Data 2401 - Exam 1

Dr. Shoemaker

Spring 2022

Please read the UHD honor code below. Turning in this exam acknowledges that all work submitted on this exam is your own, and that you have not used any off limit resource.

> "We will be honest in all our academic activities and will not tolerate dishonesty."
>
> – UHD Academic Honor Code

You are submitting this exam under the UHD honor code. Answer all below questions on the Exam1_template.Rmd provided on Github. Complete the questions, knit the file to html, commit and push to your repository *BEFORE 1:15* and turn in link to the repository to Blackboard. **Any commit that is pushed after 1:15 will not be graded.**

Part 1 (12 points)

1. Create the vector of all the integers from -10 to 10 in three ways: once using `c()`, once using `:`, and once using `seq()`
2. Create the vector `[3, 6, 9, 12, 15]` in two ways, once using `c()` and once using `seq()`
3. Create a vector that contains the log of the integers from 15 to 25. You must use a vectorized function.
4. Create a vector of your three favorite foods. Then, in a separate command, add your favorite drink to end of the vector.

Part 2 (16 points)

```
# Data for this part

student <- c("a", "b", "c" ,"d" ,"e" ,"f", "g" ,"h", "i" ,"j", "k", "l", "m", "n", "o")
midterm_grades <- c(95, 78, 76, 91, 87, 84, 92, 72, 72, 81, 82, 93, 69, 94, 80)
final_exam_grades   <- c(92, 90, 95, 87, 88, 98, 96, 79, 83, 75, 84, 73, 91, 78, 97)
```

Use vectorized functions for the following:

a. Everyone got 5 points of extra credit on the midterm. Change the midterm grade vector to reflect this.
b. How many students scored higher on the midterm than the final? (Note, use the extra credit version of the midterm for this)
c. Create a new vector that contains the sum of each student's midterm and the final exam grade divided by 2. (You are finding the average for each student)
d. Create a boolean vector that indicates whether each student has an "A" average, greater than or equal to 90.
e. Use this vector to print/display the names of the students who made an "A".
f. Challenge (+2 bonus points): How many students made an A if the midterm is 40% of the grade and the final is 60%.

Part 3 (16 points)

Use this data for the following questions. You may make the assumption that the vectors' elements are related, i.e., car "a" is pink, 13 years old, and costs 53.

```
# Data for Part 5

car.names <-  c("toyota", "ford", "honda", "bmw", "kia", "dodge", "jeep", "tesla", "subaru", "lexus")
car.colors <- c( "pink", "blue", "blue", "black", "green",
                 "green",  "black", "black", "green", "yellow")
car.ages   <- c(13, 5, 36, 2, 4, 67, 24, 2, 6, 16)
car.prices <- c(53, 87, 54, 532, 58, 99,  66, 132, 32, 345)
```

Using indexing, and using logical indexing when possible, answer the following questions. Your code output can be considered an answer.

```
a. What is the age of the last car?
a. How many green cars are there?
a. How many cars are older than 20 years old?
a. What are the names of the black cars?
a. How many cars had prices greater than 250 or less than 50?
a. Are any of the cars older than 400 years old?
a. Create a new vector that contains the prices of the black and green cars.
a. Check to see if "lambo" is the name of a car.
```

Part 4 (16 points)

Continue to use the car data from above.

1. Make a list called `car` where you store each of the vectors tagged with the name of what they are. (Example: `car.names` would be tagged `names`)
2. Access the element of the list with the ages of the cars in two different ways.
3. Add another element to `car` called `is_old` that contains a boolean vector of whether or not each car is more than 40 years old.
4. Remove `colors` and `names` from `car`, then use `lapply` to find the maximum age and price.

Part 5 (10 points)

Write a function that takes two vectors, finds the maximum value of each vector, and returns the sum of those maximums. Test the function on two vectors of your own creation.

Part 6 (15 points)

Often you will need to recode values of a dataset. For example, if you have a survey of age data, you may want to convert any crazy values (like anything below 0 or above 110) to the special value NA.

- Write a function called `check.ages()` that takes one argument: `x`, a vector of numbers (and we'll assume the user will do so correctly, don't worry about error checking here). The function should change any values of `x` that are below 0 or above 110 to NA, and then return the edited vector.
- Test the function on the vector `test_x` given below. Your function should return `18 NA 97 NA`

```
test_x <- c(18, -6, 97, 230)
```

Part 7 (15 points)

Consider a company Bamazon. They want to launch a new drone delivery service, that has rates based on how far you are from the warehouse. There is a special lower rate for their "Brime" members.

The table below has their current pricing plan:

Distance	Cost
Brime Member Rate	$2.00
0 through 10 miles	$4.00
More than 10 miles, but less than 20 miles	$7.50
More than 20 miles, but less than 50 miles	$9.00
50 miles or more	$22.50

1. Write a function `drone_cost` that takes two arguments: the distance of the delivery and whether or not the customer is a Brime member. The function should return the price of the delivery.
2. Test the function for a Brime member that needs a delivery 14 miles from the warehouse, and 4 different non-Brime members that are 5, 15,35, and 60 miles from the warehouse .

Extra Credit: 5 points

Modify your `check.ages()` function to also take the lower bound and upper bound (`lb` and `ub`) as arguements: the function should look at the values of `x`, convert any values below `lb` and above `ub` to `NA`, and then return the edited vector.