# The birthday problem

## By Callum Simpson

In this document I will be going over my solution to the birthday problem

**Part 1**

This is the code I produced for part 1.

```
## A function for checking birthday duplicates give a certain RoomSize (room size being the number of people who are comparing birthdays)
contians_duplicates <- function(RoomSize) {

  ##creates a sample of birthdays
  ## 1:365 used to repestent each day of the year
  bdaysample <- sample(1:365, RoomSize, replace=TRUE)

  ## In case you want to check that it works correctly the following will print out the vector of "brithdays"
  ##print(bdaysample)

  ##If there was any duplicates return true, else return false
  return(any(duplicated(bdaysample)))
}
```

The idea is that for a given room size (or number of people) it will generate a set of numbers between 1 and 365 which will represent birthdays. If there are any duplicates in that set it will return a TRUE else it will return FALSE.

Here is some test of room size 10, with the set of numbers gotten and the result it gave.

```
> contians_duplicates(10)
 [1]  94 248  36   6  61   5  36 106 178 171
[1] TRUE
> contians_duplicates(10)
 [1] 193 358 319 247  58 219 195 236 113  64
[1] FALSE
> contians_duplicates(10)
 [1] 273 283 135 177 286 154   2 354 176 355
[1] FALSE
```

The first test contained a duplicate of 36 so returned true. The other test contained no duplicates so returned false

## Part 2.

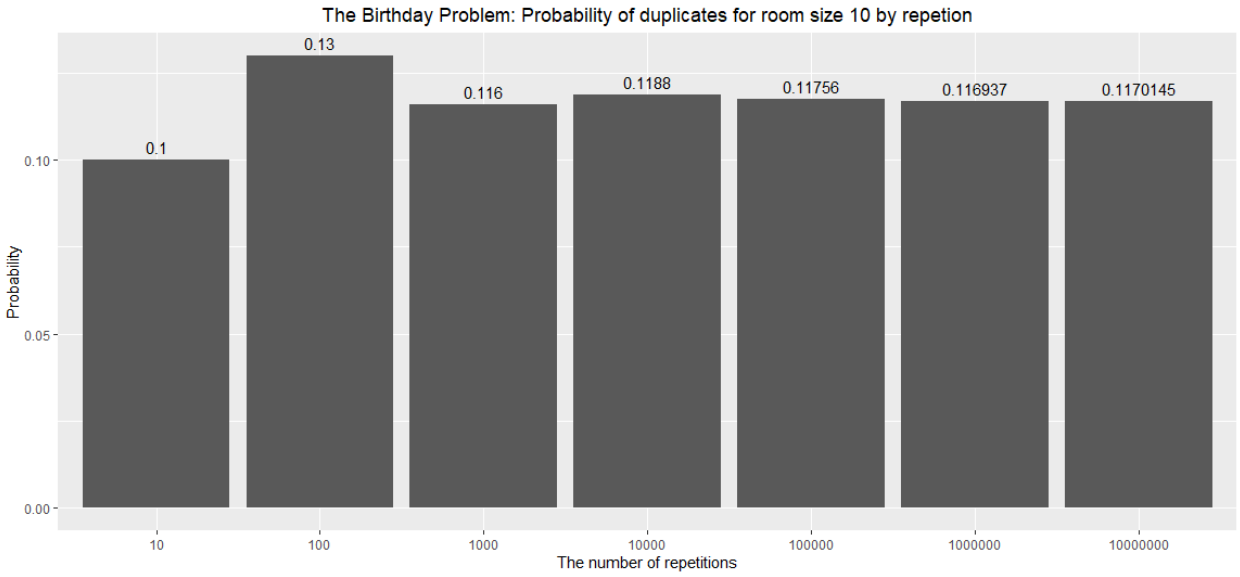
This is the code I produced for part 2.

```
## Part 2 (updated for part 3 - added a variable for Roomsize)

## A function for estimating the probability of two people having the same birthday given how many people are in the room
## Also takes in repetitions which the the number of time that the experiment gets ran
calculate_probability <- function(repetitions,RoomSize) {

  ##replicate the contains_duplicates function for the given room size for the inputted number of repetitions, store the results in results
  result <- replicate(n = repetitions, contians_duplicates(RoomSize))

  ##print(result)

  ##Calculate the probability by working out the sum of TRUE in result (so duplicates) then div by number of repetitions
  probality <- (sum(result, na.rm = TRUE)) / repetitions

  return(probality)
}
```

