

NYT Spelling Bee Solver

C-T Berezin

11/9/21

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.5    v purrr  0.3.4
## v tibble  3.1.5    v dplyr  1.0.7
## v tidyr   1.1.4    v stringr 1.4.0
## v readr   2.0.2    v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

#all english words
word_list <- read_csv(paste0("https://raw.githubusercontent.com/dwyl",
                             "/english-words/master/words_alpha.txt"),
                      col_names = "word")

## Rows: 370103 Columns: 1

## -- Column specification -----
## Delimiter: ","
## chr (1): word

##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

#adding words not in list
word_list <- word_list %>% add_row(word=c("acai", "cannellini",
                                           "videoed", "vidiot",
                                           "binging", "bonobo",
                                           "noob", "cami", "matcha",
                                           "tamari", "wiki"))

word_list <- word_list %>%
  arrange(word) %>%
  mutate(length = str_length(word)) %>%
  filter(length >= 4)
```

#words to remove, weren't accepted in puzzle

```
nonwords <- c("fluoroformol", "fluoroform", "floramour", "floramor", "formular",
  "furfural", "furfurol", "affloof", "famular", "farfara", "florula",
  "formful", "mafurra", "alfuro", "fallal", "farfal", "faroff",
  "forfar", "formol", "fulful", "fullam", "fullom", "fulmar", "furfur",
  "mafura", "afara", "afoam", "alfur", "falla", "flaff", "fluor", "flurr",
  "foram", "forma", "forra", "froom", "fural", "luffa", "mafoo", "affa",
  "alfa", "faff", "fala", "fama", "farl", "faro", "flam", "flor", "foll",
  "fram", "frau", "fuff", "fula", "loof", "luff", "moff", "olaf", "olof",
  "raff", "ralf", "rolf", "acacian", "acaciin", "acacin", "acacine", "acaena",
  "acale", "acana", "acca", "acce", "accel", "accinge", "accinging", "acclinal",
  "acini", "acinic", "acle", "aclinal", "aclinic", "aecia", "aecial", "agacella",
  "aganice", "aglace", "alangiaceae", "alca", "alcae", "alcaic", "alcali", "alcanna",
  "alce", "alcine", "alec", "algic", "alginic", "alice", "alicia", "allelic",
  "alliaceae", "alliancing", "allice", "allicin", "anaclinal", "analgie", "ancien",
  "ancience", "ancile", "ancilia", "ancilla", "ancillae", "ancille", "angle", "anelace",
  "angelical", "angelican", "angelicic", "anglic", "anglican", "anglice", "anicca",
  "anice", "anilic", "anlace", "caca", "cacalia", "cacan", "cacana", "caccia", "cace",
  "cecei", "caci", "caeca", "caecal", "caecilia", "caeciliae", "caecilian", "caelian",
  "cageling", "cagn", "caic", "caille", "cain", "caingang", "caingin", "cainian", "calc",
  "calcanea", "calcaneal", "calcanean", "calcanei", "calcannea", "calcannei", "calci",
  "calcic", "calcine", "calcining", "calean", "calic", "calicle", "caliga", "calili",
  "calin", "calina", "caline", "callan", "calli", "cana", "canaan", "canacee", "canaglia",
  "canaille", "canalage", "canale", "canali", "canaling", "canalla", "canalling", "cananaean",
  "cananga", "canc", "canccelli", "cancelli", "canel", "canela", "canell", "canella",
  "canellaceae", "canelle", "cangan", "cangia", "cangle", "canille", "caninal", "canli",
  "cann", "cannaceae", "cannalling", "cannel", "cannele", "cannelle", "cannie", "ceca",
  "cecal", "cecca", "cecil", "cecile", "cecilia", "ceil", "ceile", "cele", "celia",
  "celialgia", "cella", "cellae", "celling", "cenacle", "cene", "cicala", "cicale",
  "cienaga", "cienega", "cigala", "cigale", "cilice", "cilician", "ciliella", "cill",
  "cincinnal", "cincinni", "cine", "cinel", "cinene", "cingle", "clag", "clagging",
  "clanning", "clee", "cleg", "clin", "clinal", "cline", "clinia", "cnicin", "ecca", "ecce",
  "eccl", "egence", "elaeagnaceae", "elaic", "elance", "elec", "elegiacal", "elianic",
  "ellagic", "ellice", "encaenia", "encelia", "encell", "encia", "encina", "encinal", "encl",
  "eneclann", "englacial", "eniac", "gaelic", "gaincall", "galacaceae", "galenic",
  "galenical", "galician", "gallic", "gallican", "gallinaceae", "gallinacean",
  "gallinacei", "ganancial", "gangliac", "geic", "genic", "glaceing", "icacinaceae",
  "iceni", "icica", "ileac", "ilicaceae", "ilicic", "ilicin", "inca", "incage", "incaging",
  "incaic", "incan", "incl", "inle", "inlace", "inlacing", "lacca", "laccaic", "laccainic",
  "laccic", "laccin", "lacinia", "langca", "lecaniinae", "lecanine", "lenca", "lencan",
  "licania", "licca", "licence", "licencee", "licencing", "licinian", "lienic", "ligeance",
  "lilacin", "liliaceae", "linac", "linaceae", "lingence", "nace", "nacelle", "nance",
  "nangca", "neanic", "necia", "nicaeen", "niceling", "nicene", "nicenian", "nici",
  "deve", "devi", "devide", "devot", "devoto", "devove", "divet", "divi", "divid", "divoto",
  "evetide", "evite", "evited", "evoe", "optive", "ovid", "poove", "teevee", "totitive",
  "tovet", "veddoid", "vedet", "vedette", "veepee", "vetitive", "vide", "vidette",
  "viii", "vite", "viti", "vive", "vivo", "voet", "voidee", "voivod", "voivode", "voodooed",
  "bibb", "bibbing", "bibi", "bibio", "bibiri", "bigg", "biggin", "bigging", "bing", "bini",
  "bino", "biog", "bion", "biri", "birn", "biron", "birr", "birri", "birring", "bobo",
  "boggin", "bogo", "bogong", "boii", "bonbo", "bonging", "boni", "bono", "boong", "borg",
  "bori", "borning", "boro", "bororo", "bown", "bribri", "brin", "brob", "brog", "broo",
  "broon", "brrr", "gibbi", "gibbing", "gobbin", "gobbing", "gobi", "gobio", "gobo",
  "gorb", "inbow", "inbring", "inbringing", "inorb", "nibbing", "nibong", "nonborrowing",
```



