Hw 2

2024-01-28

## Q1: Multiples of 3

1. Firstly, we will need a counter which starts at 1 and ends at a 100.
2. We will divide each number by 3 and if there is no remainder then the number is a multiple.
3. For this we can use a modulus function.
4. We then add these numbers to a variable called sum.

a <- 1:999  
multiples\_3\_or\_5<-a[(a %% 3 == 0 | a %% 5 == 0)]  
multiples\_sum<-sum(multiples\_3\_or\_5)  
multiples\_sum

## [1] 233168

## Q2: Fibonacci Numbers

1. We will need a counter variable to start with from 1 to 4000000.
2. We can use the modulus function here as well with 2 to find the even numbers.
3. We can add these numbers to a varable called sum.
4. in order to increase the counter variable(which is our fibonnaci number in the sequence 1 to 4000000) we can take the current counter and add the previous counter to it.

prev <- 1  
counter <- 2  
sum\_even <- 0  
while (counter <= 4000000) {  
 if (counter %% 2 == 0) {  
 sum\_even <- sum\_even + counter}  
 next\_term <- prev + counter  
 prev <- counter  
 counter <- next\_term}  
sum\_even

## [1] 4613732

## Q3: Largest Prime Factor of 600851475143

1. we will need a counter variable from 1 to 600851475142, which is one less than the given number
2. We will take another variable called largest prime factor.
3. We will take the modulus of the number given in the question with the number that is the current counter in order to find out if the counter number is a prime factor.
4. Then we check if the the prime factor is the largest one we have so far by comparing it to the largest prime factor variable. If it is larger then it will be our new value in the variable.
5. We continue to repeat this process until the counter runs out.

count <- 1  
maxpf <- 0  
while (count <= 60085) {  
 if (600851475142 %% count == 0) {  
 if (count > maxpf) {  
 maxpf<-count}}  
 count<-count+1}  
maxpf

## [1] 45134