Homework 1

Han Nguyen - TXN200004

09/02/2024

Problem 1

a)

```
a <- round(8 + 9 - 7 / 3 ^ 0.3, 2)
```

Answer: $8 + 9 - \frac{7}{3^{0.3}} = 11.97$

b)

```
b <- round(log(sqrt((15 + 6) / (14 + 2)), base = 2), 2)
```

Answer: $\log_2\left(\sqrt{\frac{15+6}{14+2}}\right) = 0.2$

c)

```
c <- round(((11 + sin(pi/4)) / (factorial(3) + abs(-10))) ^ 2, 2)
```

Answer: $\left(\frac{11+\sin\left(\frac{\pi}{4}\right)}{3!+|-10|}\right)^2 = 0.54$

d)

```
d <- round(6 + 5 - 4/3^2, 2)
```

Answer: $6 + 5 - \frac{4}{3^2} = 10.56$

e)

```
e <- round(exp( sqrt( (14 + 13) / (12 + 11) ) ), 2)
```

Answer: $e^{\sqrt{\frac{14+13}{12+11}}} = 2.95$

f)

```
f <- round(((11 + factorial(12)) / (factorial(13) + 14)) ^ 2, 2)
```

Answer: $\left(\frac{11+12!}{13!+14}\right)^2 = 0.01$

Problem 2

a)

```
RF <- c(2.60, 3.05, 3.74, 3.48, 5.49, 4.25, 2.57, 2.18, 3.14, 4.82, 3.28, 3.01)
```

Answer: RF = 2.6, 3.05, 3.74, 3.48, 5.49, 4.25, 2.57, 2.18, 3.14, 4.82, 3.28, 3.01

b)

```
names(RF) <- month.abb
```

Answer: The names of the RF vector are assigned to the abbreviated month names: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec

c)

```
avg_RF <- round(mean(RF), 2)
```

Answer: The average RF value is 3.47

d)

```
min_month <- names(which.min(RF))
max_month <- names(which.max(RF))
```

Answer:
- Month with the minimum RF is Aug.
- Month with the maximum RF is May.

Problem 3

a)

```
H2 <- c(2700, 2600, 3050, 2900, 3000, 2500, 2600, 3000, 2800, 3200, 2800, 3400)
```

Answer: H2 = 2700, 2600, 3050, 2900, 3000, 2500, 2600, 3000, 2800, 3200, 2800, 3400

b)

```
names(H2) <- month.abb
```

Answer: The names of the H2 vector are assigned to the abbreviated month names Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec

c)

```
total <- sum(H2)
```

Answer: Total sales of Hummer H2 in 2002 = 34,550

d)

```
diff <- diff(H2)

max_increase <- names(which.max(diff))
max_decrease <- names(which.min(diff))
```

Answer:
Month with greatest increase: `Dec`.
Month with greatest decrease `Jun`.

Problem 4

a)

```
x <- c(1, -2, 3, -4, 5, 100)
y <- x * -1
y[y > 0]
```

```
## [1] 2 4
```

b)

```
# create a sequence from 1 to 50
z <- seq(1, 50)

# test whether an observation is even
even <- z %% 2 == 0

# subset z by the test above
z <- z[even]
```

c)

```
mean <- function(x) {
  sum(x) / length(x)
}
```

Problem 5

```
PrintSquare <- function() {
  for (i in 1:1000) {
    square_root <- sqrt(i)
    if (square_root == as.integer(square_root)) {
      print(i)
    }
  }
}

#Testing
PrintSquare()
```

```
## [1] 1
## [1] 4
## [1] 9
## [1] 16
## [1] 25
## [1] 36
## [1] 49
## [1] 64
## [1] 81
## [1] 100
## [1] 121
## [1] 144
## [1] 169
## [1] 196
## [1] 225
## [1] 256
## [1] 289
## [1] 324
## [1] 361
## [1] 400
## [1] 441
## [1] 484
## [1] 529
## [1] 576
## [1] 625
## [1] 676
## [1] 729
## [1] 784
## [1] 841
## [1] 900
## [1] 961
```

Problem 6

```
TwinPrime <- function(n) {
  # Function to check if a number is prime
  IsPrime <- function(x) {
    # Numbers <= 1 are not prime
    if (x <= 1) {
      return(FALSE)
    }

    # 2 and 3 are prime
    if (x == 2 || x == 3) {
      return(TRUE)
    }

    # Check up to square root of x
    # Skip 4 because it's not prime
    i <- 5
    for (i in 2:floor(sqrt(x))) {
      if (x %% i == 0) {
        return(FALSE)
```

```
      }
    }
      return(TRUE)
  }

  # Constructing a primes vector from 1 -> n
  primes <- c()
  for (i in 2:n) {
    if (IsPrime(i)) {
      # new vector and add new prime in
      primes <- c(primes, i)
    }
  }

  # Find and count all twin pairs
  total_pair <- 0
  for (i in 1:(length(primes) - 1)) {
    if (primes[i + 1] - primes[i] == 2) {
      total_pair <- total_pair + 1
    }
  }

  return(total_pair)
}

# Testing
TwinPrime(100)
```

```
## [1] 8
```