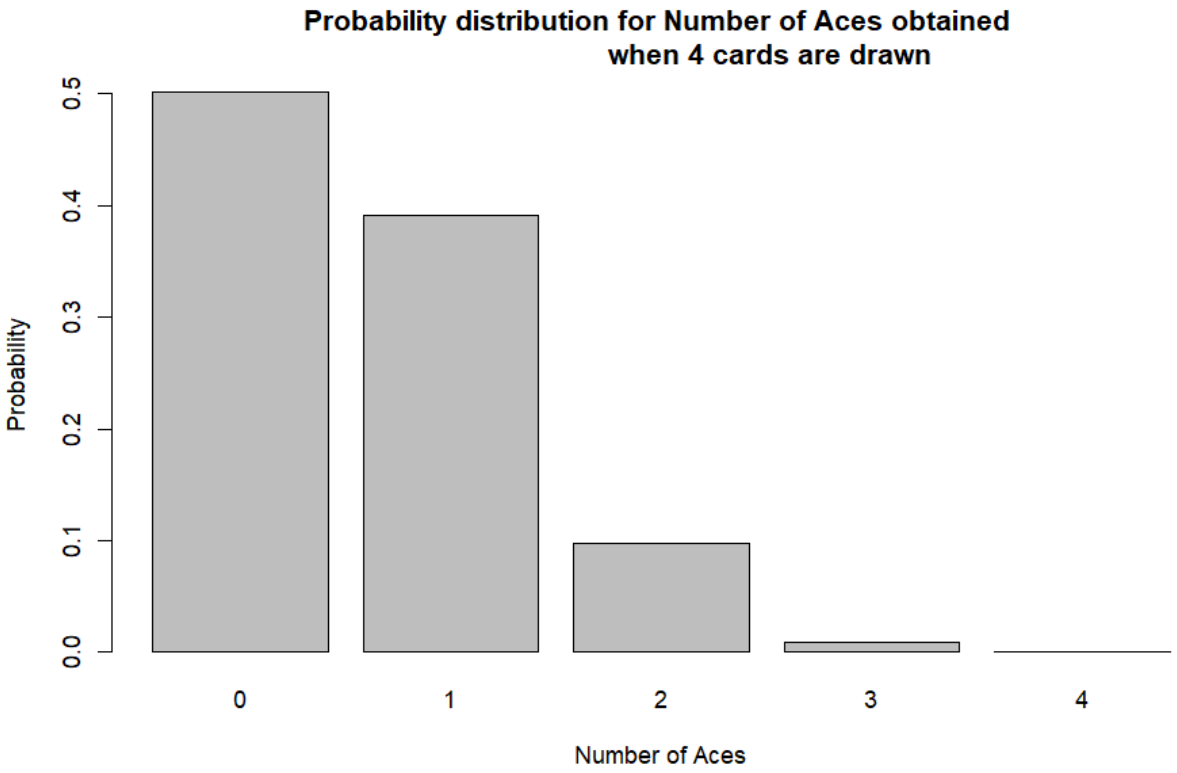
1.

