

# EEMB179\_homework\_1

Kerri Luttrell

1/7/2021

## Homework exercise

1A. Use the `:` or `c()` or `seq()` function to create a vector of numbers that ranges from 1 to 10. (1 pt) 1B. Name the vector “vector.1” using variable assignment. (1 pt) 1C. Print the vector. (1 pt)

```
vector.1<- seq(from=1, to=10, by=1)
vector.1
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

2A. Use the `rep()` function to create a holding vector for a `for()` loop populated with NaN. Make this vector the same length as vector.1 from #1 (a length of 10). (1 pt) 2B. Name the vector “vector.2” using variable assignment. (1 pt) 2C. Print the vector. (1 pt)

```
vector.2<- rep(NaN, times=length(vector.1))
vector.2
```

```
## [1] NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN
```

3. Write in words what each line of code means. Some lines have been annotated for you. (5 pts)

```
vector.1 <- 1:10
# store the vector (1,2,3,4,5,6,7,8,9,10) in the object vector.1
vector.1
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
# print the object vector.1
```

```
vector.2 <- rep(NaN, times = length(vector.1))
# store in the object vector.2, fill vector.2 with Not a Number repeatedly until it is the same length

for(i in 1:length(vector.1)) {
  vector.2[i] <- vector.1[i] + 1
}
# for every value in vector.2 replace it with the equivalent place value in vector.1 plus one.
vector.2
```

```
## [1] 2 3 4 5 6 7 8 9 10 11
```

```
# print vector.2
```

4. In #3 you used the letter 'i' to denote each variable in vector.1 and vector.2. If you replaced 'i' with 'j', would anything change about your output? Please explain your answer in the space below. (2 pts)

Yes. The for loop would not run and the values for vector.2, which we can see in the Environment, would not change from NaN. Thus, the output for when you print vector.2 would remain [ ] NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN instead of changing to [ ] 2 3 4 5 6 7 8 9 10 11.

5. We will use a for() loop to iterate over a vector and solve for variable 'y' using the  $y = mx + b$  equation. Please follow the prompts below. Some prompts have been completed for you. (8 pts)

```
# Use variable assignment to assign 10 as b and 1 as m
b <- 10
m <- 1

# Create a vector that ranges from 1 to 100 and name it 'x' using variable assignment. (2 pts)
#####
x<- (1:100)

#####

# Create a holding vector for 'y' using the rep() function that is of the same length as vector x, but j
#####

y<- rep(NaN, times= length(x))

#####

# Write out what each line below means in words:
for (i in 1:length(x)) { # for the variables in x (1pt)
  y[i] <- m * x[i] + b # replace the ith value of y with m times the ith value of x plus b (1pt)
}

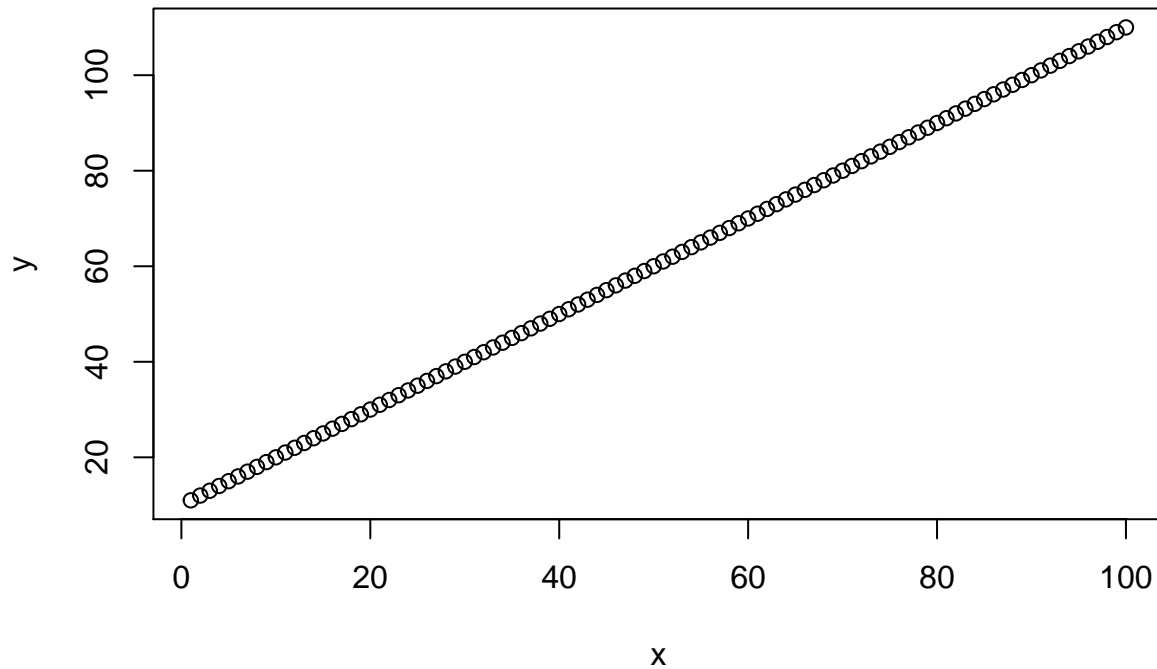
# Print y (1 pt)
#####

y
```

```
## [1] 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
## [19] 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46
## [37] 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64
## [55] 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82
## [73] 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
## [91] 101 102 103 104 105 106 107 108 109 110
```

```
#####

# Use the plot function to plot y as a function of x
plot(y~x)
```



6. Write a `for()` loop to 1) take the square root and then 2) add 2 for each value in a vector.

```
# vector of values
fibseq <- c(0, 1, 1, 2, 3, 5, 8, 13, 21, 34)
length(fibseq)
```

```
## [1] 10
```

```
# create a holding vector (1 pt)

holding<- rep(NA, times=length(fibseq))
length(holding)
```

```
## [1] 10
```

```
# write a for() loop (2 pts)
##### WRITE YOUR FOR() LOOP HERE #####
```

```

for (i in 1:length(fibseq)) { # for the variables in fibseq (1pt)
  holding[i] <- sqrt(fibseq[i]) +2 # replace the ith value of y with m times the ith value of x plus b
}

#####

# print the final vector (1 pt)
holding

```

```

## [1] 2.000000 3.000000 3.000000 3.414214 3.732051 4.236068 4.828427 5.605551
## [9] 6.582576 7.830952

```