Statistical Methods for Data Science
Mini Project #1

Group Members: Arpita Kumane, Harshali Dube

Contribution: Both the team members collaborated to learn R and worked on the project together to write the R scripts. We analyzed, discussed, and efficiently worked to submit the two questions. We both checked the correctness of the output and analyzed the values to come up with the comparison results.

Question 1:

A)

$P(T > 15) = 1 - P(T <= 15)$

$= 1 - F(T <= 15)$

$= 1 - \int_0^{15} f_T(t)\, dt$

$= 1 - \int_0^{15} (0.2e - 0.1t - 0.2e - 0.2t)$

$= 1 - {}^{15}[0.2(((e^{-0.1t})/(-0.1)) - ((e^{-0.2t})/(-0.2)))$

$= 1 - {}^{15}[-2e^{-0.1t} + e^{-0.2t}]$

$= 1 - [(-2e^{-0.1x15} + e^{-0.2x15}) - (-2e^{-0.1x0} + e^{-0.2x0})]$

$= 1 - [-2e^{-1.5} + e^{-3} + 2e^{0} - e^{0}]$

$= 1 - [e^{-3} - 2e^{-1.5} + 1]$

$= 1 - [0.049787 - 0.446260 + 1]$

$= 1 - 0.603527$

$= 0.396473$

B)

(i) Use the following steps to take a Monte Carlo approach to compute E(T) and P(T > 15).

```
> #question 1b(i)
> #Sample 10000
> T <- (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5)))
```

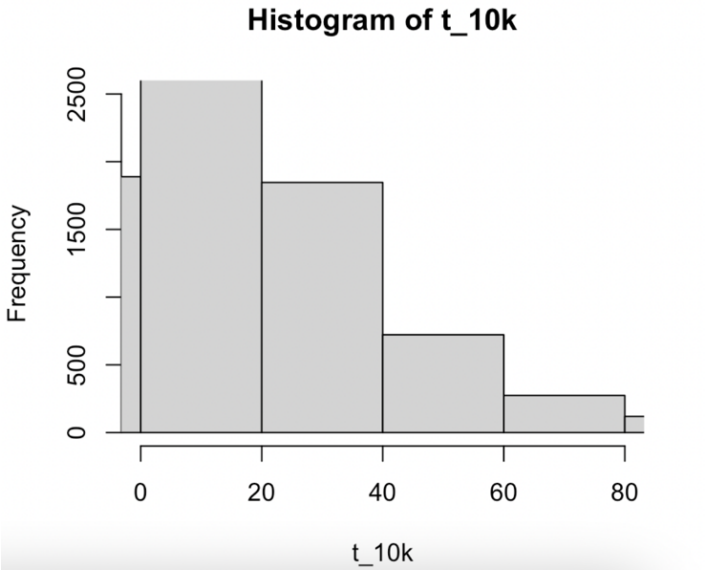
Values	
T	15.53849914…

ii) Repeat the previous step 10,000 times. This will give you 10,000 draws from the distribution of T. Try to avoid 'for' loop. Use 'replicate' function instead. Save these draws for reuse in later steps.

```
> t_10k = replicate(10000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
```

t_10k	num [1:10000] 0.771 15.594 55.982 15.583 5.231 .…

iii) Histogram of t_10k



iv) As we previously derived, the value for the probability density function was 15 and the value gained form Monte Carlo simulation is 15.64521 which is close enough.

```
> mean(t_10k)
[1] 15.64521
```

v) The probability that the satellite lasts more than 15 years.

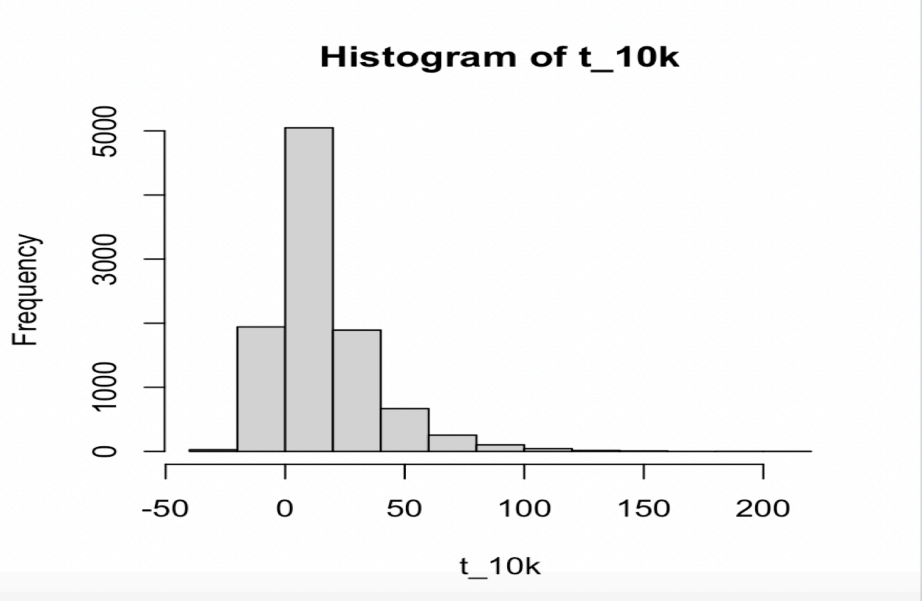
```
> 1 - pexp(15, rate = 1/mean(t_10k))
[1] 0.383368
```

Comparison: The probabilities are slightly different as the mean is different and the sample size is 10,000 random variables.

vi) Repeat the above process of obtaining an estimate of E(T) and an estimate of the probability four more times.