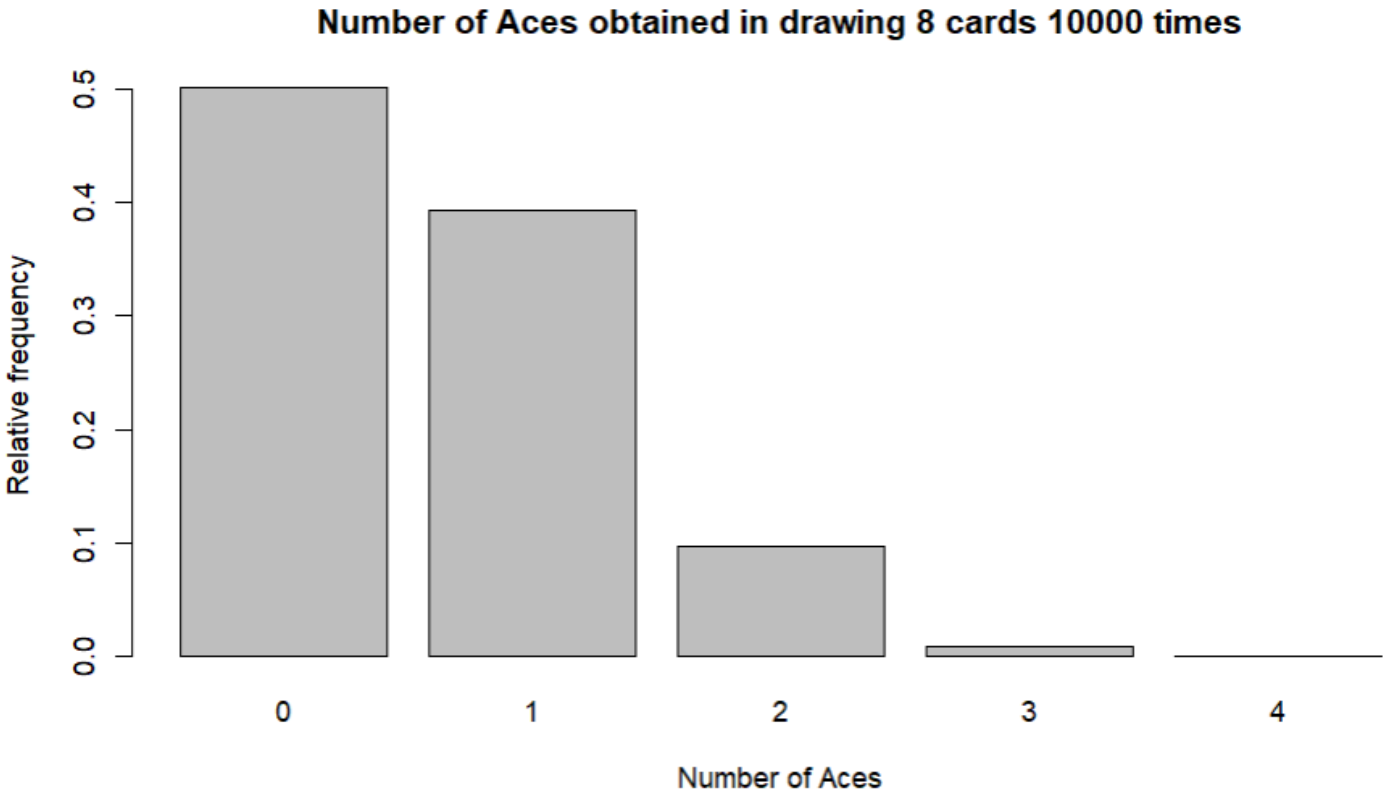



```
> NumOfAcesProbs(4,8)
   numOfProbs numOfAces
1 0.501435036         0
2 0.391363930         1
3 0.097840983         2
4 0.009101487         3
5 0.000258565         4
```

```
NumOfAcesProbs=function(n,m){
  numOfProbs<-c(0:n)
  for (i in 0:n){
    numOfProbs[i + 1]=dhyper(i,4,48,m)
  }
  numOfAces=c(0:n)
  tableAcesProbs=data.frame(numOfProbs,numOfAces)
  titleProbsOfAces = paste("Probability distribution for Number of Aces obtained
                           when",n, "cards are drawn")
  barplot(numOfProbs,main=titleProbsOfAces,xlab="Number of Aces",names=numOfAces,ylab="Probability")
  return(tableAcesProbs)
}
```

