Mini Project : 1

Members :
1. Puru Jaiswal (NetID : PXJ200018)
2. Sara Tabassi (NetID : SXT200083)

Contribution of each group member:

Both the team members worked together to finish the project. Collaborated to learn R and then write the code. Puru worked to check the accuracy of the code and Sara worked to document the report. Both members worked efficiently to complete the project requirements.

Question 1 .a)

Q1)

Density function:

$$f_T(t) = \begin{cases} 0.2\exp(-0.1t) - 0.2\exp(-0.2t) & 0 \le t < \infty \\ 0 & \text{otherwise} \end{cases}$$

$E(T) = 15$ yrs.

(a) Use the above density function to analytically compute the probability that the lifetime of the satellite exceeds 15 yrs.

$$P(T>15) = 1 - P(T \le 15)$$

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

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

$$= 1 - \left[-2e^{-0.1t} + e^{-0.2t}\right]_0^{15}$$

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

$$= 1 - 0.603527$$

$$= 0.396473$$

b)

i) Simulate one draw of the block lifetimes XA and XB. Use these draws to simulate one draw of the satellite lifetime T.

```
pdf <- function(t){ return (0.2*(exp(-0.1*t)-exp(-0.2*t)))}
```

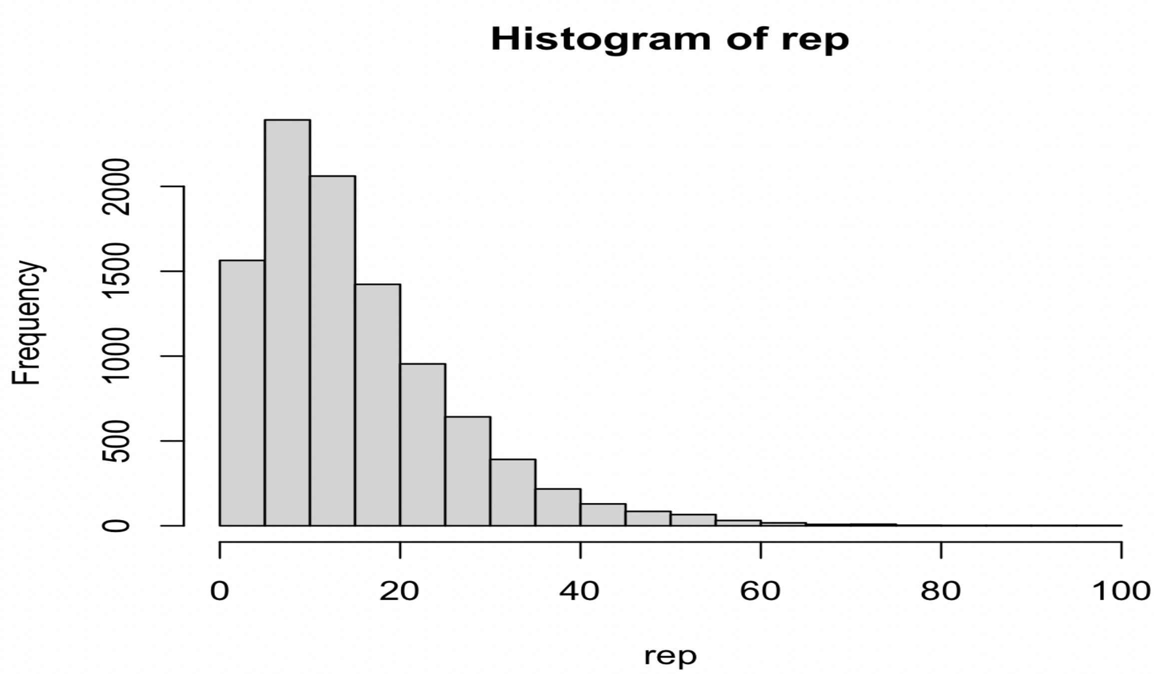
ii) Repeat the previous step 10,000 times. This will give you 10,000 draws from the distribution of T. Try to avoid the 'for' loop. Use the 'replicate' function instead. Save these draws for reuse in later steps. [Bonus: 1 bonus point for not taking more than 1 line of code for steps (i) and (ii).]

```
rep = replicate(10000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
```

Values	
rep	num [1:10000] 11.68 5.78 5.84 8.95 15.07 ...

iii) Make a histogram of the draws of T using the 'hist' function. Superimpose the density function given above. Try using the 'curve' function for drawing the density. Note what you see.

```
hist(rep)
```



iv) Use the saved draws to estimate E(T). Compare your answer with the exact answer given above.