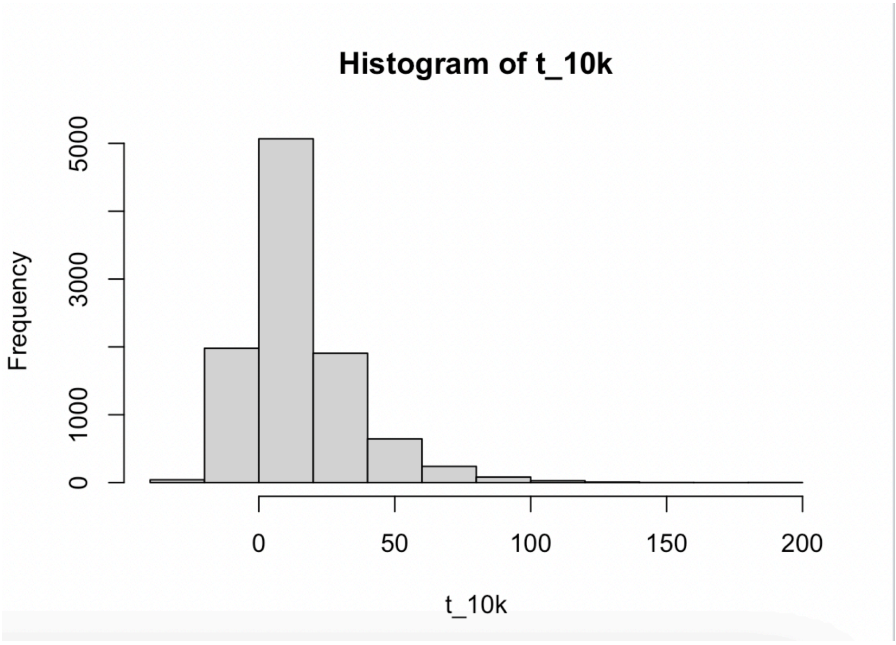
Test 1:

```
> #Test No.1
> t_10k = replicate(10000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(t_10k)
> mean(t_10k)
[1] 15.38393
> 1 - pexp(15, rate = 1/mean(t_10k))
[1] 0.377176
```



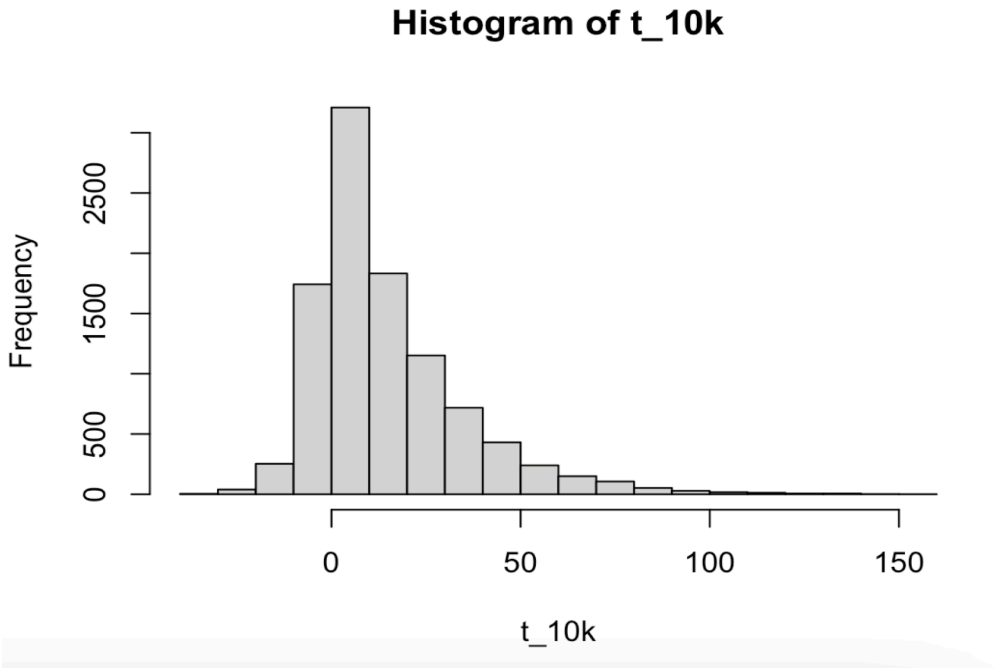
Test 2:

```
> #Test No.2
> t_10k = replicate(10000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_10k)
> mean(t_10k)
[1] 14.64536
> 1 - pexp(15, rate = 1/mean(t_10k))
[1] 0.3590781
```