2.

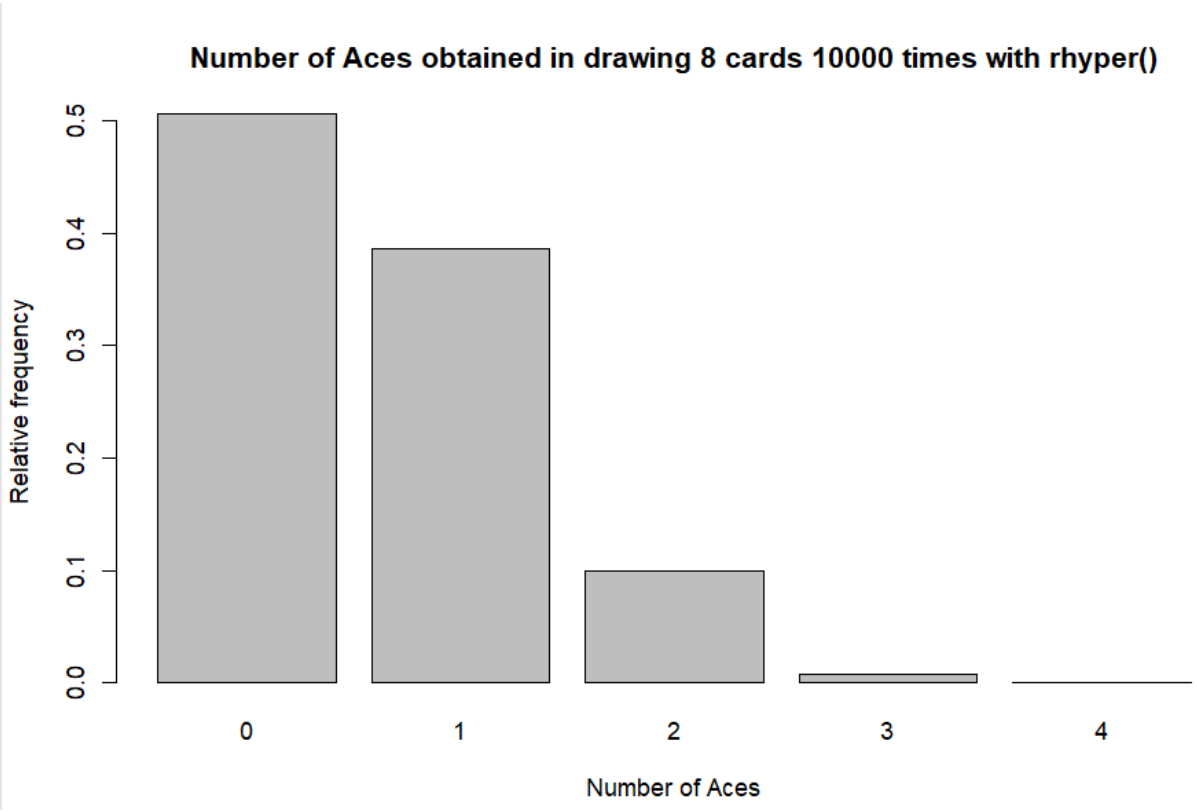
a)




```
> DrawAces(10000,8)
numOfAces
     0      1      2      3      4
0.5015 0.3927 0.0969 0.0086 0.0003
> 
```

```r
# draw m cards n times using sample()
DrawAces=function(n,m){
  deckOfCards<-c(1:52)
  for (i in 1:4){deckOfCards[i]="Ace"}
  numOfAces<-c(1:n)
  for (i in 1:n){
    numOfAces[i]<-sum(sample(deckOfCards,m,repl=F)=="Ace")
  }
  # table of relative frequency
  numOfAcesTable<-table(numOfAces)/n
  titleNumOfAces = paste("Number of Aces obtained in drawing",m, "cards",n,"times")
  barplot(numOfAcesTable, main=titleNumOfAces, xlab="Number of Aces", ylab="Relative frequency")
  return (numOfAcesTable)
}
```

