

# Homework 1

*Zoey Nguyen*

*10/8/2019*

## Question 1

Here we will be assuming that the input given by the user is formatted as expected with the correct units, whatever those units may be.

1. Create the variables distance, rate, and time.
2. Have distance, rate, and time correspond to the numbers 1, 2, and 3. Output this information to the user and ask them which variable they want to find input a number from 1 to 3.
3. Save the user's index.
4. For i in 1 to 3:
  - a. If the i = user's index: - Move on to the next value of i in this loop.
  - b. Else:
    - Tell the user which variable corresponds to the current i and ask them what value to set it to.
    - Set the variable corresponding to the current i to this input.
5. Return the correct value:
  - a. If the user's index corresponded to distance:
    - Return rate times time.
  - b. Else if it corresponded to rate:
    - Return distance over time.
  - c. Else it corresponded to time:
    - Return distance over rate.

## Question 2

A "higher" letter/name/book is one that comes later in the alphabet.

1. Arrange all the books side-by-side in a line from left to right.
2. For each book in the bookline, starting from the left ("main book"):
  - For every other book in the bookline, starting from the left ("comparison book"):
    - Call the comparison function on the names of the two books to get the name of the book which is higher.
    - If the function returns the main book's name (indicating the main book's name is higher), AND the main book is to the left of the comparison book, insert the main book in the bookline immediately to the right of the comparison book.
    - Move on to the next comparison book (which will either be immediately to the right of the current comparison book or immediately to the right of the main book).
3. Comparison function, which takes in two book names and returns the name which is higher.
  - For i in 1 through the length of the first book's name:
    - If i is greater than the length of the second book:
      - \* Return the first book's name.
    - Else:
      - \* If the ith character in the first book's name comes earlier in the alphabet than the ith character in second book's name:
        - Return the second book's name.
  - Return the second book's name. (Since if the comparison got to this point, it means the first book's name exactly match the first part of the second book's name, but the second book's name is longer and therefore higher.)

### Question 3

(Hong Kong egg tart recipe.)[<https://thewoksoflife.com/hong-kong-egg-tarts/>] Not technically a cake because I don't like cake much but I love these. I love these to the point where I feel like I need to shoehorn egg tarts into this assignment somehow.

1. Go into the kitchen.
2. Gather a measuring cup, measuring spoons, a whisk, a large mixing bowl, a medium bowl, saran wrap, a cutting board, a fine mesh strainer, and at least 12 small tart tins.
3. Gather the necessary packs of ingredients and put them out on the counter.
4. Using the measuring materials, measure out from these pack the following amounts and place each amount into a separate appropriately-sized bowl:
  - a. 1 1/2 cups all purpose flour
  - b. 1/8 teaspoon salt
  - c. 1/2 cup sugar, plus 1 tablespoon
  - d. 14 tablespoons butter
  - e. 2 tablespoons cold water
  - f. 3 large eggs, at room temperature
  - g. 1/2 cup evaporated milk, at room temperature
  - h. 3/4 teaspoon vanilla
5. Put away your packs of ingredients except for the flour.
6. Pour the measured flour, salt, and sugar into the large mixing bowl.
7. Mix this with the whisk until you cannot see any contrasting concentration of any single ingredient.
8. Put the butter in this bowl and mash it into the flour with your hands until little chunks of butter (about 1/2 inch across) are evenly distributed in the bowl.
9. Put the cold water in this bowl and continue mashing the mixture together with your hands until the mixture is one consistency.
10. Cover the bowl with saran wrap and place it in the fridge for 30 minutes.
11. Sprinkle flour over the cutting board until the flour is just evenly distributed.
12. Take the bowl out of the fridge, put the dough on the cutting board, and knead until smooth.
13. Form an 8"x20" rectangle with the dough by rolling the dough only away from you such that it looks taller than it is wide from your perspective.
14. Fold the farthest third of the dough from you towards you, then the bottom third of the dough away from you.
15. Turn the dough 90 degrees on the counter.
16. Repeat steps 13 and 14.
17. Wrap dough rectangle in saran wrap and place in fridge for 30 minutes.
18. Begin making the filling as it is chilling.
19. Measure 1 cup of hot water with the measuring cup and pour into a medium bowl.
20. Pour sugar into the hot water and mix with the whisk until dissolved.
21. Let hot water mix cool.
22. In the other medium bowl, crack eggs (without shell).
23. Pour the evaporated milk into the egg mixture.
24. Mix the evaporated milk with the eggs using the whisk until the mixture is consistent.
25. Pour the sugar water and vanilla into the egg mixture.
26. Mix the egg mixture until the mixture is completely smooth and aerated.
27. Clean the measuring cup.
28. Pour the egg mixture through the strainer into the measuring cup.
29. Take saran-wrapped dough out of fridge.
30. Remove saran wrap.
31. Roll out dough evenly until about 1/4" thickness.
32. Cut circles in the dough just large enough so that when a circle is pressed into a tart tin it just reaches the outside edge of the tin.
33. For each circle of dough, place into tart tin and smooth out by pushing from the center up the side of the tin.