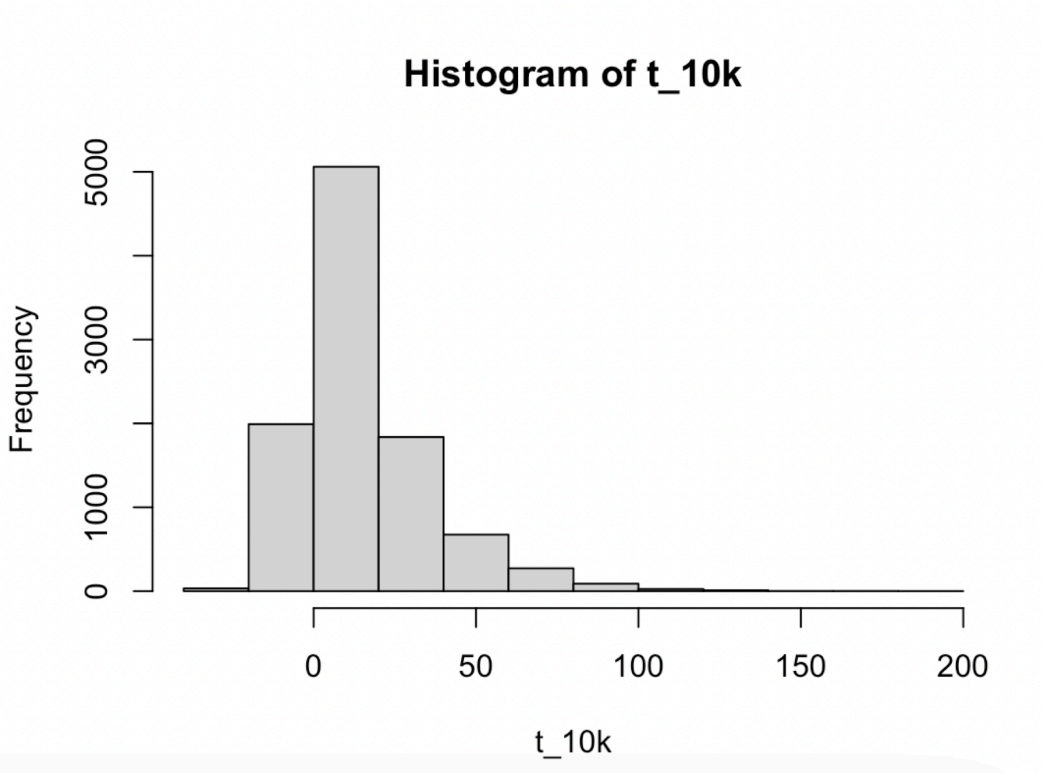
Test 3:

```
> #Test No.3
> t_10k = replicate(10000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_10k)
> mean(t_10k)
[1] 14.63264
> 1 - pexp(15, rate = 1/mean(t_10k))
[1] 0.3587586
```



Test 4:

```
> #Test No.4
> t_10k = replicate(10000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_10k)
> mean(t_10k)
[1] 14.74262
> 1 - pexp(15, rate = 1/mean(t_10k))
[1] 0.3615127
```



a. Comparison Table:

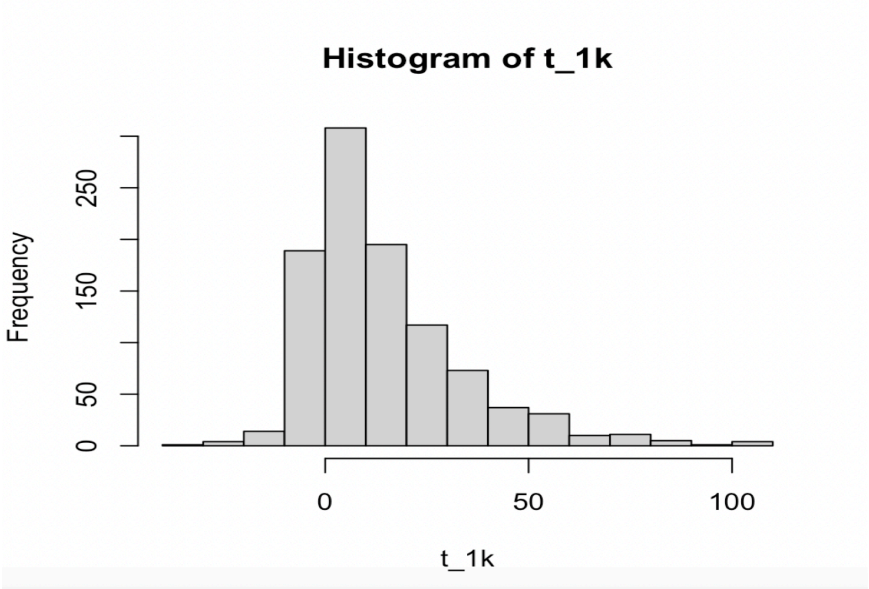
Test for Sample Size 10000	E(T)	P(T>15)
Test 1	15.64521	0.38336
Test 2	15.38393	0.37717
Test 3	14.64535	0.35906
Test 4	14.63264	0.35875
Test 5	14.74262	0.36151

As seen in the comparison table above, the E(T) value is close to the numerical value 15 with slight variations. Also, the value of P(T>15) has slightest variations as these values are approximately same to the one calculated on question (1a). Thus, this proves the Central Limit Theorem correctly.