b)




```
> DrawAcesWithRhyper(10000,8)
numOfAces
     0      1      2      3      4
0.5061 0.3860 0.0997 0.0080 0.0002
```

```
# draw m cards n times using rhyper()
DrawAcesWithRhyper=function(n,m){
  numOfAces<-c(1:n)
  for (i in 1:n){
    numOfAces[i]<-rhyper(n,4,48,m)
  }
  # table of relative frequency
  numOfAcesTable<-table(numOfAces)/n
  titleNumOfAces = paste("Number of Aces obtained in drawing",m, "cards",n,"times with rhyper()")
  barplot(numOfAcesTable, main=titleNumOfAces, xlab="Number of Aces", ylab="Relative frequency")
  return (numOfAcesTable)
}
```

3.

a)

```
> dhyper(100,333,666,300)
[1] 0.05834763
>
```

b)

```
> phyper(100,333,666,300)
[1] 0.5304624
>
```

c)

```
> 1-phyper(200,666,333,300)
[1] 0.4721148
```

d)

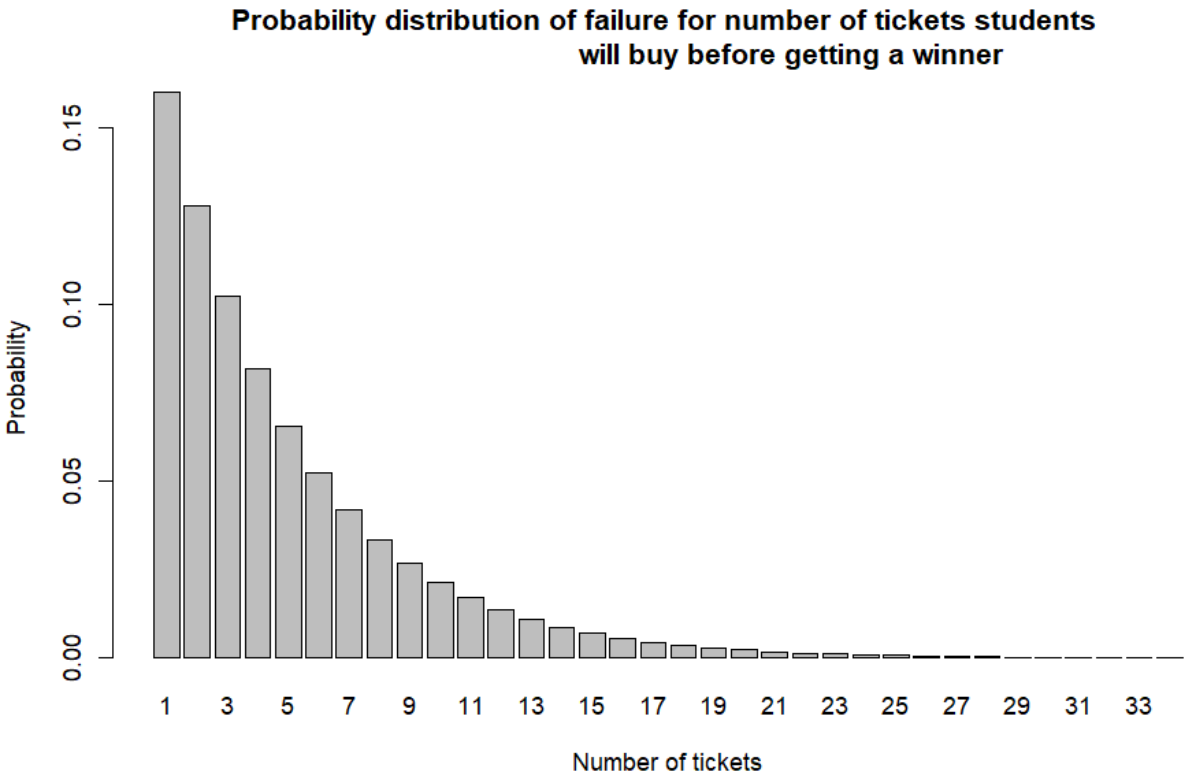
```
> phyper(190,666,333,300)
[1] 0.08253631
>
```

