**EXERCISES 4.1** [1]

1. Four bits are transmitted over a digital communication channel. Each bit is either distorted or received without distortion. List the sample space S and the event of interest E so that at most one bit will be distorted.

   Let *d* denote a distorted bit, and *g* and non-distorted bit.

   ```
   Sample Space
   S:  {dddd,
         dddg, ddgd, dgdd, gddd,
          ggdd,  gdgd, gddg, ddgg, dgdg, gddg
          gggd, gdgg, dggg, ggdg,
         gggg}

   E: at most one bit distorted:

   E:  {gggd, gdgg, dggg, ggdg,  gggg}
   ```

2. An order for a computer system can specify memory of 2, 4, or 6 GB and disk storage of 250 or 500 GB. Describe the set of all designs.

   ```
   All Possible Designs {(2,250),(2,500),(4,250),(4,500), (6,250),(6,500)}
   ```

3. Computer chips coming off an assembly line are tested for quality and are rated defective (d) or good (g). A quality control inspection carried out every hour tests the chips until two consecutive chips are defective or until four chips have been tested, whichever occurs first. List the sample space S for this experiment and the event E so that four chips will be inspected.

   ```
   Sample Space
   S:  {dd, gdd, gdgd, ggdd, ggdg, gggd, gggg}

   E:  {gdgd, ggdd, ggdg, gggd, gggg}
   ```

4. A computer student can repeat an examination until it is passed, but is allowed to attempt the examination at most four times. List the sample space S and the event E that the student fails.

   ```
   Sample Space
   S:  {p, fp, ffp, fffp, ffff}

   E:  {ffff}
   ```

[1] Probability with *R*: An Introduction with Computer Science Applications: Jane M. Horgan, Wiley 2008

EXERCISES 4.2 Use R to solve the following:

1. If passwords can consist of six letters, find the probability that a randomly chosen password will not have any repeated letters.

   Nunber of Possible Passwords: $26^6$

   Favourable Cases: ${}^{26}C_6$

   P(No repeated letters) $= \frac{{}^{26}C_6}{26^6}$

   We could calculate this probability in $R$ using

```
prod(26:21)/26^6
 0.5366045
```

2. A sample of size 10 is chosen at random from a class of 100 consisting of 60 females and 40 males. Obtain the probability of getting 10 females.

   Class:

| 60 | 40 |
|---|---|
| Females | Males |

   P(10 females without replacement) $= \frac{{}^{60}P_{10}}{{}^{100}P_{10}}$

   In $R$

```
> prod(60:51)/prod(100:91)
[1] 0.004355441
```

3. A box with 15 IC chips contains 5 defective chips. If a sample of three chips is drawn at random without replacement, what is the probability that all the three are defective?

   Box:

| 5 | 10 |
|---|---|
| Defectives | Good |

   P(all three defective) $= \frac{{}^{5}P_{3}}{{}^{15}P_{3}}$

   In $R$

```
> prod(5:3)/prod(15:13)
[1] 0.02197802
```

4. A batch of 50 semiconductors contains 10 that are defective.

   Batch:

| 10 | 40 |
|---|---|
| Defectives | Good |

   Two are selected at random, without replacement.

   (a) What is the probability that the first one selected is defective? 10/50

   (b) What is the probability that the second one selected is defective?

$$10/50 \times 9/49 + 40/50 \times 10/49$$

   In $R$

```
> (10/50)*9/49  + (40/50)*10/59
[1] 0.1723279
```

(c) What is the probability that both are defective? $\frac{10*9}{50*49}$

In $R$

```
> (10*9)/(50*49)
[1] 0.03673469
```

(d) How would the probability in (b) change if the chips selected were replaced before the next selection? 10/50

5. In a party of five students, compute the probability that at least two have the same birthday (month/day), assuming a 365-day year.

$$P(\text{All different birthdays}) = \frac{{}^{365}P_5}{365^5},$$

$$P(\text{At least 2 the same}) = 1 - \frac{{}^{365}P_5}{365^5},$$

In $R$

```
> k<-5
>  prod(365:(365-k+1))/365^k  #all different
[1] 0.9728644
>  1-prod(365:(365-k+1))/365^k  #at least 2 the same
[1] 0.02713557
```

6. The probability that two students in a class have the same birthday is at least 75%. What is the minimum size of the class?

$$P(\text{At least 2 the same in k}) = 1 - \frac{{}^{365}P_k}{365^k}$$

Choose $k$ so that

$$1 - \frac{{}^{365}P_k}{365^k} \geq 0.75$$

From the table on p55 we can see that $k$ is somewhere between 30 and 40.