C. Repeat part (vi) five times using 1,000 and 100,000 Monte Carlo replications.

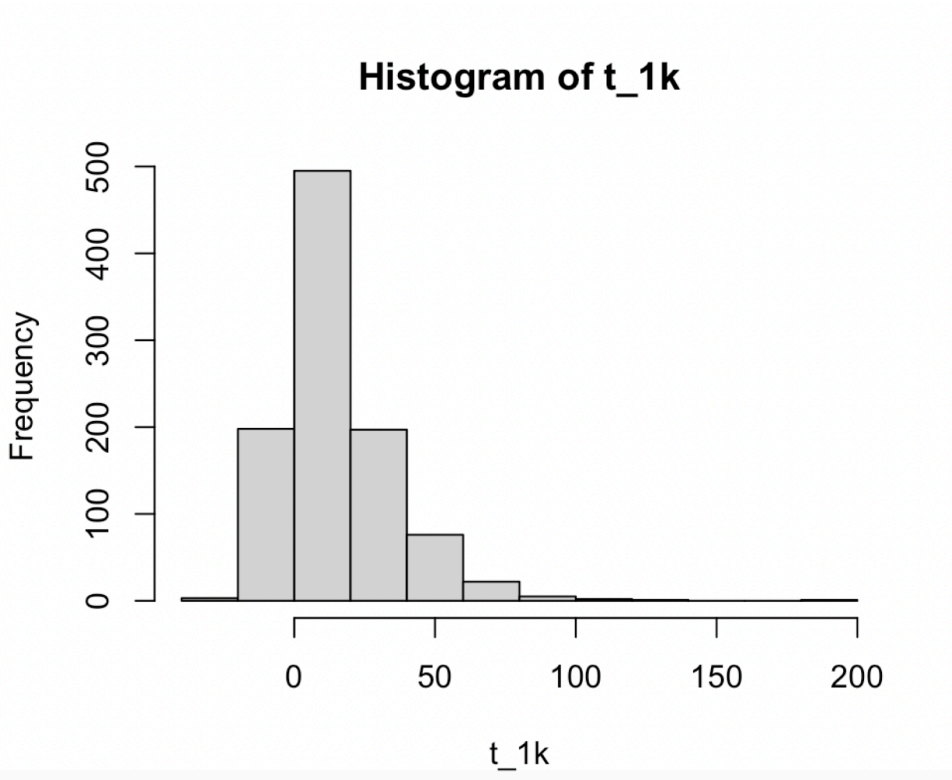
==> Test series for Sample size: 1000
Test 1:

```
> #Test No.1
> #Sample: 1000
> t_1k = replicate(1000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_1k)
> mean(t_1k)
[1] 14.46446
> 1 - pexp(15, rate = 1/mean(t_1k))
[1] 0.3545079
```



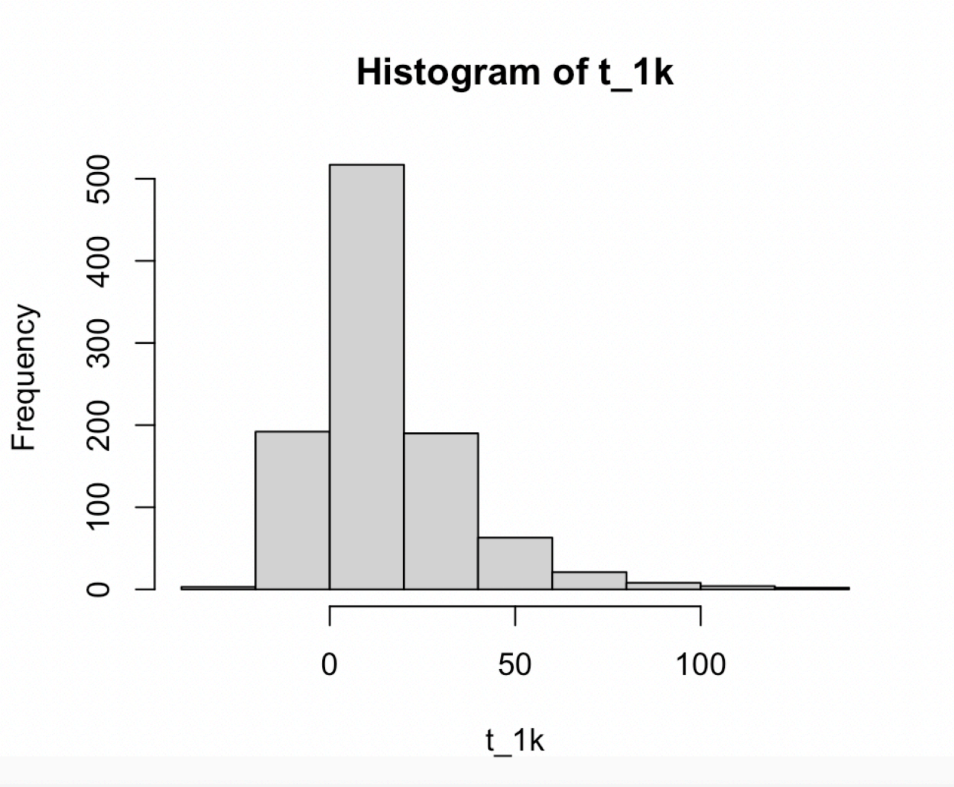
Test 2:

```
> #Test No.2
> #Sample: 1000
> t_1k = replicate(1000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_1k)
> mean(t_1k)
[1] 14.47527
> 1 - pexp(15, rate = 1/mean(t_1k))
[1] 0.3547826
```