e)

```
> (1-phyper(190,666,333,300))-(1-phyper(210,666,333,300))+dhyper(190,666,333,300)
[1] 0.8759101
>
```

4.

a)

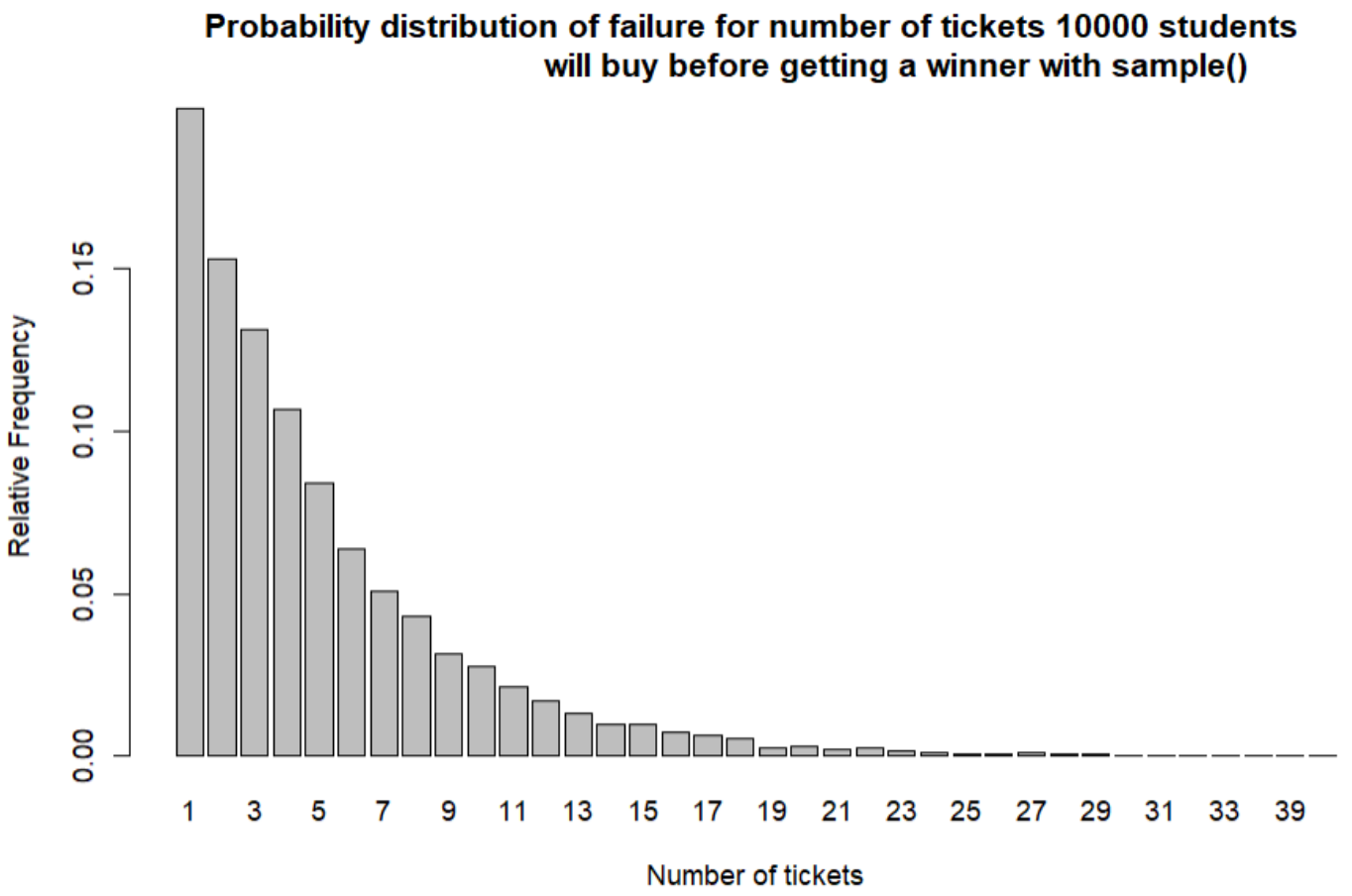


```
> NumTicketsProbs(100,0.2)
   numOfTicketsProbs numOfTickets
1       0.1600000000             1
2       0.1280000000             2
3       0.1024000000             3
4       0.0819200000             4
5       0.0655360000             5
6       0.0524288000             6
7       0.0419430400             7
8       0.0335544320             8
9       0.0268435456             9
10      0.0214748365            10
11      0.0171798692            11
12      0.0137438953            12
13      0.0109951163            13
14      0.0087960930            14
15      0.0070368744            15
16      0.0056294995            16
17      0.0045035996            17
18      0.0036028797            18
19      0.0028823038            19
20      0.0023058430            20
21      0.0018446744            21
22      0.0014757395            22
23      0.0011805916            23
24      0.0009444733            24
25      0.0007555786            25
26      0.0006044629            26
27      0.0004835703            27
28      0.0003868563            28
29      0.0003094850            29
30      0.0002475880            30
31      0.0001980704            31
32      0.0001584563            32
33      0.0001267651            33
34      0.0001014120            34
>
```

```r
# probability distribution of failure for the number of tickets
# the student will buy before getting a winner
NumTicketsProbs=function(n,m){
  numOfAllTicketsProbs<-c(1:n)
  counter = 0;
  for (i in 1:n){
    numOfAllTicketsProbs[i]<-dgeom(i,m)
    if(numOfAllTicketsProbs[i]>0.0001){
      counter = counter + 1
    }
  }
  numOfTicketsProbs<-c(1:counter)
  j = 0
  for(i in 1:n){
    if(numOfAllTicketsProbs[i]>0.0001){
      j = j + 1
      numOfTicketsProbs[j]=numOfAllTicketsProbs[i]
    }
  }
  numOfTickets=c(1:counter)
  tableWinProbs=data.frame(numOfTicketsProbs,numOfTickets)
  titleProbsOfTickets = paste("Probability distribution of failure for number of tickets students
                              will buy before getting a winner")
  barplot(numOfTicketsProbs,main=titleProbsOfTickets,xlab="Number of tickets",names=numOfTickets,ylab="Probability")
  return(tableWinProbs)
}
```

