letterboxd

Maxwell Chu 405962883

2024-10-01

```
library(rvest)

start_time <- Sys.time()

words <- read.table("https://people.sc.fsu.edu/~jburkardt/datasets/words/anagram_dictionary.txt")
words <- tolower(words[[1]]) # converts df to a vector

html <- read_html("https://letterboxed.aliceyliang.com/populate")
string <- strsplit(as.character(html), "")[[1]]

CUSTOM_LETTERS <- tolower(paste(c(string[125:127], string[131:133], string[137:139], string[143:145]), c
lets <- matrix(strsplit(CUSTOM_LETTERS, "")[[1]], 4, byrow = TRUE)
lets_vec <- lets
attributes(lets_vec) <- NULL

make_visual <- function(lets) {
  visual <- matrix(rep("", 25), 5)
  visual[1, 2:4] <- lets[1, ]
  visual[2:4, 5] <- lets[2, ]
  visual[5, 2:4] <- lets[3, ]
  visual[2:4, 1] <- lets[4, ]
  noquote(visual)
}

message("TODAY'S LETTERS")
```

```
## TODAY'S LETTERS
```

```
make_visual(lets)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]      d    t    h
## [2,] w                   o
## [3,] r                   f
## [4,] i                   s
## [5,]      b    a    n
```

```
message("\nBEST SOLUTIONS")
```

```
##
## BEST SOLUTIONS
```

```
check_spellable <- function(word) {

  is_spellable <- TRUE
  split_word <- strsplit(word, "")[[1]]

  if(nchar(word) < 3) {
    is_spellable <- FALSE
  } else if(!all(split_word %in% lets_vec)) {
    is_spellable <- FALSE
  } else {
    for(i in seq_len(4)) {
      if(sum(diff(which(lets[i, ][1] == split_word | lets[i, ][2] == split_word | lets[i, ][3] == split_
        is_spellable <- FALSE
      }
    }
  }

  is_spellable
}

for(i in seq_along(words)) {
  if(!check_spellable(words[i]))
    words[i] <- ""
}

spellable_words <- words[words != ""]
one_word_solve <- character(0)

for(i in seq_along(spellable_words)) {
  if(all(lets_vec %in% strsplit(spellable_words[i], "")[[1]])) {
    one_word_solve <- c(one_word_solve, spellable_words[i])
  }
}

if(length(one_word_solve) != 0) {
  message("ONE WORD SOLVE!")
  noquote(one_word_solve)
} else {

  word_lengths <- vapply(spellable_words, nchar, numeric(1), USE.NAMES = FALSE)
  spellable_words <- spellable_words[rev(order(word_lengths))]

  second_pass <- function(first_word_split) {

    lets_remaining <- lets_vec[!(lets_vec %in% first_word_split)]
    spellable_words_second <- spellable_words

    for(i in seq_along(spellable_words)) {
      second_word_split <- strsplit(spellable_words[i], "")[[1]]
      if(first_word_split[length(first_word_split)] != second_word_split[1] || !all(strsplit(lets_remai
        spellable_words_second[i] <- ""
    }
```

```
    spellable_words_second <- spellable_words_second[spellable_words_second != ""]
    spellable_words_second
  }

  result_pairs_first <- character(0)
  result_pairs_second <- character(0)

  for(i in seq_along(spellable_words)) {
    first_word_split <- strsplit(spellable_words[i], "")[[1]]
    spellable_words_second <- second_pass(first_word_split)
    if(length(spellable_words_second != 0)) {
      for(j in seq_along(spellable_words_second)) {
        result_pairs_first <- c(result_pairs_first, spellable_words[i])
        result_pairs_second <- c(result_pairs_second, spellable_words_second[j])
      }
    }
  }

  result_pairs <- cbind(result_pairs_first, result_pairs_second)
  noquote(result_pairs)
}
```

```
##       result_pairs_first result_pairs_second
##  [1,] strawboard         downshifts
##  [2,] strawboard         downshift
##  [3,] downshifts         strawboard
##  [4,] downshifts         starboard
##  [5,] broadsword         downshifts
##  [6,] broadsword         downshift
##  [7,] washboard          downshifts
##  [8,] washboard          downshift
##  [9,] starboard          downshifts
## [10,] starboard          downshift
## [11,] dashboard          downshifts
## [12,] dashboard          downshift
## [13,] indrafts           showboats
## [14,] indrafts           showboat
## [15,] fashions           strawboard
## [16,] obtains            swordfish
## [17,] broad              downshifts
## [18,] broad              downshift
## [19,] braid              downshifts
## [20,] braid              downshift
## [21,] board              downshifts
## [22,] board              downshift
## [23,] brad               downshifts
## [24,] brad               downshift
```

```
message("\nIt took ", round(as.numeric(Sys.time() - start_time), 3), " seconds to run this program")
```

```
##
## It took 3.777 seconds to run this program
```