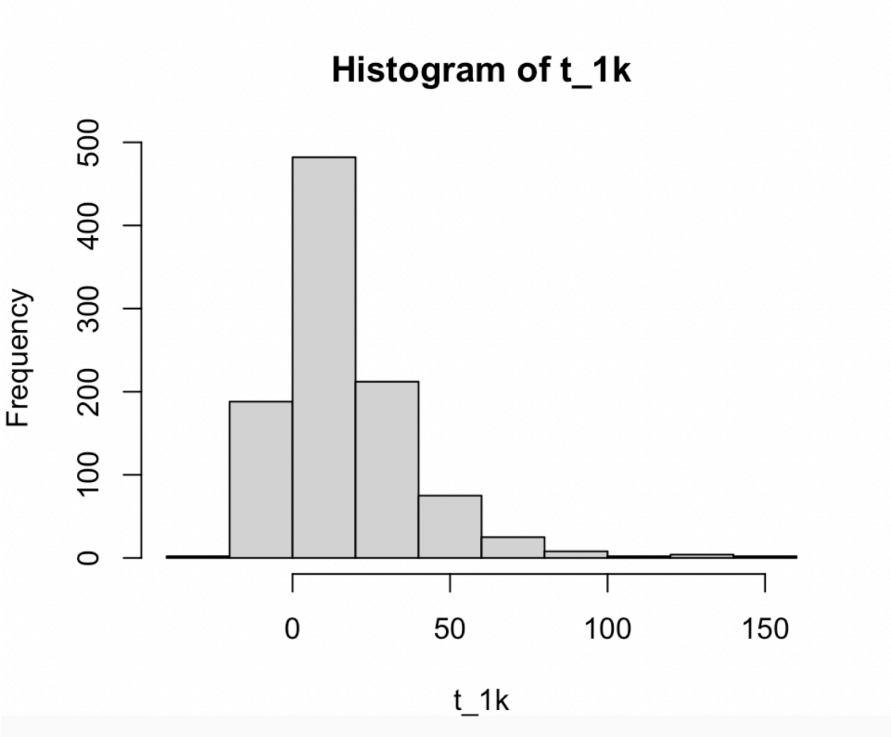
Test 3:

```
> #Test No.3
> #Sample: 1000
> t_1k = replicate(1000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_1k)
> mean(t_1k)
[1] 14.92215
> 1 - pexp(15, rate = 1/mean(t_1k))
[1] 0.3659652
```



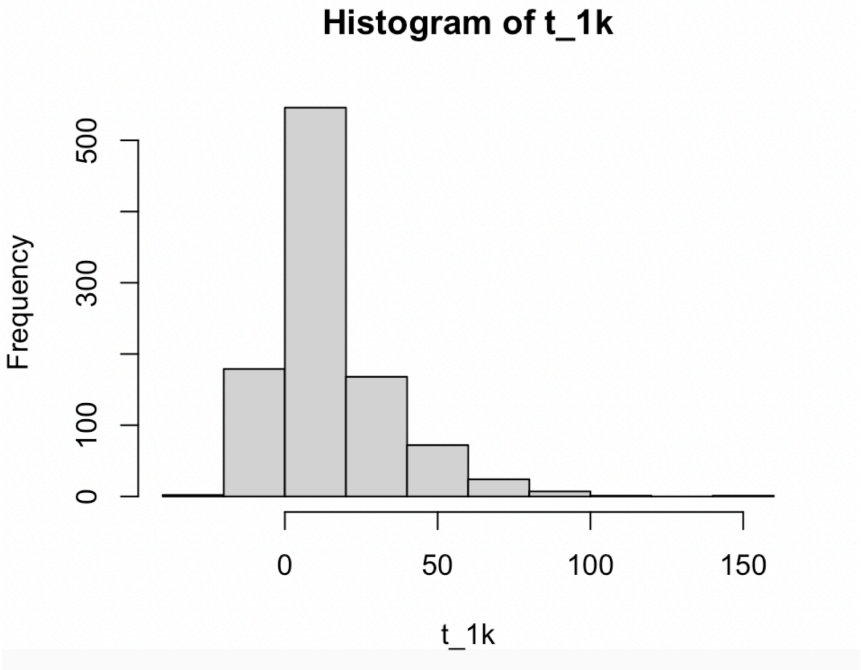
Test 4:

```
> #Test No.4
> #Sample: 1000
> t_1k = replicate(1000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> mean(t_1k)
[1] 15.90968
> hist(x=t_1k)
> 1 - pexp(15, rate = 1/mean(t_1k))
[1] 0.3895269
```



