##### INDEPENDENT MONTE CARLO SIMULATION OF A POWER PLANT ============

daily\_power=function()

{

supply\_info=matrix(c(30,0,0,0,20,0,3,7,3,10,0,0,6,5,0,20,0,10,0,0,10,6,5,3,10,0,0,6,7,0,15,8,0,0,0,10,9,6,2,10,0,20,6,8,0,0,0,0,80,0,0,0,8,2,10,0,0,6,6,0,0,20,0,0,0,0,15,4,4,0,0,8,6,5,10,21,0,0,0,0,15,0,5,4,0,0,0,6,4,10,14,0,10,0,10,0,11,5,5,0,0,0,6,5,10,0,41,0,0,0,0,7,5,5,0,0,15,6,5,10,0,0,25,0,0,7,0,5,10,10,0,0,8,8,0,0,15,5,0,15,8,10,8,4,0,0,0,6,5,15,0,0,10,0,0,0,0,4,8,15,0,41,0,8,0,0,7,0,0,10,0,12,10,5,0,0,0,6,0,10,0,0,10,0,0,0,0,5,5,0,0,9,0,10,0,0,0,0,0,0,0,0,5,5,15,14,0,6,4,0,0,0,10,0,0,9,14,4,5,0,21,0,0,0,15,0,9,0,20,10,6,0,4,4,0,0,0,6,2,0,0,0,0,0,10,20,0,2,8,0,0,0,4,2,20,0,0,10,0,10,0,0,2,6,0,15,7,6,2,0,0,0,10,0,0,15,0,3,5,0,20,0,4,3,0,0,0,0,0,15,0,13,3,7,20,30,0,6,0,0),nrow=15,ncol=20)

station\_info=matrix(c(1,1,1,9,6,6,1,3,2,8,11,8,18,6,4,13.60,12.00,5.60,0.76,0.65,1.12,5.53,5.53,7.35,0.78,1.09,0.78,1.45,7.74,0.95,0.16,0.16,0.14,0.02,0.01,0.02,0.17,0.17,0.12,0.02,0.02,0.02,0.02,0.02,0.01),nrow=15,ncol=3)

daily\_demand=matrix(c(6.04,7.15,9.04,10.12,5.80,5.40,6.20,6.06,7.97,6.52,9.05,5.37,3.99,6.69,5.85,5.88,4.86,5.86,7.00,8.30),nrow=20,ncol=1)

working\_station=c()

cnt=0

prob=c()

for(i in 1:15)

{

cnt=station\_info[i,1]

prob=sample(0:1,size=cnt,prob=c(station\_info[i,3],1-station\_info[i,3]),replace=TRUE)

working\_station=c(working\_station,sum(prob==1))

}

sum=working\_station\*station\_info[,2]

station\_info=cbind(station\_info,sum,working\_station)

colnames(station\_info)=NULL

power\_on\_day=c()

sum1=0

for(i in 1:20)

{

for(j in 1:15)

{

sum1=sum1+(supply\_info[j,i]/100)\*station\_info[j,4]

}

power\_on\_day=c(power\_on\_day,sum1)

sum1=0

}

q1\_table=cbind(daily\_demand,power\_on\_day,(power\_on\_day-daily\_demand))

colnames(q1\_table)=c("Demand"," Total-power-supply"," Difference")

colnames(station\_info)=c(" Generators"," Power per Generator", " Non-operational prob", " Total power"," # Working-Stations")

print(station\_info)

return(q1\_table)

}

failure=function(){

q2\_table=daily\_power()

cat("\014")

print(q2\_table)

flag=0

for(i in 1:20)

{

if(q2\_table[i,3]<0)

flag=1

}

if(flag==0)

{

#print("NOT A BLACKOUT DAY!!")

return(0)

}

else

{

#print("BLACKOUT DAY!!")

return(1)

}

}

#------------------ CODE FOR CALCULATING SAMPLE MEAN (P(f))----------------------------

#sum=0

#ssq=0

#rounds=10^6

#delta=0.90

# for(i in 1:rounds)

# {

# x=sample()

# cat("\014")

# sum=sum+x

# ssq=ssq+x\*x

#}

#lambda=sum/rounds

#sigsquare=(ssq-(lambda\*lambda\*rounds))/(rounds-1)

#stderror=sqrt(sigsquare/rounds)

#re = stderror/lambda

#qdelta=qnorm((1+delta)/2)

#left=lambda-qdelta\*stderror

#right=lambda+qdelta\*stderror

#cat("\n Standard Error ::",stderror)

#cat("\n Variance ::",sigsquare)

# cat("\n 0.90-confidence interval :: [",left," , ",right,"]")

#cat("\n Sample Mean ::",lambda)

# cat("\n Mean time Between Failure ::",1/(1-lambda))

#====================RESULTS OF IMC SIMULATION==========================

#Standard Error :: 0.0004871739

#Variance :: 0.2373384

#0.90-confidence interval :: [ 0.6117237 , 0.6133263 ]

#Lambda :: 0.612525

failure2=function(s,n)