5.

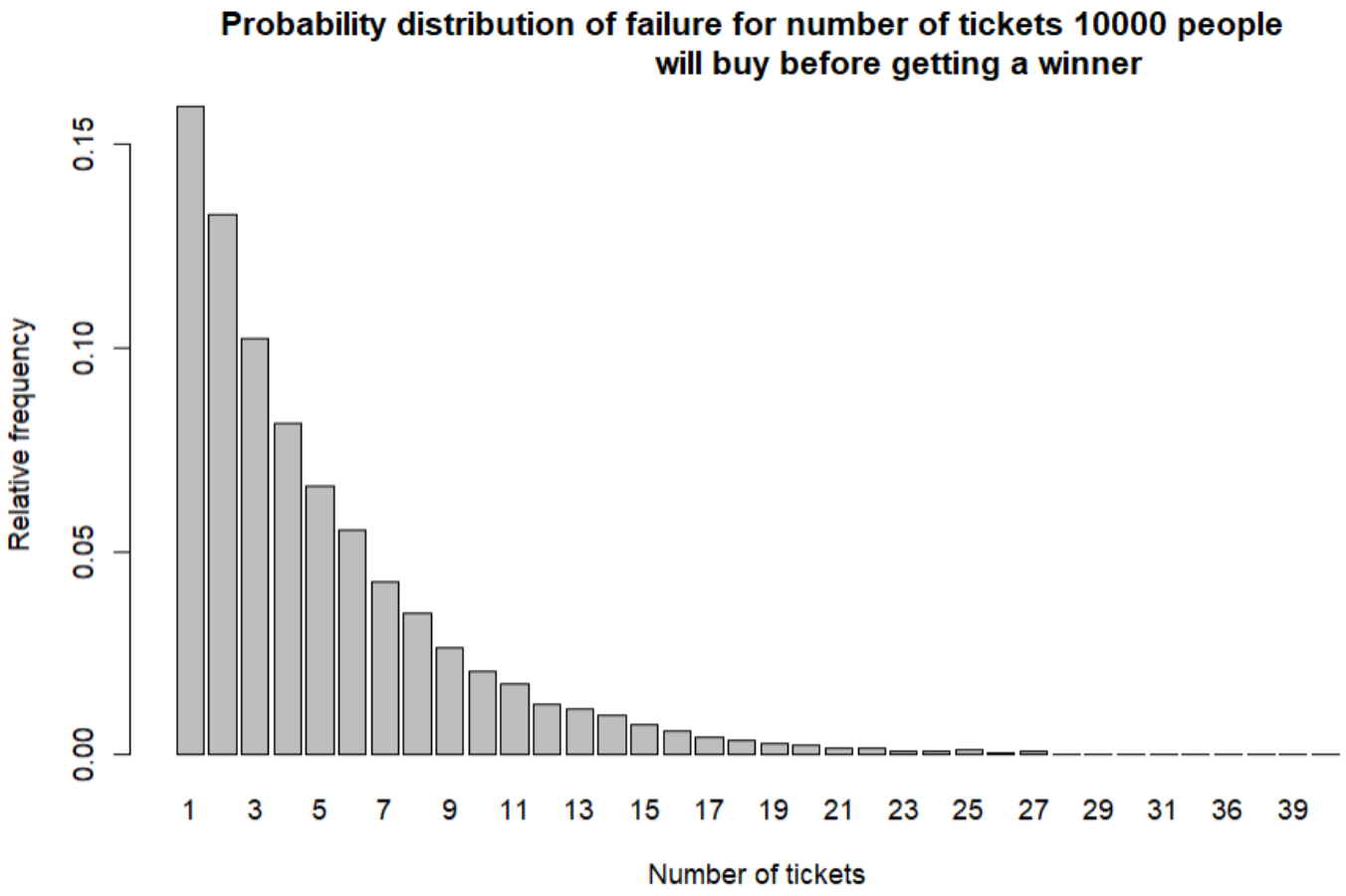
a)




```
> WinTicketProbWithSample(10000,100)
numOfTicketsToWin
     1      2      3      4      5      6      7      8      9     10     11     12     13     14
0.1992 0.1529 0.1313 0.1066 0.0842 0.0640 0.0510 0.0431 0.0315 0.0276 0.0212 0.0171 0.0133 0.0098
    15     16     17     18     19     20     21     22     23     24     25     26     27     28
0.0099 0.0072 0.0063 0.0057 0.0027 0.0030 0.0022 0.0027 0.0015 0.0012 0.0006 0.0007 0.0010 0.0007
    29     30     31     32     33     34     39     40
0.0007 0.0001 0.0002 0.0001 0.0001 0.0003 0.0001 0.0002
>
```

```r
# With sample(): probability distribution for the number of lottery tickets the student
# must buy before obtaining a winner
# n : number of students
# m : number of times buying lottery tickets
WinTicketProbWithSample=function(n,m){

  numOfTicketsToWin<-c(1:n)
  for (i in 1:n){
    buyTicketsRec<-sample(c("W","L"),m,repl=T,prob = c(0.2,0.8))
    counter = 0
    for (j in 1:m){
      if(buyTicketsRec[j]=="W"){
        counter = counter + 1
        break
      } else{
        counter = counter + 1
      }
    }
    numOfTicketsToWin[i]=counter
  }

  WinProbsTable=table(numOfTicketsToWin)/n

  titleProbsOfTickets = paste("Probability distribution of failure for number of tickets",n,"students
                              will buy before getting a winner with sample()")
  barplot(WinProbsTable,main=titleProbsOfTickets,xlab="Number of tickets",ylab="Relative Frequency")
  return(WinProbsTable)

}
```

b)



```r
# With rgeom(): probability distribution for the number of lottery tickets the student
# must buy before obtaining a winner
# n : number of people buy lottery tickets
# m : probability of winning
WinTicketProbWithRgeom=function(n,m){
  numOfTickets = rgeom(n,m)
  numOfTickets[numOfTickets==0]=NA
  winTable = table(numOfTickets)/n

  titleWinProbs = paste("Probability distribution of failure for number of tickets",n,"people
                        will buy before getting a winner")
  barplot(winTable,main=titleWinProbs,xlab="Number of tickets",ylab="Relative frequency")
  return(winTable)
}
```

```
> WinTicketProbWithRgeom(10000,1/5)
numOfTickets
     1      2      3      4      5      6      7      8      9     10     11     12     13     14
0.1592 0.1327 0.1022 0.0813 0.0660 0.0554 0.0426 0.0347 0.0263 0.0206 0.0176 0.0125 0.0112 0.0098
    15     16     17     18     19     20     21     22     23     24     25     26     27     28
0.0074 0.0059 0.0042 0.0036 0.0030 0.0026 0.0016 0.0015 0.0009 0.0008 0.0011 0.0005 0.0010 0.0002
    29     30     31     32     36     38     39     48
0.0002 0.0002 0.0002 0.0001 0.0001 0.0001 0.0001 0.0001
>
```

6.

