Introduction to Probability and Information Theory
Homework #3

G&S pg. 197

#7 (a) If we count the first roll as '0', then T is distributed according the the geometric distribution with $p = \frac{1}{6}$ (i.e., $T \sim \text{Geom}(\frac{1}{6})$)

(b) Note that $P(T > 3) = 1 - P(T \leq 2)$, and we can get $1 - P(T \leq 2)$ using R thusly:

```
> 1-pgeom(2,1/6)
[1] 0.5787037
```

(c) We know:

$$\begin{aligned} P(T > 6 | T > 3) &= \frac{P(T > 6 \wedge T > 3)}{P(T > 3)} \\ &= \frac{P(T > 6)}{P(T > 3)} \end{aligned}$$

and we can use R to evaluate:

```
> (1-pgeom(5,1/6))/(1-pgeom(2,1/6))
[1] 0.5787037
```

Notice that this is exactly what we got in (b) above.

#14 Let's assume people's blood types are i.i.d. (they aren't, since they're inherited genetically and presumably some of the city's residents are related to each other, but we'll ignore that). That means that X, the number of people in a city that have a particular blood type is binomially distributed ($X \sim \text{Binom}(n,p)$), where n is the total number of people in the city and p is the probability of any one person having that blood type). Since we know that on average 1 out of 1,000 people have this blood type, we can say that $X \sim \text{Binom}(1000, p)$ and $E[X] = 1$. In this case, $p = 0.001$.

(a) Since $X \sim \text{Binom}(10000, 0.001)$:

```
> dbinom(0,10000,.001)
[1] 4.517335e-05
```

(b) If the number of people X in a group who have a particular blood type is distributed binomially, then the number of people you'll need to check Y before finding one is distributed geometrically. That is, $Y \sim \text{Geom}(0.001)$. Now what we want to know the smallest value of n such that $P(Y \leq n) > 0.5$:

```
> qgeom(0.5,.001)
[1] 692
```

So, our sample size should be 693 (the 692 people who might fail the test plus the one who finally passes). To double check this, we can confirm the probability of finding at least one success in 693 trials is 0.5:

```
> 1-pbinom(0,693,0.001)
[1] 0.5000998
```

#20 We can use the binomial distribution again. There are 10,000 leaflets (trials), and the probability of any one landing on our block (success) is $\frac{1}{2{,}000}$. So:

```
> dbinom(0,10000,1/2000)
[1] 0.006729527
```

That's the probability that one *particular* block didn't get any leaflets. The probability that at least one block somewhere in the city didn't get any is:

```
> 1-dbinom(0,2000,dbinom(0,10000,1/2000))
[1] 0.9999986
```

#24 I'd do this one in two steps: first, find the probability of one day without mail:

```
> dpois(0,10)
[1] 4.539993e-05
```

If we call getting no mail 'success', we can use the binomial distribution to find the probability of having at least one success in 3,000 days:

```
> 1-dbinom(0,3000,dpois(0,10))
[1] 0.1273344
```

Since the hint (puzzlingly) called for two applications of the Poisson distribution, it's worth verifying that this answer makes sense. We can easily simulate ten years of mail using `rpois` to generate random mail counts, and see how many ten year periods include at least one day with no mail:

```
> R = 100000
> sum(replicate(R,any(rpois(3000,10)==0)))/R
[1] 0.12748
```

Looks good!

G&S pg. 247

#12 The expected value of this game is \$0, no matter what order you place the bets in. But just to verify this: the probability of losing 6 games in a row (and our \$63) is $\frac{1}{2^6} = \frac{1}{64}$. And, the probability of not losing (and making a \$1 profit) is $1 - \frac{1}{64} = \frac{63}{64}$. So,

$$\begin{aligned} E[X] &= 1 \times \frac{63}{64} - 63 \times \frac{1}{64} \\ &= 0 \end{aligned}$$

#19 Let's assume that, when they say that a student "guesses a subset randomly and uniformly," they mean that the student has a 50/50 chance of including each answer in the subset. That is, each answer is a Bernoulli trial with $p = \frac{1}{2}$. For each answer that gets selected, it is either going to be the right answer with probability $\frac{1}{4}$ (we'll assume that all answers have the same probability of being the right one). So, the expected point value of each answer is:

$$\begin{aligned} E[X] &= 0 \times \frac{1}{2} + \left(3 \times \frac{1}{4} - 1 \times \frac{3}{4}\right) \times \frac{1}{2} \\ &= 0 \end{aligned}$$

The point value of a whole question is the sum of the values of its answers, and since the expectation of a sum is just the sum of the expectations, the expected point value of a question is also zero.

#25 First off, this problem is hard! If you found it challenging (or impossible), don't worry. Next, since we'll be doing a couple of related simulations for this problem, we'll start by defining a function for dealing out cards:

```
> stars = 10
> circles = 10
> deal = function() {
+   sample(c(rep('s',stars),rep('c',circles)),stars+circles)
+ }
```

Now we're ready to get started...

(a) If the subject is just guessing, they should guess right about ten times per deck on average. Here's a simple R function to simulate that strategy:

```
> random = function() {
+   cards = deal()
+   p = stars/(stars+circles)
+   guesses = sample(c('s','c'),length(cards),replace=TRUE,prob=c(p,1-p))
```

```
+   sum(cards==guesses)
+ }
>
> mean(replicate(10000,random()))
[1] 9.9889
```

(b) Keep a running count of how many stars and circles have been seen. Each time we need to guess, choose the one that makes up the majority of the remaining cards in the deck.

(c) This one's a little more complicated to implement in R:

```
> informed = function() {
+   cards = deal()
+   counts = list(s = stars,c = circles)
+   right = 0
+   for (c in cards) {
+     guess = names(which.max(counts))
+     if (guess == c) {
+       right = right + 1
+     }
+     counts[[c]] = counts[[c]] - 1
+   }
+   return(right)
+ }
>
> mean(replicate(10000,informed()))
[1] 12.3447
```

(d) The recursion relation:

$$h(S,C) = \frac{S}{S+C}h(S-1,C) + \frac{C}{S+C}h(S,C-1) + \frac{\max(S,C)}{S+C}$$

has three terms. The third term, $\frac{\max(S,C)}{S+C}$, is the probability of guessing the current card correctly and, also, the expected value of the current card. The first term is the expected value of the rest of the deck if the current card is star, times the probability that the current card is actually a star. The second term is same as the first, but for circles. And here's an implementation in R:

```
> h <- function(S,C) {
+   if ((S==0&C==0)|(S==0&C==-1)|(S==-1&C==0)) {
+     return(0)
+   } else {
+    if (S > 0) {
```

```
+    term1 = S/(S+C) * h(S-1,C)
+  } else {
+    term1 = 0
+     }
+    if (C > 0) {
+    term2 = C/(S+C) * h(S,C-1)
+  } else {
+    term2 = 0
+     }
+   return(term1 + term2 + max(S,C)/(S+C))
+ }
+ }
>
> h(10,10)
[1] 12.33773
```