E(T) time given above is 15 and using simulation : **mean(rep) : 14.8873** which is comparable.

v) Use the saved draws to estimate the probability that the satellite lasts more than 15 years. Compare with the exact answer computed in part (a).

greater_than_15 <- rep[rep>15]

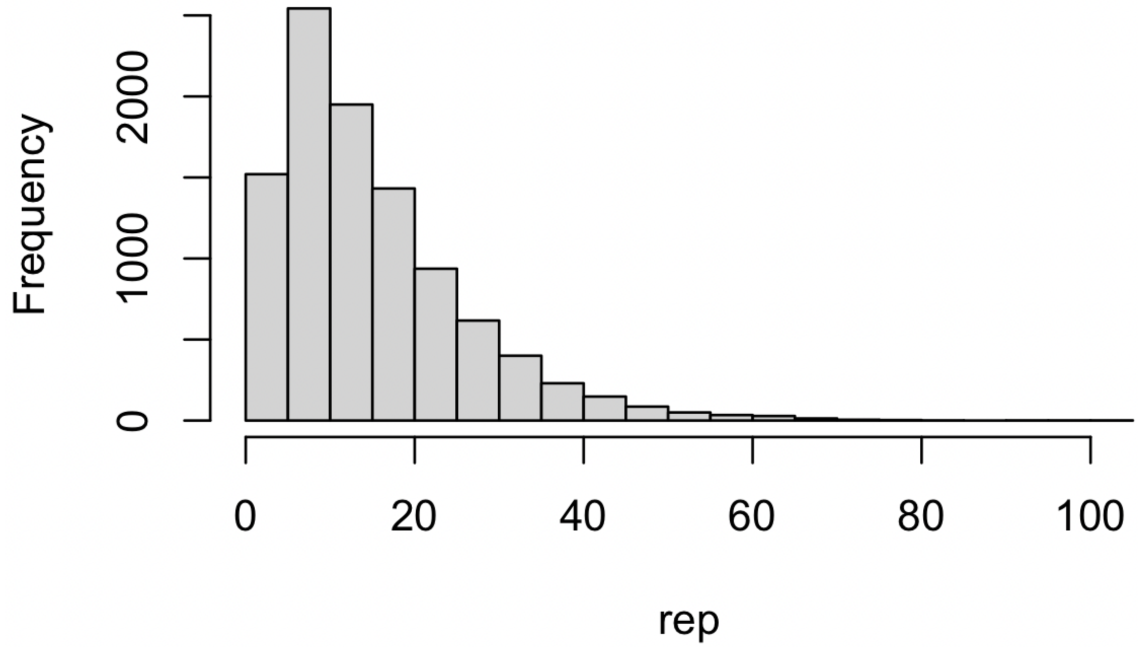
prob <- length(greater_than_15)/length(rep)

Res := 0.3888

There is a slight difference in comparison with the original probability.

vi) Test 1

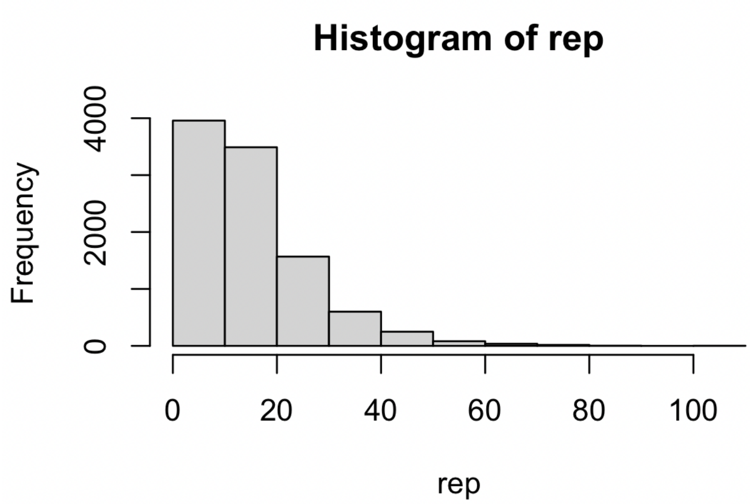
```
> rep <- replicate(10000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
>
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
>
> prob
[1] 0.3987
> mean(rep)
[1] 15.01023
```