a)

```
> pgeom(500,0.001)
[1] 0.3942274
>
```

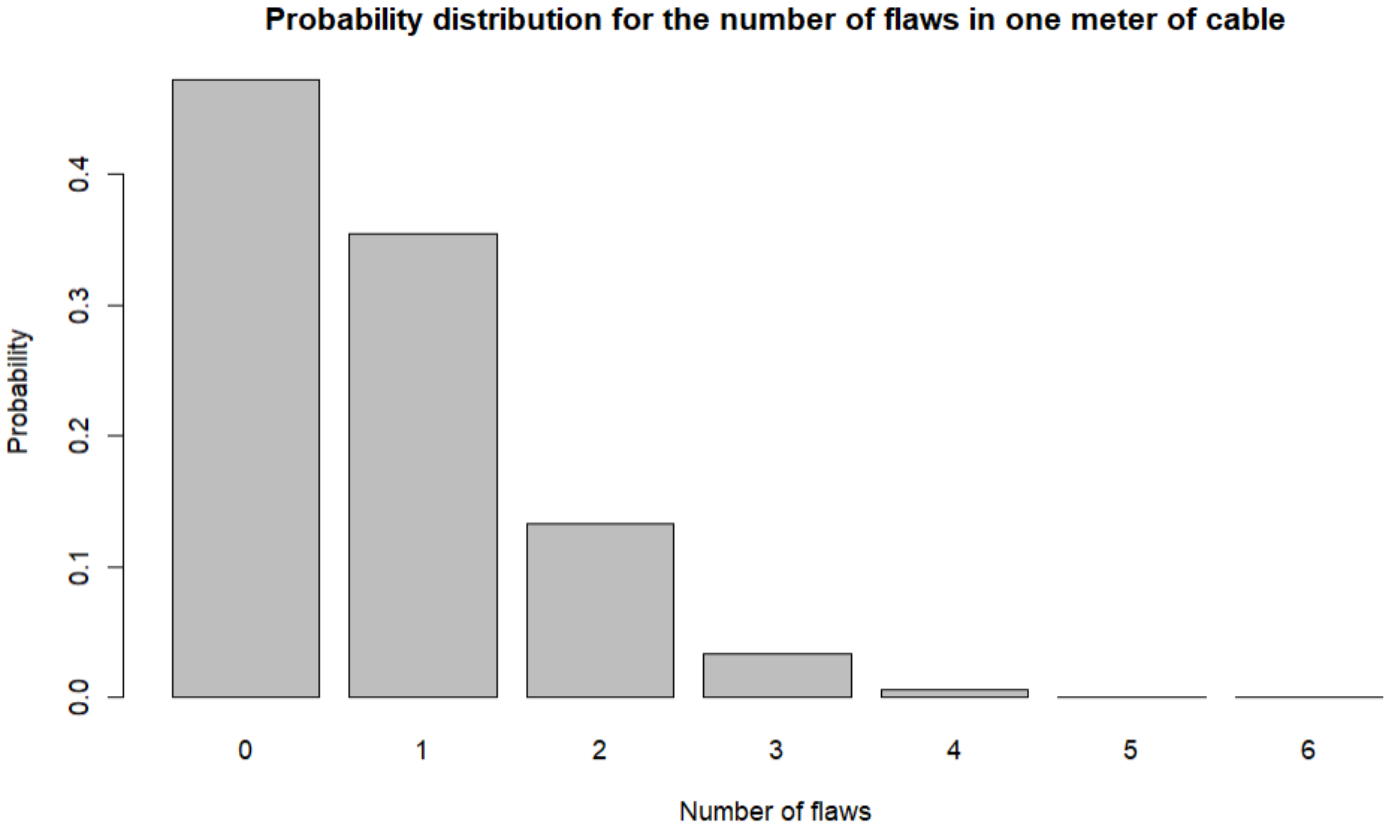
b)

```
> 1-pgeom(1199,0.001)
[1] 0.30101
```