Test 5:

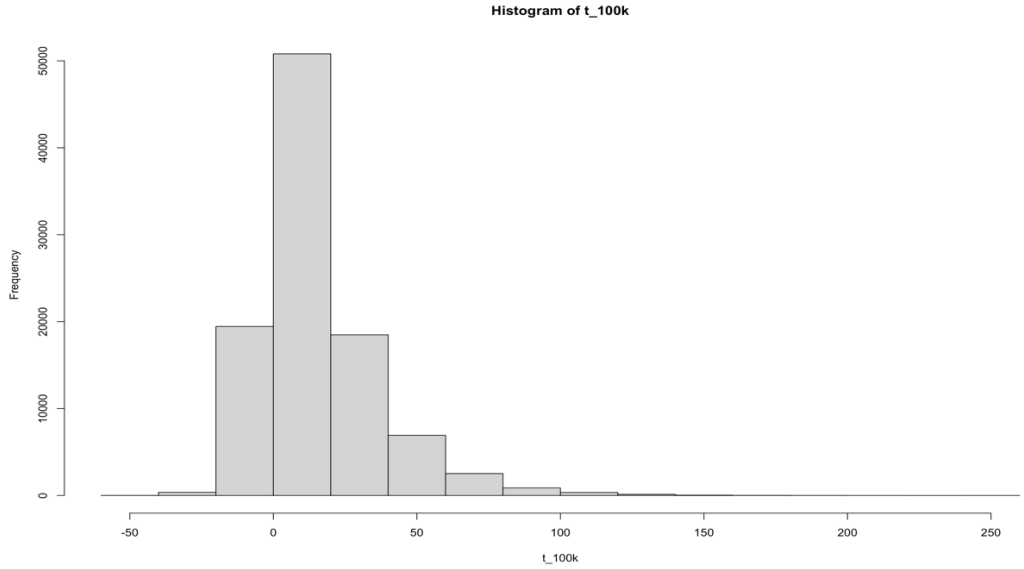
```
> #Test No.5
> #Sample: 1000
> t_1k = replicate(1000, (2*rexp(n=1, rate=1/10) - (rexp(n=1, rate=1/5))))
> hist(x=t_1k)
> mean(t_1k)
[1] 14.67372
> 1 - pexp(15, rate = 1/mean(t_1k))
[1] 0.3597896
```



==> Sample Test Size is now 100000:

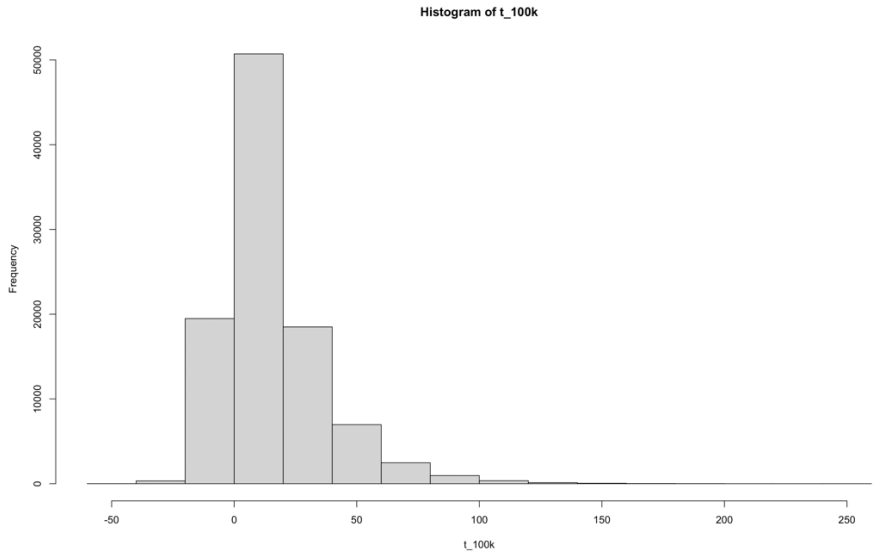
Test 1.

```
> t_100k = replicate(100000,(2*rexp(n=1, rate=1/10) -
 (rexp(n=1, rate=1/5))))
> hist(x=t_100k)
> mean(t_100k)
[1] 14.9861
> 1-pexp(15,rate = 1/mean(t_100k))
[1] 0.3675384
>
```