Test 2

```
> rep <- replicate(10000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> mean(rep)
[1] 15.11615
> prob
[1] 0.3996
> |
```

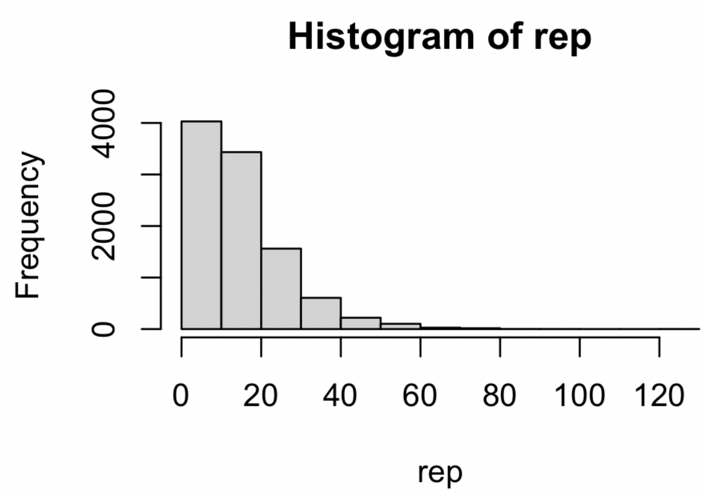
hist(rep)



Test 3

```
> rep <- replicate(10000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> mean(rep)
[1] 15.08679
> prob
[1] 0.3997
```

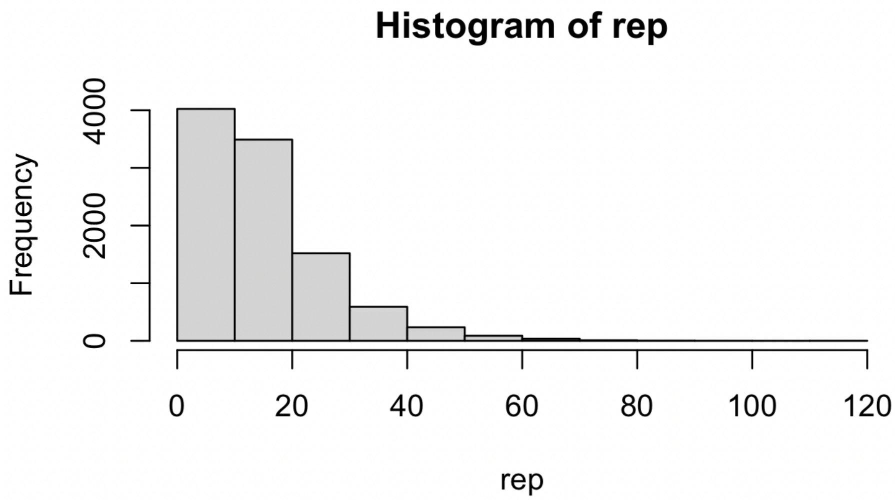
hist(rep)



Test 4

```
> rep <- replicate(10000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> mean(rep)
[1] 14.98108
> prob
[1] 0.3909
```

hist(rep)



Comparison Table

Test for Sample Size 10000	E(T)	P(T>15)
Test 1	15.010	0.3987
Test 2	15.116	0.399
Test 3	15.086	0.399
Test 4	14.981	0.390

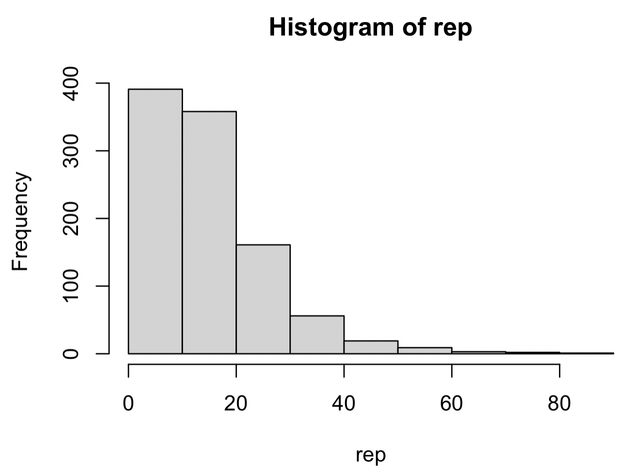
With sample size 10000, E(T) is closer to the numerical value 15 with < a slight variation and P(T>15) also has a slight variation hence the central limit theorem holds.