Here is the results for a ten-room size by 10 reparations

```
> calculate_probability(10,10)
 [1] 227 130  46 145 237 154  11 307 330 145
 [1] 233  69  51  80  18  20 254 229 141 341
 [1]  61 117 257 227 328 235 190  17  96 356
 [1] 179 280  14  71 215  11 280 246 189 325
 [1] 294  93 276 319 133 161 272  55 219 168
 [1] 191 219 142 134 229 101 241 117 187 268
 [1] 226 309 127  89 315   6 330  71 239 290
 [1] 215 248 162 186 161  76 343 328 126 219
 [1] 174 346 288 153  45 272 102 338 271  51
 [1]   6 199  65 130 225  81  58 117 301 159
 [1]  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[1] 0.2
```

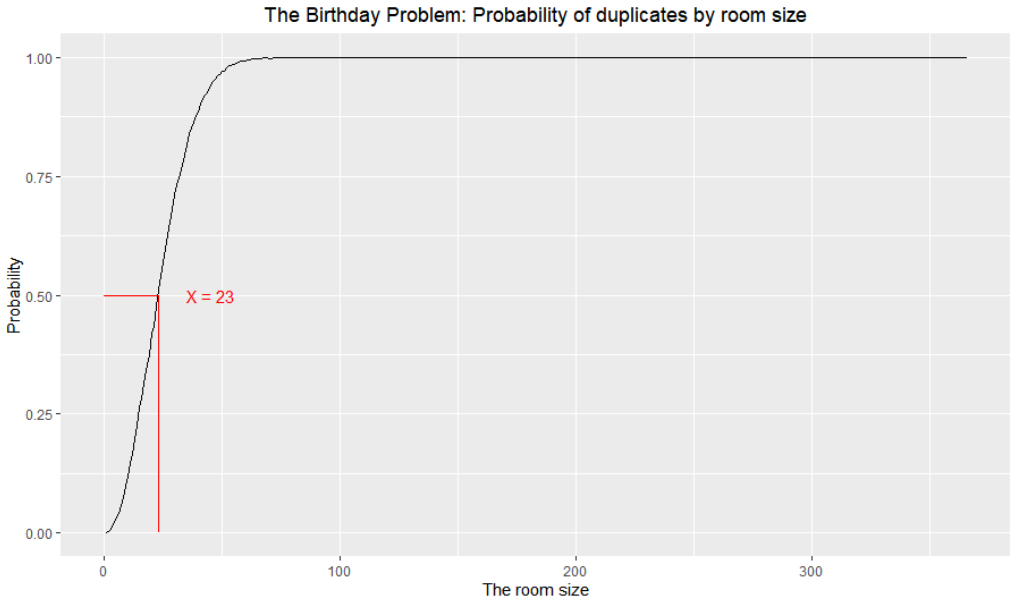
As you can see the first set and the fourth set contained duplicates and the rest didn't, so 2 out of 10 produced True which translate to 0.2. To further test this I used a list of the powers of 10 and produced the following line graph. As you can see from this graph the probability gotten from low number of repetitions fluctuated a bit however as we start to greatly increased the number of repetitions done the line starts to stabilize at around 0.117 which is the expected probability of the room size 10.




This graph is subject to change depending on the how lucky / unluck you are with collision for the first few sets of repetitions. Originally, I made this graph to be a line graph (the code is still there to do this) however "a plot of proportion" seemed more bar cart. Sorry if I've missed understood this

**Part 3**

After amending my code to work with different room sizes and different repetitions I went about testing the probabilities of duplicates between the room sizes 1 to 366 (with repetition size of 10000).




As the room size increase the probability that two people will share the same birthday also star increases. As shown by the red line the number of people that we need to be in the room for the probability the there is a shared birthday is at least 50% is 23, the table below shows further evidence of this.

| | |
|---|---|
| 20 | 0.4177 |
| 21 | 0.4414 |
| 22 | 0.4685 |
| 23 | 0.5098 |
| 24 | 0.5357 |
| 25 | 0.5664 |
| 26 | 0.5970 |

This result (and other probabilities that have been generated for different room sizes) matches that found on the Wikipedia page for the birthday problem https://en.wikipedia.org/wiki/Birthday_problem.

## Note the probability table generated for this graph tends to round to 1 before we reach 366, this is an issue with the number of repetitions being used to create the table(I.e the more reptation's we use the more accurate the results and less probabilities of 1). I know that really when we reach a room size of 200 the probability should be 0.999999999999999999999999999998% (as shown on the wiki page) and not 1 but I believe that would take way too much processing power. Even at 500000 repetitions it was only able to get a correct probability up to room size 87.