I. Geometric Distribution. Provide an R code for the geometric distribution. The geometric distribution is a probability distribution that models the number of trials required to achieve the first success in a sequence of Bernoulli trials, where each trial has a constant probability of success.

1. Set the probability of success: p <- 0.2

```r
p = 0.2
```

2. Generate 1000 random variables from the geometric distribution.

```r
x = rgeom(n = 1000, prob = p)
x
```

```
##    [1]  1  6  6  6  3 23  0  2  6  5  7  3  1 10  0 12  1  1  3  1  9  3  0  0
##   [25]  1  2  5  2  7  6  3 18  1  5  2 14  9  7  0  5  0  2  1  0  4  1  1  7
##   [49]  0  4  2  1  5  2  0  2  4  1  0  7  1  7  0  2 11  3  2  0  2  5  7 27
##   [73]  5  6 11  2  0  0  0 11  0  7  1  1  0  1  1  0  0  6  0  4  0  0 14  1
##   [97] 16  0  7  9  0  0  5  7 11  0  1  1  5  2  2  1  2  1  7  2 14  1  3  0
##  [121]  1  2  0  2  1  0  3  0  0  1  0  0  8  3  0  2  1  1  7  1  8  1  0  6
##  [145]  1  4  1  2  3  3  2  7  1  2  2  0  2  4  3  1  2  8  5  0  1  2  3  0
##  [169]  3  6  8 13  6  1  1  3  7 12 15  0  5  1  1 14  1  4  3  5  2  0  7  3
##  [193]  3  5  5  6  7  0  1  6  5  7  1  3  1  2  3  6  3  5  3  8  6  1  4  4
##  [217]  4  2  3  1  1  2  2  1  8  1  4  4  1  1  2  4  2  1  6  0  0  4  2  8
##  [241]  5  0  3 11  9  6  1  0  1  4  1  0  5  1  3  8  0  1  1  6  4  0  2  0
##  [265] 10  2  1 12  8  7  1 19  0  0  5  3  0  4  2  0  1 17  2  1  1  2  2  2
##  [289]  0  4  8  7  6  8  5  5 18  2  2  1  5  0  1  0  0  1  2  0  1  9  0  3
##  [313]  0  2  4  1  3  1  7  7  2  3  5  0  0  2  3  0 11  1  0  2  1  1 10  1
##  [337]  9  0  8  3  0  3  5  0  5  1  6  4  2  1  4  3  2  7  0  3  6  1  3  2
##  [361] 10  6  3  4  1  4  1  0  1  0  0  8  5  2  2 18  0  8  0 13  0  2  3  1
##  [385]  3  0  0  6  3  4  2  6  2 18  3  1  1  1  0  1  6  6  4  1 18  1  1  5
##  [409]  1  1  3  0  3  1  0 17  3  0 14  1  1 17  9  9  0  3  1  2  5  5  9  6
##  [433]  0  2 12  2  2  7  3  3 11  3  4  1  4  0  2  3  2  2  4  0 14 11  7  0
##  [457]  7  8  8  4  4  1  2  3  1  5 21  6  0 12  3  1 16  1  4  4  3  0  9  0
##  [481]  0  3  2  0  0  0  0  1  4  7  7  8  3  1  3  0  2  2  0  1  8  0 13  2
##  [505]  7  4  2  2  0  0  3  1  6  0  2  3 10  0  9  6 13 10  7  5  1 10  8  4
##  [529]  0  0  1  6  0  0  2  5  1  2  4  2  1  3  0  0  8  9  5  2  3  0  1  6
##  [553]  0  3  6  2  8  2  1  2  3  3  2  0  0  9  1  3  0  9  2  0  0  0  1  0
##  [577] 10  4  3  0  5  2 10  6  1  0  3  4  6  0  0  2  0 11  5 18  2 12  2  0
##  [601]  4  0  9  4  0  5  5  3  9  1  0  4  2  8  2  6  2  2  1  3  8 11  0  0
##  [625]  2 11  2  3  1 13  2  1  7  0  0  6  2  0  2 11  2  4  2  2  2  0  2  1
##  [649]  1  4  0  1  3  7  6  0 22  1 10  8  1  7 20  0  5 15  0  6 17  4  3  0
##  [673]  0  6  2  1  1  7  0  4  1  6  2  6  0  3  0 16  3 13  1  0  3 24  1  2
##  [697]  4  6  1 11  8  0  3  2  3  8  1  8  0  4  0  9  5  9  8  0 11  5  0  6
##  [721]  6 14  0  4  2  3  1  6  5  3  1  3  1 24  1  8  7  3  2  0  4  0  5  5
##  [745]  0  6  3  2  0  7  3  4 14  0  2  3  0  6  5  4 12 11  3  3  0  2  0  4
##  [769]  4  1  6  1  0  4  1  0  0  5  2  6  2  3 12 17  0  5  9  2 12  0  0  1
##  [793]  0  2  2  1  3  0 13 10  1  0  2  0  9  1  6  2  7  5  3  1  0  1  1  2
##  [817]  2  0  2  7  6  3  0  0  0  2  4  0 11  0  2  0  7  0  0  2 12  7  2  0
##  [841]  4  4  1  3  7  1  4  1  1  0  1  2  0  3  7  6  1  1  2 23  5  1  2  0
##  [865]  5  2  2  3  5  0  1  3 18  4 10  3  0  1 23  3  2  6  2  2  3  4  1  0
##  [889]  5  0  1  1  2  4  1 18  4  1  3  0  1  2  1  4  4  0  6  9  0  0  2  3
##  [913]  0  8 11  0  4  1  1  3  3 13  0 15  7  4 11 12  0  2  7  0  1  3  3  1
```

```
## [937]  1  1  7  6  0  2  0  3  4  0  7  2 13  8 12  4 19  1  0  2  1  1  1  3
## [961]  7  8  4 10  7  2 12  0  2  1  1  2  2  1  0  4  7  2  7  0  7  3 14  1
## [985] 13  0  0  4 11  7  3  7  3  1  2  4  2  8  0  4
```