c)

Sample Size 1000

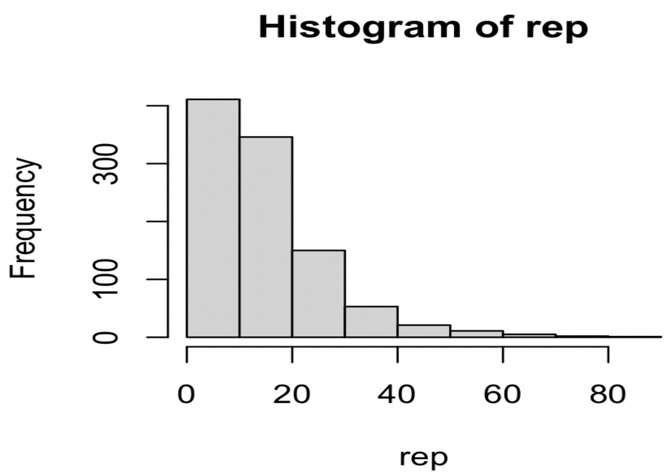
Test 1:

```
> rep <- replicate(1000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.406
> mean(rep)
[1] 15.15938
```



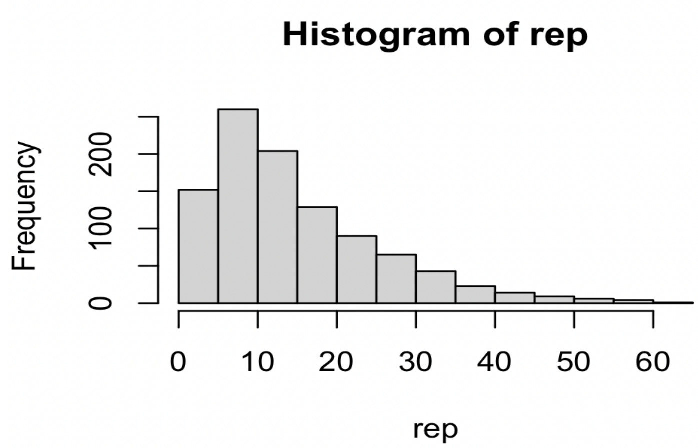
Test 2:

```
> rep <- replicate(1000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.387
> mean(rep)
[1] 14.95306
```



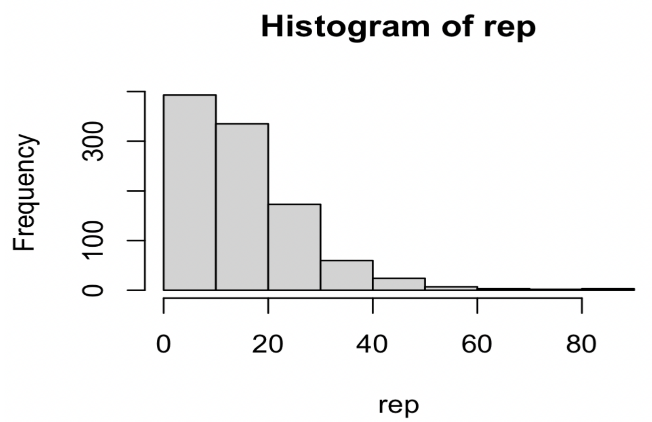
Test 3 :

```
> rep <- replicate(1000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> hist(rep)
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> mean(rep)
[1] 14.77964
> prob
[1] 0.384
```



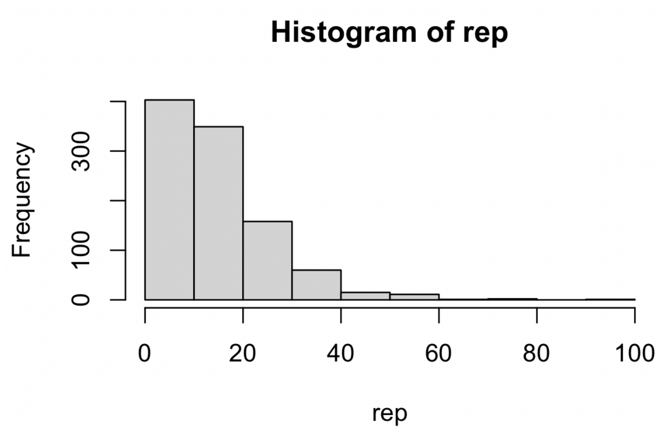
Test 4 :

```
> rep <- replicate(1000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> hist(rep)
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.426
> mean(rep)
[1] 15.35512
```



