (max_cals[2] < max_cals[min_index]) {
        min_index <- 2
      }
      if (max_cals[3] < max_cals[min_index]) {
        min_index <- 3
      }
      max_cals[min_index] <- v
      break
    }
  }
}
```

```

    }
  }
}

sum(max_cals)

```

```
## [1] 211805
```

Day 2

Part 1

What would your total score be if everything goes exactly according to your strategy guide?

```

data <- read_delim("day2.txt", col_names = c("You", "Me"), delim = " ") %>%
  mutate(
    You = case_when(
      You == "A" ~ "R",
      You == "B" ~ "P",
      You == "C" ~ "S"
    ),
    Me = case_when(
      Me == "X" ~ "R",
      Me == "Y" ~ "P",
      Me == "Z" ~ "S"
    ),
    ShapeScore = case_when(
      Me == "R" ~ 1,
      Me == "P" ~ 2,
      Me == "S" ~ 3
    ),
    OutcomeScore = case_when(
      Me == You ~ 3,
      Me == "R" & You == "S" ~ 6,
      Me == "P" & You == "R" ~ 6,
      Me == "S" & You == "P" ~ 6,
      TRUE ~ 0
    ),
    Score = ShapeScore + OutcomeScore
  )

sum(data$Score)

```

```
## [1] 12276
```

Part 2

Following the Elf's instructions for the second column, what would your total score be if everything goes exactly according to your strategy guide?

```

data <- read_delim("day2.txt", col_names = c("You", "Outcome"), delim = " ") %>%
  mutate(
    You = case_when(
      You == "A" ~ "R",
      You == "B" ~ "P",
      You == "C" ~ "S"
    ),
    Outcome = case_when(
      Outcome == "X" ~ "Lose",
      Outcome == "Y" ~ "Draw",
      Outcome == "Z" ~ "Win"
    ),
    OutcomeScore = case_when(
      Outcome == "Lose" ~ 0,
      Outcome == "Draw" ~ 3,
      Outcome == "Win" ~ 6
    ),
    Me = case_when(
      Outcome == "Draw" ~ You,
      Outcome == "Win" & You == "R" ~ "P",
      Outcome == "Win" & You == "P" ~ "S",
      Outcome == "Win" & You == "S" ~ "R",
      Outcome == "Lose" & You == "R" ~ "S",
      Outcome == "Lose" & You == "P" ~ "R",
      Outcome == "Lose" & You == "S" ~ "P",
    ),
    ShapeScore = case_when(
      Me == "R" ~ 1,
      Me == "P" ~ 2,
      Me == "S" ~ 3
    ),
    Score = ShapeScore + OutcomeScore
  )

sum(data$Score)

```

```
## [1] 9975
```

Day 3

Part 1

Find the item type that appears in both compartments of each rucksack. What is the sum of the priorities of those item types?

```

charToPriority <- function(char) {
  ascii <- utf8ToInt(char)
  if (ascii > 90) {
    return(ascii - 96)
  } else {
    return(ascii - 38)
  }
}

```

```

word2vec <- function(word) {
  return(strsplit(word, " ")[[1]])
}

common_item.rucksack <- function(items) {
  left <- substr(items, 1, nchar(items) / 2) # raw string
  left <- word2vec(left) # vector of chars

  right <- substr(items, (nchar(items) / 2) + 1, nchar(items)) # raw string
  right <- word2vec(right) # vector of chars

  for (c in left) {
    if (c %in% right) {
      return(c)
    }
  }

  return('')
}

charToPriority <- function(char) {
  ascii <- utf8ToInt(char)
  if (ascii > 90) {
    return(ascii - 96)
  } else {
    return(ascii - 38)
  }
}

data <- read_csv("day3.txt", col_names = c("Items")) %>%
  mutate(
    CommonItem = sapply(Items, common_item.rucksack),
    Priority = sapply(CommonItem, charToPriority)
  )

sum(data$Priority)

```

```
## [1] 8401
```

Part 2

Find the item type that corresponds to the badges of each three-Elf group. What is the sum of the priorities of those item types?

```

common_item.team <- function(items) {
  items <- strsplit(items, " ")[[1]] # list of list
  rucksack.1 <- word2vec(items[1])
  rucksack.2 <- word2vec(items[2])
  rucksack.3 <- word2vec(items[3])
  for (i in 1:3) {
    rucksack <- word2vec(items[i])
    for (c in rucksack) {

```

```

    if (c %in% rucksack.1 & c %in% rucksack.2 & c %in% rucksack.3) {
      return(c)
    }
  }
}
return('')
}

data <- read_csv("day3.txt", col_names = c("Items")) %>%
  mutate(TeamId = rep(1:(length(Items) / 3), each = 3)) %>%
  group_by(TeamId) %>%
  summarise(Items = str_c(Items, collapse = " ")) %>%
  mutate(
    CommonItem = sapply(Items, common_item.team),
    Priority = sapply(CommonItem, charToPriority)
  )

sum(data$Priority)

```

```
## [1] 2641
```

Day 4

Part 1

In how many assignment pairs does one range fully contain the other?

```

lower_range <- function(r) {
  as.numeric(str_split(r, "-")[1][1])
}

upper_range <- function(r) {
  as.numeric(str_split(r, "-")[1][2])
}

data <- read_csv("day4.txt", col_names = c("Elf1", "Elf2")) %>%
  mutate(
    Elf1Start = sapply(Elf1, lower_range),
    Elf1End = sapply(Elf1, upper_range),
    Elf2Start = sapply(Elf2, lower_range),
    Elf2End = sapply(Elf2, upper_range),
    OneContainsAnother =
      (Elf1Start >= Elf2Start & Elf1End <= Elf2End) |
      (Elf2Start >= Elf1Start & Elf2End <= Elf1End)
  )

sum(data$OneContainsAnother)

```

```
## [1] 571
```

Part 2

In how many assignment pairs do the ranges overlap?

```
data <- read_csv("day4.txt", col_names = c("Elf1", "Elf2")) %>%
  mutate(
    Elf1Start = sapply(Elf1, lower_range),
    Elf1End = sapply(Elf1, upper_range),
    Elf2Start = sapply(Elf2, lower_range),
    Elf2End = sapply(Elf2, upper_range),
    Overlap =
      (Elf2Start >= Elf1Start & Elf1End >= Elf2Start) |
      (Elf1Start >= Elf2Start & Elf2End >= Elf1Start)
  )

sum(data$Overlap)
```

```
## [1] 917
```