From the diagram on p56, we see that $k$ is just above 30.

In $R$

```
> k<-30
>   1-prod(365:(365-k+1))/365^k  #at least 2 the same
[1] 0.7063162
> k<-31
>   1-prod(365:(365-k+1))/365^k  #at least 2 the same
[1] 0.7304546
> k<-32
>   1-prod(365:(365-k+1))/365^k  #at least 2 the same
[1] 0.7533475
```

The minimum $k$ is 32. i.e. There needs to be a class size of 32 or more to be 75% sure so two students will have the same birthday.

7. A series of 20 jobs arrive at a computing center with 50 processors. Assume that each of the jobs is equally likely to go through any of the processors.

   (a) Find the probability that a processor is used at least twice.

$$P(\text{ All Different}) = \frac{{}^{50}P_{20}}{50^{20}}$$

$$P(\text{At least twice}) = 1 - \frac{{}^{50}P_{20}}{50^{20}}$$

   In $R$

```
>   prod(20:(20-k+1))/20^k  #All different processors
[1] 2.320196e-08
```

   Probability at least one used twice $= 1 - 2.320196e - 08 \approx 1$. Almost certain. Do a few simulations in $R$ to see for yourself.

```
sample(seq(1:50), 20, replace = T)  #samples 20 from 50 with replacement
 [1] 29 13 24 28 14  9  3 45  5  4 17 42 34 40 20 24 13 37 34 40

sample(seq(1:50), 20, replace = T)  #samples 20 from 50 with replacement
 [1] 16 29 31 16 28 10  8 24 14 35 40 23 18  3 46 28 13 40 23 47
```

   (b) What is the probability that at least one processor is idle? This is equivalent to the probability that at least one is used twice.

8. Simulate the allocation problem in the previous exercise a large number of times, and estimate how many times the most used processor was used.

```
freq <-  0
 for(i in 1:10000)
  {
   x <-sample(seq(1:50), 20, replace = T)  #samples 20 from 50 with replacement
   freq[i] <- max(table(x)) # obtains the max no of uses of proc in a sample.
   }

table(freq) #tabluates the multiple uses of processors
freq
   1    2    3    4    5    6
 107 6761 2829  289   12    2
```

This means that 107 out of 10000 samples have all distinct processors, 6761 had one repeat etc. The most used processor was used 6 times in sample; and this occurred in 2 of the 10,000 simulations. What this means that the processors should have a capacity of 6; it is very unlikely that any processor will be required to process more than 6 jobs.

We could also express this as probability estimates.

```
    1      2      3      4      5      6
0.0107 0.6761 0.2829 0.0289 0.0012 0.0002
```

How many of the processors were not used at all? Eventually all the processors will be used

9. Recall that in Example 4.19 we calculated that 431 or more processors are needed to be at least 90% sure that no processor will receive more than one of the 10 jobs to be allocated. You will agree that this appears excessively large. To assure that the answer is correct, simulate the experiment a large number of times and record the usage patterns of the processors.

```
 s = 0
 for(i in 1:10000)
 {
 x <-sample(seq(1:431), 10, replace = T)  #samples 10 from 431 with replacement
 y <- unique (x) #unique numbers in x
 diff <- 10 - length(y) # number of repeats in x
 if (diff) s = s else s = s+1  #if diff = 0 (no repeats) s = s+1, if diff  is not equal
}
s   # number of samples with no repeats
[1] 9008

> s/10000  # proportion of samples with no repeats
[1] 0.9008
```

So, about 90% of the samples contained no repeats. i.e. 90% sure that no processor will receive more than one of the 10 jobs allocated when there are 431 processors.

Suppose that we increase the number of processors to 1000, and again do 10000 replications:

```
> s = 0
>  for(i in 1:10000)
+  {x <-sample(seq(1:1000), 10, replace = T)
+  y <- unique (x) #unique numbers in x
+  diff <- 10 - length(y) # number of repeats in x
+  if (diff) s = s else s = s+1
+ }
> diff
[1] 0
> s
[1] 9552
>  s/10000
[1] 0.9552
```

Just under 96% of the samples contained no repeats. i.e. less that 96% sure that no processor will receive more than one of the 10 jobs allocated when there are 1000 processors. Equivalently, over 4% chance that with 10 jobs and 1000 processors, at least one processor will receive more than one job.