Test 5 :

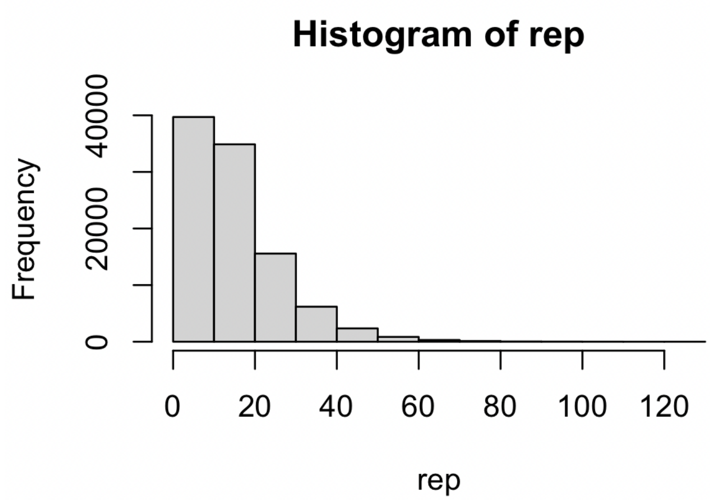
```
> rep <- replicate(1000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> hist(rep)
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.396
> mean(rep)
[1] 14.84159
```



Sample Size 100000

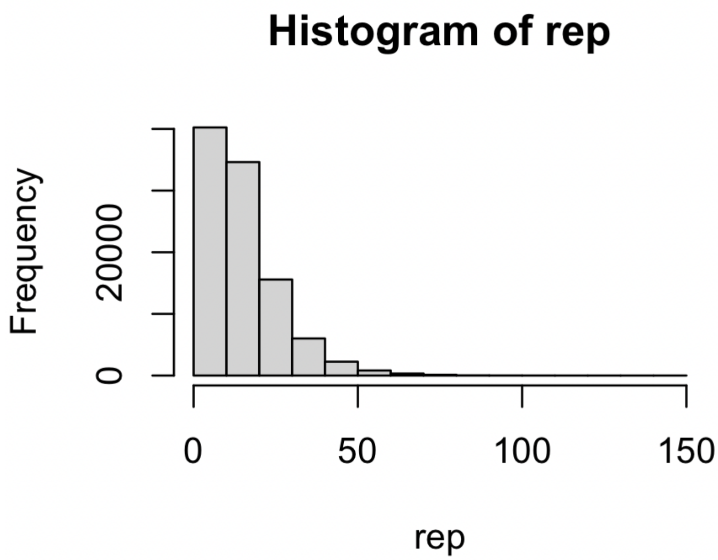
Test 1 :

```
> rep <- replicate(100000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> mean(rep)
[1] 15.07951
> mean(rep)
[1] 15.07951
> prob
[1] 0.3994
```



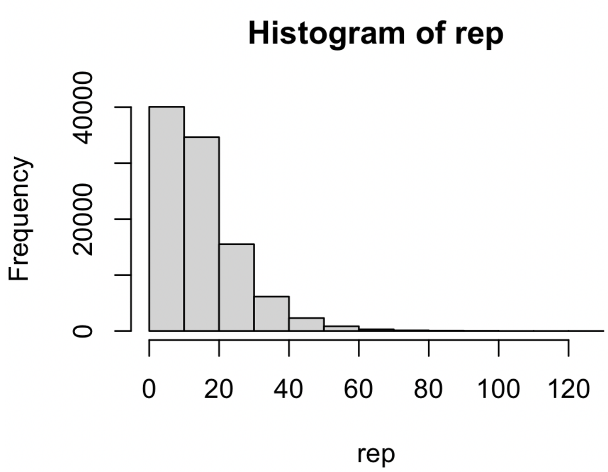
Test 2 :

```
> rep <- replicate(100000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.39413
> mean(rep)
[1] 14.93449
```



