1. Use the seq() function (actually, I’m going to give you this one: seq(from=2, to=98, by=3)) to create a vector that contains a sequence from 2 to 98 by 3’s. Store this vector in an object called s. Use this object to answer the following:
   1. What are the first 5 values of s?
   2. What are the first, third, tenth, and twenty first values of s?
   3. Return all the values of s except the second, eighteenth, and thirty-second value.
   4. Return all the even values of s. (HINT: even values return a 0 when mod 2 applied. A mod can be applied with %%).
2. For this question, “xor” will be used to mean an exclusive or and “or” will be used to mean an inclusive or.
   1. Create the following logical objects with the corresponding truth values:

A = TRUE

B = TRUE

C = FALSE

D = TRUE

* 1. Use R to determine the truth values of the following expressions:

1. A and B
2. A xor B
3. (A and C) or D
4. (A or D) and C
5. Not(A and C)
6. Not(A and B) or B
7. The Sieve of Eratosthenes is an ancient algorithm used to find prime numbers from 1 to n quickly. It works like this: Write down all numbers between 1 and n. Starting with 2, go through and remove every value from the list that is a multiple of 2. Next move to 3 and remove every remaining value that is a multiple of 3. Repeat with the remaining numbers until you have reached n/2. The numbers that are left are all the prime numbers between 1 and n. Using this approach, use loops to create a vector that contains all prime numbers between 1 and 1,000.

HINT: You will need to remove numbers from a vector. The best way to do this is to create a new vector and bring over only the values you want from the original vector. Think about how you can use logical operators to accomplish this.

1. Did you take a pass on the last homework’s for() loop? Maybe give it another try. I think you’re ready for it now!