```
## Wrap sample into a function that avoids the "convenience"
## behaviour that happens when the length of x is one

sample_safer <- function(to_sample, n) {
  assert_that(n <= length(to_sample))
  if (length(to_sample) == 1)
    return(to_sample)
  else {
    return(sample(to_sample, n))
  }
}


## Simulate a generalised Monty Hall situation with
## w prizes, d doors and o doors that are opened.

sim_choice <- function(w, d, o) {
  ## There has to be less prizes than unopened doors
  assert_that(w < d - o)
  wins <- rep(1, w)
  losses <- rep(0, d - w)
  doors <- c(wins, losses)

  ## Pick a door
  choice <- sample_safer(1:d, 1)

  ## Doors that can be opened
  to_open_from <- which(doors == 0)

  ## Chosen door can't be opened
  to_open_from <- to_open_from[to_open_from != choice]

  ## Doors to open
  to_open <- sample_safer(to_open_from, o)

  ## Switch to one of the remaining doors
  possible_switches <- setdiff(1:d, c(to_open, choice))
  choice_after_switch <- sample_safer(possible_switches , 1)

  result_hold <- doors[choice]
  result_switch <- doors[choice_after_switch]
  c(result_hold,
    result_switch)
}


## Formulas for probabilities

mh_formula <- function(w, d, o) {
  ## There has to be less prizes than unopened doors
```

```
  assert_that(w < d - o)

  p_win_switch <- w/d * (w - 1)/(d - o - 1) +
                      (1 - w/d) * w / (d - o - 1)
  p_win_hold <- w/d
  c(p_win_hold,
    p_win_switch)
}


## Standard Monty Hall

mh <- replicate(1000, sim_choice(1, 3, 1))

> mh_formula(1, 3, 1)
[1] 0.3333333 0.6666667
> rowSums(mh)/ncol(mh)
[1] 0.347 0.653
```

The Monty Hall problem problem

Guest & Martin (2020) use this simple problem as their illustration for computational model building: two 12 inch pizzas for the same price as one 18 inch pizza is not a good deal, because the 18 inch pizza contains more food. Apparently this is counter-intuitive to many people who have intuitions about inches and pizzas.

They call the risk of having inconsistencies in our scientific understanding because we cannot intuitively grasp the implications of our models ”The pizza problem”, arguing that it can be ameliorated by computational modelling, which forces you to spell out implicit assumptions and also makes you actually run the numbers. Having a formal model of areas of circles doesn’t help much, unless you plug in the numbers.

The Monty Hall problem problem is the pizza problem with a vengeance; not only is it hard to intuitively grasp what is going on in the problem, but even when presented with compelling evidence, the mental resistance might still remain and lead people to write angry letters and tweets.