c)

```
> (1-pgeom(999,0.001))-(1-pgeom(1999,0.001))
[1] 0.232495
```

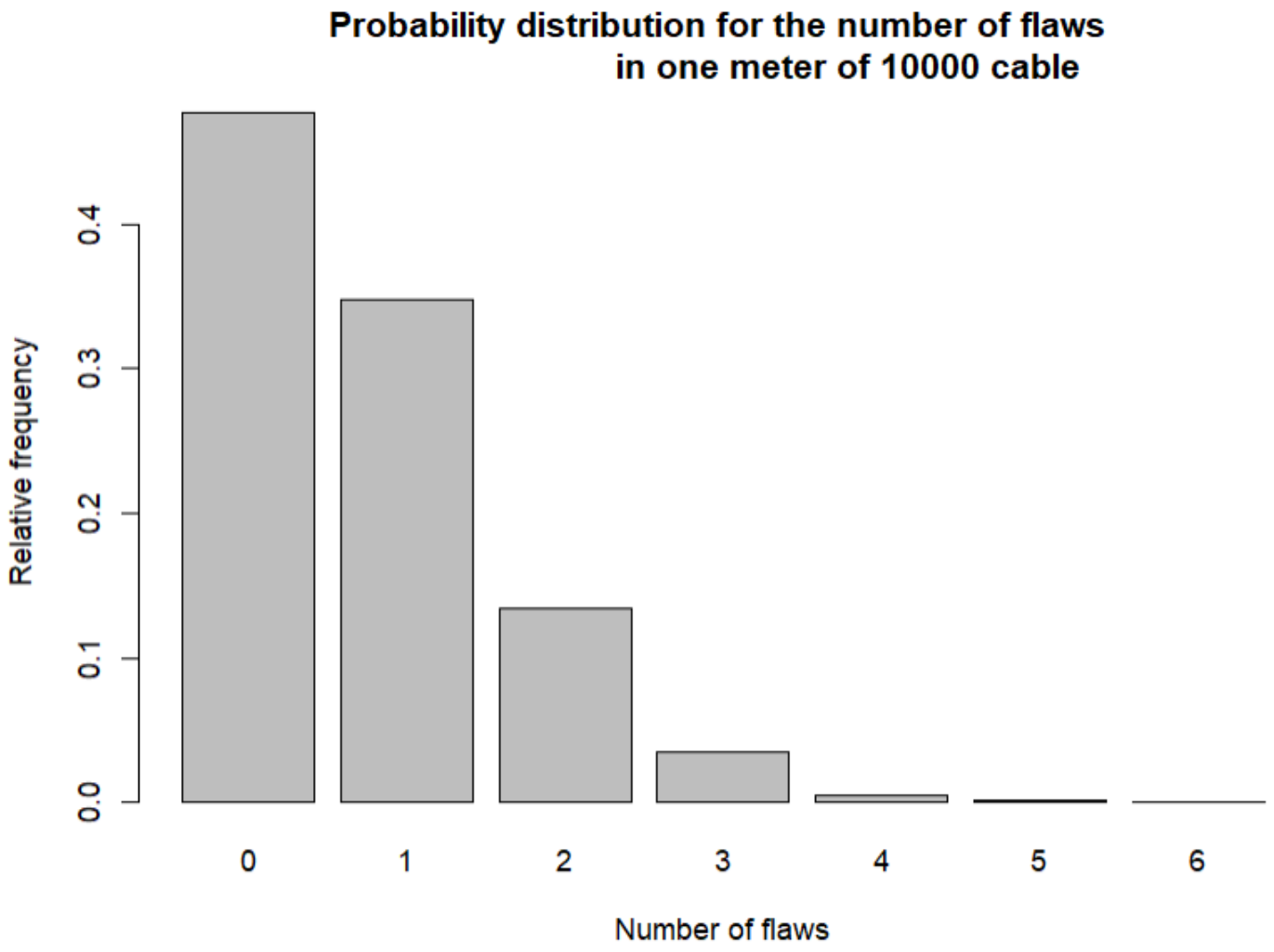
7.




```
> NumOfFlaws(100,0.75)
  flawProbs numFlaws
1 0.4723666        0
2 0.3542749        1
3 0.1328531        2
4 0.0332133        3
5 0.0062275        4
6 0.0009341        5
7 0.0001168        6
>
```

```r
# probability distribution for the number of flaws in one metre of cable
# n : maximum number of flaws
# m : lambda
NumOfFlaws=function(n,m){
  flawAllProbs<-c(0:n)
  counter = 0
  for (i in 0:n){
    flawAllProbs[i+1]<-dpois(i,m)
    if(flawAllProbs[i+1]>0.0001){
      counter = counter + 1
    }
  }
  counter = counter - 1
  flawProbs<-c(0:counter)
  j = 0
  for (i in 0:n){
    if(flawAllProbs[i+1]>0.0001){
      j = j + 1
      flawProbs[j] = flawAllProbs[i+1]
    }
  }
  numFlaws=c(0:counter)
  tableFlawProbs=data.frame(flawProbs,numFlaws)
  titleFlawProbs = paste("Probability distribution for the number of flaws in one meter of cable")
  barplot(flawProbs,main=titleFlawProbs,xlab="Number of flaws",names=numFlaws,ylab="Probability")
  return(tableFlawProbs)
}
```

8.



```
> NumOfFlawsRpois(10000,0.75)

     0      1      2      3      4      5      6
0.4769 0.3474 0.1340 0.0348 0.0055 0.0013 0.0001
> |
```

```
# With rpois(): probability distribution for the number of flaws in one metre of cable
# n : number of cables
# m : lambda
NumOfFlawsRpois=function(n,m){
  numFlaws<-table((rpois(n,m)))/n
  titleFlawProbs = paste("Probability distribution for the number of flaws
                         in one meter of",n,"cable")
  barplot(numFlaws,main=titleFlawProbs,xlab="Number of flaws",ylab="Relative frequency")
  return(numFlaws)
}
```

9.

a)

```
> ppois(34,34)-ppois(33,34)
[1] 0.06825
> |
```

b)

```
> ppois(30,34)
[1] 0.2804
> |
```

c)

```
> ppois(29,34)
[1] 0.2235
> |
```

d)

```
> 1-ppois(38,34)
[1] 0.2166
> |
```

e)

```
> 1-ppois(37,34)
[1] 0.2681
> |
```

f)

```
> (1-ppois(29,34))-(1-ppois(40,34))
[1] 0.6429
> |
```

g)

```
> ppois(34,34)*ppois(34,34)
[1] 0.2975
```

h)

```
> ppois(68,68)
[1] 0.5322
> 
```