**EXERCISES 4.3**

1. Use R to illustrate that the probability of getting

   (a) a head is 0.5 if a fair coin is tossed repeatedly;

```
> x<-sample(c("H","T"), 1000, replace=T)
> table(x)
x
  H   T
```

```
476 524
> table(x)/1000
x
    H     T
0.476 0.524
> x<-sample(c("H","T"), 10000, replace=T)
> table(x)/10000
x
     H      T
0.5081 0.4919
```

(b) a red card is 0.5 if cards are drawn repeatedly with replacement from a well-shuffled deck;

```
> x<-sample(c("R","B"), 10000, replace=T)
> table(x)/10000
x
     B      R
0.4991 0.5009
```

(c) an even number is 0.5 if a fair die is rolled repeatedly.

```
> x<-sample(seq(1:6), 10000, replace=T)
> table(x)/10000
x
     1      2      3      4      5      6
0.1685 0.1629 0.1646 0.1715 0.1687 0.1638
```

2. An experiment consists of tossing two fair coins. Use R to simulate this experiment 100 times and obtain the relative frequency of each possible outcome.

```
> for (i in 1:100)
+ {
+ x[i]<-sample(c("HH","HT", "TH", "TT"), 1,  replace=TRUE)
+ }
> table(x)
x
HH HT TH TT
28 28 26 18
> table(x)/100
x
  HH   HT   TH   TT
0.28 0.28 0.26 0.18
```

or

```
> x<-sample(c("HH","HT", "TH", "TT"), 100,  replace=TRUE)
> table(x)/100
x
  HH   HT   TH   TT
0.35 0.30 0.16 0.19
```

Here number of simulations too small. Try 1000

```
 x<-sample(c("HH","HT", "TH", "TT"), 1000,  replace=TRUE)
> table(x)/1000
```

```
x
   HH    HT    TH    TT
0.252 0.244 0.246 0.258
```

```
> round(table(x)/1000, 2)
x
  HH   HT   TH   TT
0.25 0.24 0.25 0.26
```

Try 10000

```
> x<-sample(c("HH","HT", "TH", "TT"), 10000,  replace=TRUE)
> round(table(x)/10000, 2)
x
  HH   HT   TH   TT
0.25 0.24 0.26 0.25
```

Finally 20000

```
> x<-sample(c("HH","HT", "TH", "TT"), 20000,  replace=TRUE)
> round(table(x)/20000, 2)
x
  HH   HT   TH   TT
0.25 0.25 0.25 0.25
```

Hence, estimate the probability of getting one head and one tail in any order.

$= .25+.25 = .5$

3. An experiment consists of tossing a die. Use R to simulate this experiment 600 times and obtain the relative frequency of each possible outcome. Hence, estimate the probability of getting each of 1, 2, 3, 4, 5, and 6.

```
>  x<- sample (seq(1:6), 600, replace = T)
> relfreq <- table(x)/600
> relfreq
x
        1         2         3         4         5         6
0.1700000 0.1733333 0.1783333 0.1583333 0.1600000 0.1600000
> round(relfreq, 2) #rounds to 2 decimal places
x
   1    2    3    4    5    6
0.17 0.17 0.18 0.16 0.16 0.16
```

Best to increase number of simulations to get better estimate of probabilities.

4. Amy and Jane are gambling against one another.A fair coin is tossed repeatedly. Each time a head comes up, Amy wins two euros from Jane, and each time a tail comes up, Amy loses 2 euros to Jane. Use R to simulate this game 100 times,

```
 x<- sample (c(2, -2), 100, replace = TRUE)
```

and estimate

(a) the number of times that Amy is ahead in these 100 tosses;

```
> add<-0
> add[1] <- x[1]
> for (i in 2:100) add[i] = add[i-1] + x[i]
> plot(x)
```

(b) how much Amy has won or lost.

```
table(x)
x
-2  2
49 51
```

This means that Amy lost 49 tosses, and won 51. Also

```
table(add)
add
-10  -8  -6  -4  -2   0   2   4   6   8  10  12
  3   6   5   6   6   6  14  20  18  11   4   1
```

means that Amy was down 10 euro 3 times, down 8 six times and so on to up 12 once.

```
sum(x)
[1] 4
```

or equivalently

```
add[50]
[1] 4
```

Amy has won 4 euro

5. While plotting, we used type = o in the plot command, which joined the points. Alternative characterizations of a plot may be obtained by using different options. Explore the possibilities of different types of plots by varying the Type symbols Try

```
num <- 1:100
plot(num, add, type = "o", xlab = "Toss number", ylab = "Winnings")
plot(num, add, type = "l", xlab = "Toss number", ylab = "Winnings")
plot(num, add, type = "s", xlab = "Toss number", ylab = "Winnings")
plot(num, add, type = "S", xlab = "Toss number", ylab = "Winnings")
```