34. For each filled tart tin, pour egg mixture in the measuring cup into the tart tin until about 90% of the height of the inner tart.
35. Preheat oven to 400 degrees Fahrenheit.
36. When oven is preheated, slide tarts onto a tray and place into the oven.
37. After 15 minutes of baking, reduce temperature to 350 degrees Fahrenheit.
38. After 10 minutes of baking, turn oven off.
39. Remove tarts from oven and place onto counter.
40. Let cool until they won't burn your mouth.
41. Eat as many as desired.

## Question 4

```
getwd()

## [1] "C:/Users/zooeey/Desktop/UCLA/Fall 2019/Stats 20"

dir()

## [1] "hangman.html"          "hangman.rmd"
## [3] "Homework-1.html"       "Homework-1.pdf"
## [5] "Homework-1.Rmd"        "Homework 1.Rmd"
## [7] "rmarkdown_by_example.Rmd" "Stats 20.Rproj"
```

## Question 5

### Part a

```
cylinder_area <- function(radius, height) {
  circle_faces <- 2*pi*radius^2
  rectangle_faces <- 2*pi*radius*height
  total_area <- circle_faces + rectangle_faces
  total_area
}
```

### Part b

```
cylinder_area(5,8)
```

```
## [1] 408.407
```

## Question 6

### Part a

```
142 %% 24
```

```
## [1] 5
```

Andy was lost for 5 full days.

### Part b

```
(13 + 142 %% 24) %% 24
```

```
## [1] 11
```

Andy arrives at 11:00am.

### Part c

```
function arrival_time(t, x) {  
  hours to add to hour t = remainder of hours when x is converted into days and hours  
  
  total = t + hours to add to hour t  
  
  time of arrival = remainder of hours when total is converted into days and hours  
  
  return time of arrival  
}
```

### Part d

```
arrival_time <- function(t,x) {  
  total <- t + x%%24  
  total%%24  
}  
  
# Verification for t = 13, x = 142. Should be 11.  
arrival_time(13,142)  
  
## [1] 11
```

## Question 7

### Part a

```
payment_size <- function(P, i, n) {  
  result <- P*i/(1-(1+i)^-n)  
  result  
}
```

### Part b

```
payment_size(50000,.04,24)
```

```
## [1] 3279.342
```

## Question 8

### Part a

There is no y passed into the function to be locally treated and only one argument can be passed into the function. This means that you get the sum of the y already globally defined as 4, and whatever gets passed into `sum_y`. If you pass nothing into `sum_y`, the default number that is added to y is 0 (as seen in the arguments, `x=0`).

### Part b

A function nested inside the other would mean that once you call the outer function, the outer function call works as the start of the call (or the name) for the inner function. Typing just `sum(xy)` itself will not output a numeric value but it shows the inner function structure.

### Part c

```
x <- 1
y <- 4
sum_xy <- function(x){
  function(y){
    x + y
  }
}

sum_xy(143)(637)

## [1] 780
```