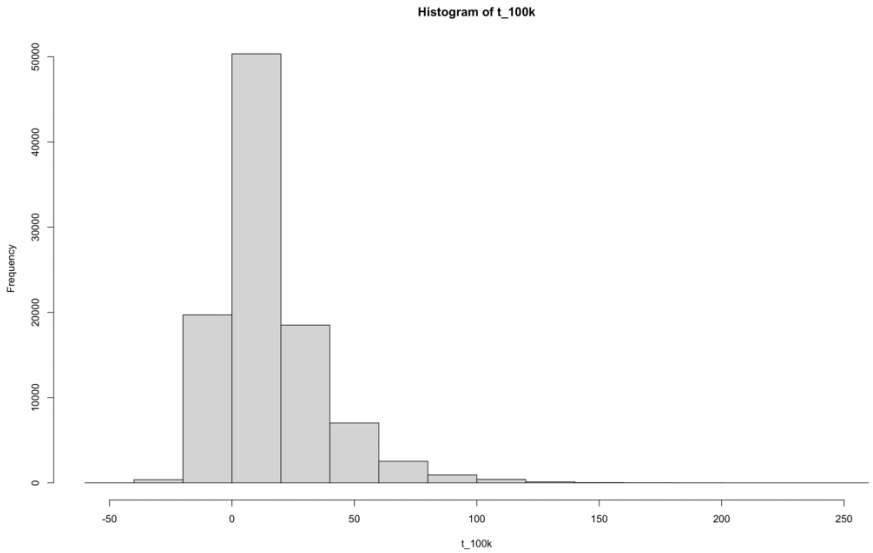
Test 2:

```
> t_100k = replicate(100000,(2*rexp(n=1, rate=1/10) -
 (rexp(n=1, rate=1/5))))
> hist(x=t_100k)
> mean(t_100k)
[1] 15.11322
> 1-pexp(15,rate = 1/mean(t_100k))
[1] 0.3706457
>
```



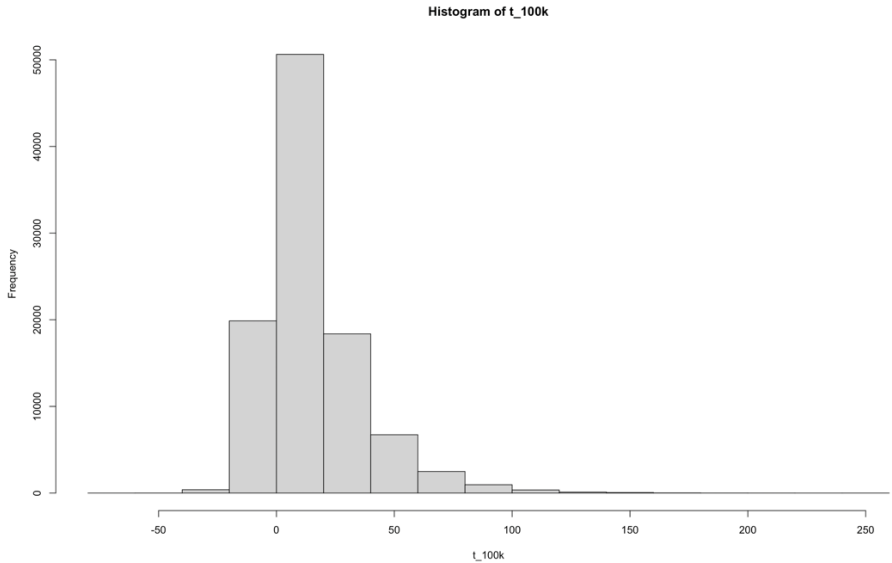
Test 3:

```
> t_100k = replicate(100000,(2*rexp(n=1, rate=1/10) -
 (rexp(n=1, rate=1/5))))
> hist(x=t_100k)
> mean(t_100k)
[1] 15.03577
> 1-pexp(15,rate = 1/mean(t_100k))
[1] 0.3687557
>
```



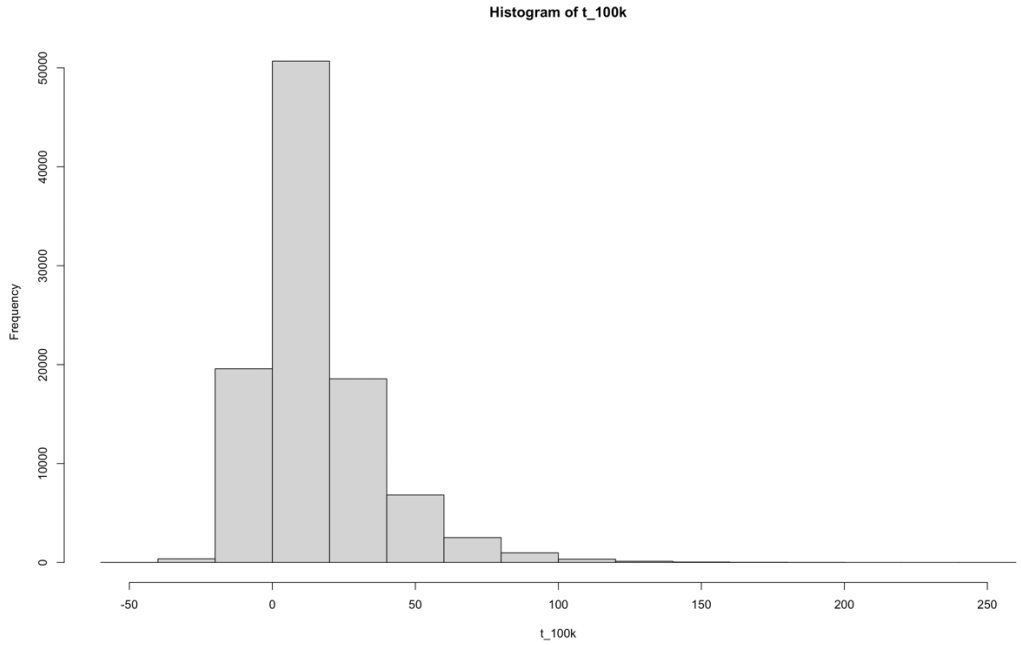
Test 4:

```
> t_100k = replicate(100000,(2*rexp(n=1, rate=1/10) -
 (rexp(n=1, rate=1/5))))
> hist(x=t_100k)
> mean(t_100k)
[1] 14.91038
> 1-pexp(15,rate = 1/mean(t_100k))
[1] 0.3656749
>
```