3. Calculate some basic statistics:

```r
mean_x = mean(x)

var_x = var(x)

sd_x = sd(x)
```

4. Print the results in item 3 with the following output (string):

Number of trials required to achieve first success:

Mean (in 2 decimal places):

Variance (in 2 decimal places):

Standard deviation (in 2 decimal places):

```r
cat("Number of trials required to achieve first success:\n" )
```

```
## Number of trials required to achieve first success:
```

```r
cat("Mean (in 2 decimal places):", round(mean_x, 2), "\n")
```

```
## Mean (in 2 decimal places): 3.79
```

```r
cat("Variance (in 2 decimal places):", round(var_x, 2), "\n")
```

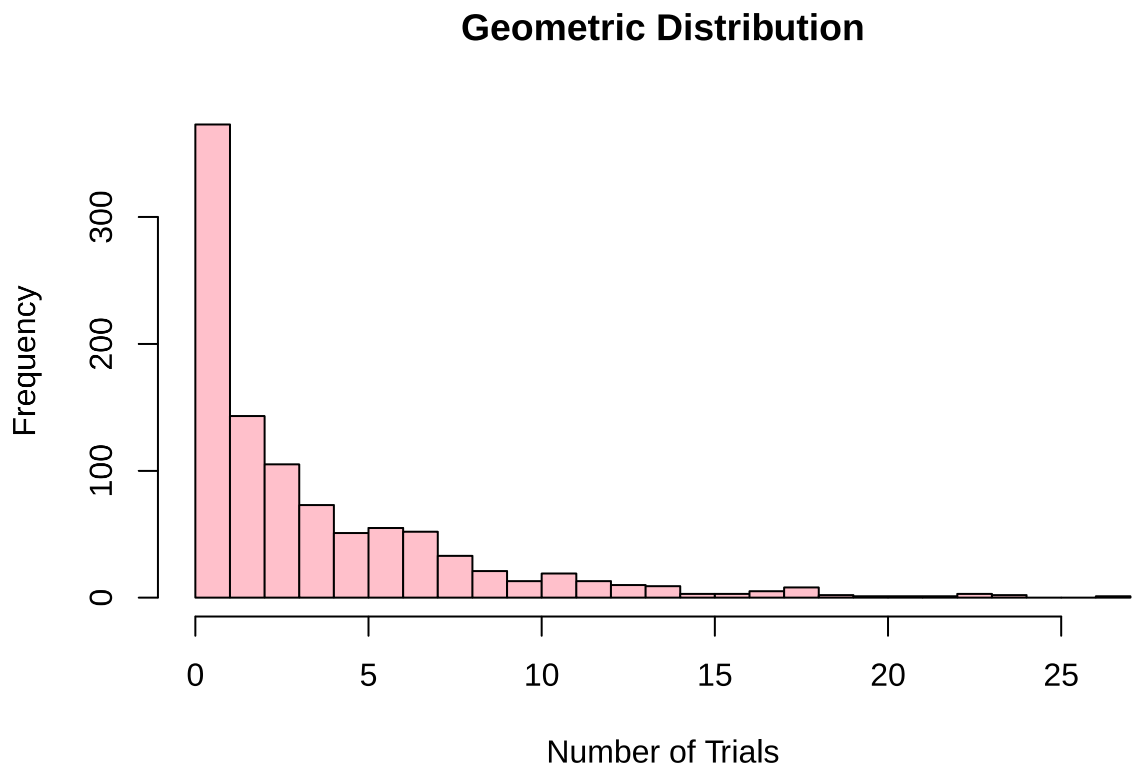
```
## Variance (in 2 decimal places): 18.08
```

```r
cat("Standard deviation ( in 2 decimal places):", round(sd_x, 2), "\n")
```

```
## Standard deviation ( in 2 decimal places): 4.25
```

5. Plot the histogram of the results.

```r
hist(x, breaks = 30, main = "Geometric Distribution",
     xlab = "Number of Trials", col = "pink")
```



II. Hypergeometric Distribution. Consider a plant manufacturing IC chips of which 10% are expected to be defective. The chips are packed in boxes for export. Before transportation, a sample is drawn from each box. Estimate the probability that the sample contains more than 10% defectives, when:

1. A sample of 10 is selected from a box of 40;

```
N = 40 # Total Population
K = 4 # 10% expected to be defective
n = 10  # Sample