{

supply\_info=matrix(c(30,0,0,0,20,0,3,7,3,10,0,0,6,5,0,20,0,10,0,0,10,6,5,3,10,0,0,6,7,0,15,8,0,0,0,10,9,6,2,10,0,20,6,8,0,0,0,0,80,0,0,0,8,2,10,0,0,6,6,0,0,20,0,0,0,0,15,4,4,0,0,8,6,5,10,21,0,0,0,0,15,0,5,4,0,0,0,6,4,10,14,0,10,0,10,0,11,5,5,0,0,0,6,5,10,0,41,0,0,0,0,7,5,5,0,0,15,6,5,10,0,0,25,0,0,7,0,5,10,10,0,0,8,8,0,0,15,5,0,15,8,10,8,4,0,0,0,6,5,15,0,0,10,0,0,0,0,4,8,15,0,41,0,8,0,0,7,0,0,10,0,12,10,5,0,0,0,6,0,10,0,0,10,0,0,0,0,5,5,0,0,9,0,10,0,0,0,0,0,0,0,0,5,5,15,14,0,6,4,0,0,0,10,0,0,9,14,4,5,0,21,0,0,0,15,0,9,0,20,10,6,0,4,4,0,0,0,6,2,0,0,0,0,0,10,20,0,2,8,0,0,0,4,2,20,0,0,10,0,10,0,0,2,6,0,15,7,6,2,0,0,0,10,0,0,15,0,3,5,0,20,0,4,3,0,0,0,0,0,15,0,13,3,7,20,30,0,6,0,0),nrow=15,ncol=20)

station\_info=matrix(c(1,1,1,9,6,6,1,3,2,8,11,8,18,6,4,13.60,12.00,5.60,0.76,0.65,1.12,5.53,5.53,7.35,0.78,1.09,0.78,1.45,7.74,0.95,0.16,0.16,0.14,0.02,0.01,0.02,0.17,0.17,0.12,0.02,0.02,0.02,0.02,0.02,0.01),nrow=15,ncol=3)

daily\_demand=matrix(c(6.04,7.15,9.04,10.12,5.80,5.40,6.20,6.06,7.97,6.52,9.05,5.37,3.99,6.69,5.85,5.88,4.86,5.86,7.00,8.30),nrow=20,ncol=1)

flag=0

station\_info[s,3]=n

working\_station=c()

cnt=0

prob=c()

for(i in 1:15)

{

cnt=station\_info[i,1]

prob=sample(0:1,size=cnt,prob=c(station\_info[i,3],1-station\_info[i,3]),replace=TRUE)

working\_station=c(working\_station,sum(prob==1))

}

sum=working\_station\*station\_info[,2]

station\_info=cbind(station\_info,sum,working\_station)

colnames(station\_info)=NULL

power\_on\_day=c()

sum1=0

for(i in 1:20)

{

for(j in 1:15)

{

sum1=sum1+(supply\_info[j,i]/100)\*station\_info[j,4]

}

power\_on\_day=c(power\_on\_day,sum1)

sum1=0

}

q1\_table=cbind(daily\_demand,power\_on\_day,(power\_on\_day-daily\_demand))

colnames(q1\_table)=c("Demand"," Total-power-supply"," Difference")

colnames(station\_info)=c(" Generators"," Power per Generator", " Non-operational prob", " Total power"," Working")

print(q1\_table)

for(i in 1:20)

{

if(q1\_table[i,3]<0)

flag=1

}

if(flag==0)

{

#print("NOT A BLACKOUT DAY!!")

return(0)

}

else

{

#print("BLACKOUT DAY!!")

return(1)

}

}

consortium\_report=function()

{

temp=0

sum=0

ssq=0

delta=0.90

lambda1=c()

sigsquare1=c()

stderror1=c()

re1=c()

qdelta1=c()

left1=c()

right1=c()

#final\_op=c()

station\_no=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)

#station\_info=matrix(c(1,1,1,9,6,6,1,3,2,8,11,8,18,6,4,13.60,12.00,5.60,0.76,0.65,1.12,5.53,5.53,7.35,0.78,1.09,0.78,1.45,7.74,0.95,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),nrow=15,ncol=3)

station\_info=matrix(c(1,1,1,9,6,6,1,3,2,8,11,8,18,6,4,13.60,12.00,5.60,0.76,0.65,1.12,5.53,5.53,7.35,0.78,1.09,0.78,1.45,7.74,0.95,0.16,0.16,0.14,0.02,0.01,0.02,0.17,0.17,0.12,0.02,0.02,0.02,0.02,0.02,0.01),nrow=15,ncol=3)

rounds=10000

for(i in 1:15)

{

temp=station\_info[i,3]

sum=0

ssq=0

for(j in 1:rounds)

{

x=failure2(i,temp/2)

cat("\014")

sum=sum+x

ssq=ssq+x\*x

}

lambda=sum/rounds

sigsquare=(ssq-(lambda\*lambda\*rounds))/(rounds-1)

stderror=sqrt(sigsquare/rounds)

re = stderror/lambda

qdelta=qnorm((1+delta)/2)

left=lambda-qdelta\*stderror

right=lambda+qdelta\*stderror

lambda1=c(lambda1,lambda)

sigsquare1=c(sigsquare1,sigsquare)

stderror1=c(stderror1,stderror)

re1=c(re1,re)

qdelta1=c(qdelta1,qdelta)

left1=c(left1,left)

right1=c(right1,right)

}

pf=(1/lambda1)-1

final\_op=cbind(station\_no,lambda1,left1,right1,pf)

colnames(final\_op)=c("#Station"," Pf"," Left"," Right"," MTBF")

return(final\_op)

#pf=(1/lambda1)-1

#print(lambda1)

#print(pf)

#return(lambda1)

}

#QUESTION 5 -------

#Looking at the Mean time between failures and its definition it is clear that, the greater the MTBF the

#better it is for the system as it is indicative of the time gap between two failures. Yes, I would recommend

#them to use this criteria. I would suggest to upgrade station 8 as its supply matrix indicates that it

#provides some amount of energy to each and every city, and hence an upgrade on it will mean lesser blackouts