Test 5:

```
> t_100k = replicate(100000,(2*rexp(n=1, rate=1/10) -
 (rexp(n=1, rate=1/5))))
> hist(x=t_100k)
> mean(t_100k)
[1] 14.99876
> 1-pexp(15,rate = 1/mean(t_100k))
[1] 0.3678491
>
```



Comparison Table For Sample size 1000:

Test for Sample Size is 1000	E(T)	P(T>15)
Test 1	14.46446	0.354507
Test 2	14.47527	0.354782
Test 3	14.92215	0.365965
Test 4	15.90968	0.389526
Test 5	14.67372	0.359789

Comparison Table For sample size 10000:

Test for Sample Size 10000	E(T)	P(T>15)
Test 1	15.64521	0.38336
Test 2	15.38393	0.37717
Test 3	14.64535	0.35906
Test 4	14.63264	0.35875
Test 5	14.74262	0.36151

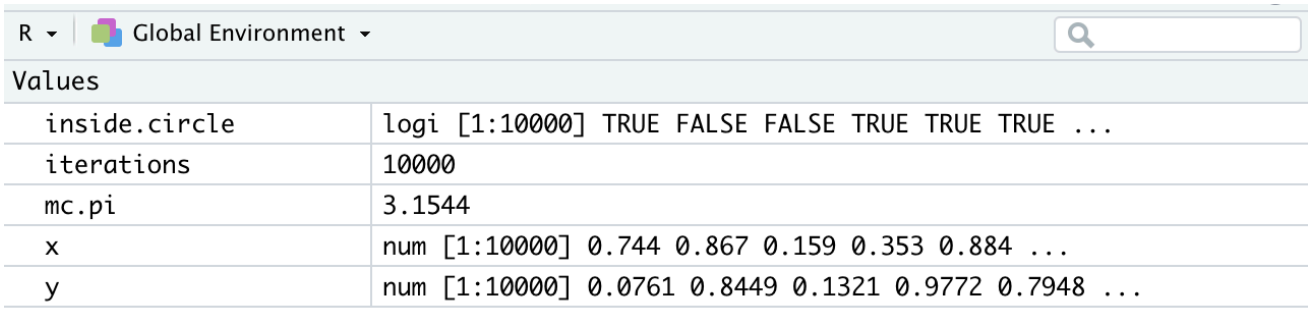
Comparison Table For sample size 100000:

Test for Sample Size is 100000	E(T)	P(T>15)
Test 1	14.9861	0.36753
Test 2	15.1132	0.37064
Test 3	15.0357	0.36875
Test 4	14.9103	0.36567
Test 5	14.9887	0.36784

==> As we observe the comparison tables above, it is evident that as the sample size increases, the variation in E(T) and P(T>15) values decreases and this correlates with the definition of the Central Limit Theorem. More the number of cases, the calculated value is inclined towards the given value.
As we all know that the Monte Carlo approach is more accurate when many replications are considered. This is because of the standard deviation of a less sample size is more when compared to the sample size which has many number of draws

Question 2: Use a Monte Carlo approach to estimate the value of π based on 10, 000 replications. [Ignorable hint: First, get a relation between π and the probability that a randomly selected point in a unit square with coordinates — (0, 0), (0, 1), (1, 0), and (1, 1) — falls in a circle with center (0.5, 0.5) inscribed in the square. Then, estimate this probability, and go from there.]

```
> iterations <- 10000
> x <- runif (iterations, min=0, max=1)
> y <- runif (iterations, min=0, max=1)
> inside.circle <- (x-0.5)^2 + (y-0.5)^2 <= 0.5^2
> mc.pi <- (sum(inside.circle)/iterations)*4
> mc.pi
[1] 3.1544
>
```



The calculated value of pi is 3.154 which is very close to 3.146 which is the true value of pi.