# We want to find P(X > 1) which is = 1 - P(X = 0) - P(X = 1),
# first find  P_X_0 and P_X_1

P_X_0 = choose(K, 0) * choose(N - K, n - 0) / choose(N, n)
P_X_0
```

```
## [1] 0.2998687
```

```
P_X_1 = choose(K, 1) * choose(N - K, n - 1) / choose(N, n)
P_X_1
```

```
## [1] 0.4442499
```

```r
P_X_g_1 = 1 - P_X_0 - P_X_1
cat("The probability that a sample contains more than 10% defectives when a sample of 10
    is selected from a box of 40 is", P_X_g_1, "or", round(100*P_X_g_1, 2), "%.")
```

```
## The probability that a sample contains more than 10% defectives when a sample of 10
##     is selected from a box of 40 is 0.2558814 or 25.59 %.
```

2. A sample of 10 is selected from a box of 5000.

```r
# Perform the same calculations but with different values
N = 5000 # Total Population
K = 500 # 10% expected to be defective
n = 10  # Sample

# We want to find P(X > 1) which is = 1 - P(X = 0) - P(X = 1),
# first find  P_X_0 and P_X_1

P_X_0 = choose(K, 0) * choose(N - K, n - 0) / choose(N, n)
P_X_0
```

```
## [1] 0.3483295
```

```r
P_X_1 = choose(K, 1) * choose(N - K, n - 1) / choose(N, n)
P_X_1
```

```
## [1] 0.3878084
```

```r
P_X_g_1 = 1 - P_X_0 - P_X_1
cat("The probability that a sample contains more than 10% defectives when a sample of 10
    is selected from a box of 40 is", P_X_g_1, "or", round(100*P_X_g_1, 2), "%.")
```

```
## The probability that a sample contains more than 10% defectives when a sample of 10
##     is selected from a box of 40 is 0.2638622 or 26.39 %.
```

Github Link: https://github.com/SylTana/APM1110-QUIJANO-JULIAN_PHILIP/tree/main/FA6