

Day 4

Evan Bolyen

12/4/2021

Util

```
read_chunk = function(conn, n=-1, until=NULL) {  
  # R is such a nightmare for this kind of thing.  
  if (!missing(until)) {  
    until = substitute(until)  
    env = new.env()  
  }  
  lines = vector('character')  
  i = 0  
  while(n != 0) {  
    n = n - 1 # iteration limit  
    i = i + 1 # observed lines  
    line = readLines(conn, n=1)  
    if (length(line) == 0) { break }  
    if (!is.null(until)) {  
      assign('line', line, envir=env)  
      if(eval(until, env)) { break }  
    }  
    lines[i] = line  
  }  
  if (length(lines) == 0) { return (NULL) }  
  textConnection(lines)  
}
```

Shared

```
fh = file('input', open='r')
draws = unlist(read.csv(read_chunk(fh, until=(line == '')), header=FALSE), use.names=FALSE)
boards = list()
board_masks = list()
repeat {
  chunk = read_chunk(fh, until=(line == ''))
  if (is.null(chunk)) { break }
  boards = c(boards, list(as.matrix(read.table(chunk, header=FALSE))))
}
close(fh)
```

draws

```
##      [1] 23 30 70 61 79 49 19 37 64 48 72 34 69 53 15 74 89 38 46 36 28 32 45  2 39
##     [26] 58 11 62 97 40 14 87 96 94 91 92 80 99  6 31 57 98 65 10 33 63 42 17 47 66
##     [51] 26 22 73 27  7  0 55  8 56 29 86 25  4 12 51 60 35 50  5 75 95 44 16 93 21
##     [76]  3 24 52 77 76 43 41  9 84 67 71 83 88 59 68 85 82  1 18 13 78 20 90 81 54
```

boards[[1]]

```
##      V1 V2 V3 V4 V5
## [1,] 50 98 65 14 47
## [2,]  0 22  3 83 46
## [3,] 87 93 81 84 58
## [4,] 40 35 28 74 48
## [5,] 45 99 59 37 64
```

```
bingo = function(table) {
  any(apply(table, 1, all), apply(table, 2, all))
}
```

Problem 1

```
find_winner = function(draws, boards) {
  board_masks = lapply(boards, function(x) { x == Inf })
  for (draw in draws) {
    for (i in seq(boards)) {
      board_masks[[i]] = mask = board_masks[[i]] | boards[[i]] == draw
      if (bingo(mask)) {
        return(list(board=boards[[i]], mask=mask, board_number=i, draw=draw))
      }
    }
  }
}
```

```
winner = find_winner(draws, boards)
c(winner, result=sum(winner$board[!winner$mask]) * winner$draw)
```

```
## $board
##      V1 V2 V3 V4 V5
## [1,] 68 89 40 31 39
## [2,] 32 48 64 38 28
## [3,] 80 98 88 14 97
## [4,]  6 60 52 11 55
## [5,] 95 34 63 81  4
##
## $mask
##      V1    V2    V3    V4    V5
## [1,] FALSE TRUE FALSE FALSE FALSE
## [2,] TRUE  TRUE  TRUE  TRUE  TRUE
## [3,] FALSE FALSE FALSE FALSE FALSE
## [4,] FALSE FALSE FALSE FALSE FALSE
## [5,] FALSE TRUE  FALSE FALSE FALSE
##
## $board_number
## [1] 49
##
## $draw
## [1] 32
##
## $result
## [1] 31424
```

Problem 2

```
find_loser = function(draws, boards) {
  board_masks = lapply(boards, function(x) { x == Inf })
  to_check = seq(boards)
  for (draw in draws) {
    for (i in to_check) {
      board_masks[[i]] = mask = board_masks[[i]] | boards[[i]] == draw
      if (bingo(mask)) {
        to_check = to_check[to_check != i]
        result = list(board=boards[[i]], mask=mask, board_number=i, draw=draw)
      }
    }
  }
  result
}
```

```
loser = find_loser(draws, boards)
c(loser, result=sum(loser$board[!loser$mask]) * loser$draw)
```

```
## $board
##      V1 V2 V3 V4 V5
## [1,] 36 81 66 78 90
## [2,]  2 25 94 82 55
## [3,] 34 45  1 14 37
## [4,] 13  4 70 48 75
## [5,] 67 73 32 18 91
##
## $mask
##      V1    V2    V3    V4    V5
## [1,] TRUE FALSE TRUE FALSE FALSE
## [2,] TRUE  TRUE TRUE  TRUE  TRUE
## [3,] TRUE  TRUE FALSE TRUE  TRUE
## [4,] FALSE TRUE  TRUE TRUE  TRUE
## [5,] TRUE  TRUE TRUE FALSE TRUE
##
## $board_number
## [1] 16
##
## $draw
## [1] 82
##
## $result
## [1] 23042
```