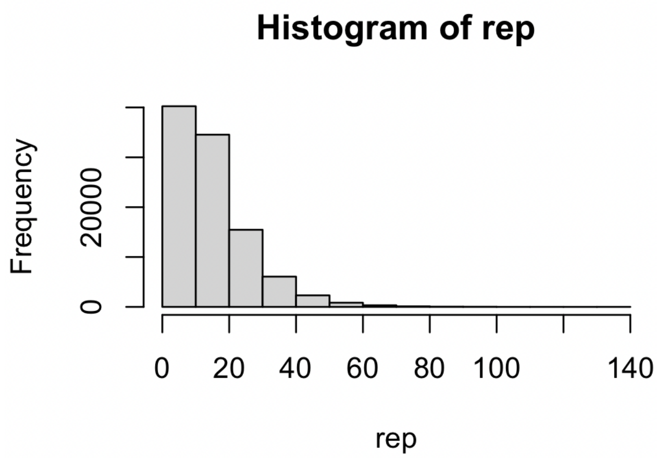
Test 3 :

```
> rep <- replicate(100000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.39512
> mean(rep)
[1] 15.01244
> hist(rep)
```



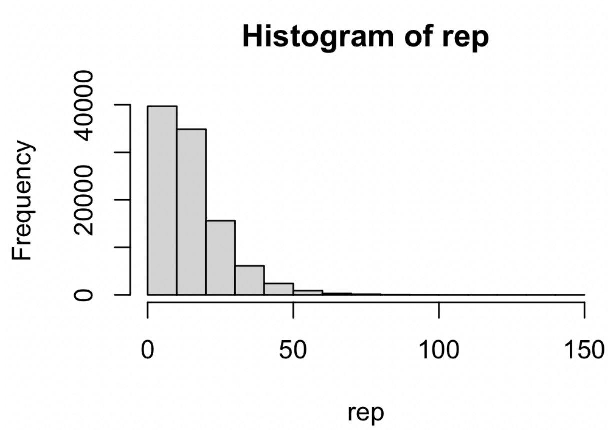
Test 4 :

```
> rep <- replicate(100000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.39454
> mean(rep)
[1] 14.98633
> hist(rep)
```



Test 5 :

```
> rep <- replicate(100000, max(rexp(n=1, rate=1/10),rexp(n=1, rate=1/10)))
> greater_than_15 <- rep[rep>15]
> prob <- length(greater_than_15)/length(rep)
> prob
[1] 0.39753
> mean(rep)
[1] 15.0692
> hist(rep)
```



Comparison Table :

Test for Sample Size 10000	**E(T)**	**P(T>15)**
Test 1	**14.8873**	**0.3888**
Test 2	**15.01023**	**0.3987**
Test 3	**15.11615**	**0.3996**
Test 4	**15.08679**	**0.3997**
Test 5	**14.98108**	**0.3909**

Sample Size 1000	**E(T)**	**P(T>15)**
Test 1	**15.15938**	**0.406**
Test 2	**14.95306**	**0.387**
Test 3	**14.77964**	**0.384**
Test 4	**15.35512**	**0.426**
Test 5	**14.84159**	**0.396**

Sample Size 100000	**E(T)**	**P(T>15)**
Test 1	**15.07951**	**0.3994**
Test 2	**14.93449**	**0.39413**
Test 3	**15.01244**	**0.39512**
Test 4	**14.98633**	**0.39454**
Test 5	**15.0692**	**0.39753**

As the sample size becomes larger the variation begins to reduce and this relates with the CLM.

2. (10 points) 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.]

A. We have a square with the following coordinates - (0,0), (0,1), (1,0) and (1,1) and a circle inscribed within the square with the center having the coordinates - (0.5,0.5).

Calculating the probability of a point falling inside the circle within the square -

Area of Circle / Area of Square = π r2 / 1 = π/4.

The above is an approximation of - **(Number of points falling inside the circle) / (Total number of points).**

We now generate a large number of points inside the square and calculate the number of points inside the circle.

Therefore, we have **π=4*(Number of points falling inside the circle) / (Total number of points)**

Monte carlo Approach for evaluating the value of π :

```
> x = runif(1000, min=0, max=1)
> y = runif(1000, min=0, max=1)
> dist = sqrt((x-(0.5))^2+(y-(0.5))^2)
> mc.pi=(4*sum(dist<=0.5))/10000
> print(mc.pi)
[1] 0.3108
>
```

Final Output after running 10,000 Replications on R:

<u>π = 3.108</u>