```
"kraal", "kral", "lall", "lalo", "lawk", "loka", "lokao", "lola", "lollard", "lora",
"loral", "lowa", "oadal", "oakwood", "odal", "olla", "oolak", "orad", "orra", "raad",
"rada", "roak", "roka", "roral", "waar", "waka", "walla", "wallawalla", "wallwork",
"wardword", "wark", "warl", "warlow", "warrok", "warwork", "wawa", "wawl", "wloka",
"woad", "woak", "woald", "woodlark", "woodwall", "woodward", "woolwa", "woolward",
"worldward", "worral", "wraw", "wrawl")
```

```
word_list <- word_list %>% filter(!word %in% nonwords)
```

```
center_letter <- "a"
circle_letters <- c("w",
                    "k",
                    "o",
                    "l",
                    "r",
                    "d")
all_letters <- c(center_letter, circle_letters)
```

```
solving <- word_list %>%
  #filter to words with center letter
  filter(str_detect(word, center_letter) == TRUE) %>%
  #break work into letters
  mutate(lets = str_split(word, pattern=""))

solving <- solving %>%
  #check the word's letters against the puzzle's letters
  mutate(match = map(lets, ~.x %in% all_letters, lets=.x)) %>%
  #count number of matching letters
  mutate(matches = map(match, sum)) %>%
  unnest(matches)

answers <- solving %>%
  #keep words that only use the puzzle's letters
  filter(matches == length)

answers$word
```

```
## [1] "allow"    "ardor"    "arrow"    "award"    "awkward"  "dark"
## [7] "dollar"   "doodad"   "draw"     "drawl"    "koala"    "kola"
## [13] "lard"     "lark"     "load"     "okra"     "oral"     "radar"
## [19] "road"     "roadwork" "roar"     "walk"     "wall"     "wallaroo"
## [25] "wallow"   "ward"     "warlord"  "workload"
```

```
pangrams <- answers %>%
  filter(length >= 7) %>%
  #check the puzzle's letters against the word's letters
  mutate(lets = map(lets, unique)) %>%
  mutate(pangram = map(lets, ~.x %in% all_letters, lets=.x)) %>%
  #count number of matching letters
  mutate(p_matches = map(pangram, sum)) %>%
  unnest(p_matches)
```

```
pangrams %>%  
  #keep words that only use the puzzle's letters  
  filter(p_matches == 7) %>%  
  pull(word)
```

```
## [1] "workload"
```