Name: ____________________________

Directions: **Work only on this sheet** (on both sides, if needed). MAKE SURE TO COPY YOUR ANSWERS TO A SEPARATE SHEET FOR SENDING ME AN ELECTRONIC COPY LATER.

Important note: Remember that in problems calling for R code, you are allowed to use any built-in R function, e.g. **choose()**, **sum()**, etc.

1. Consider the simulation on p.62.

(a) (15) Suppose line 7 were replaced by

```
mean(dvals^2) - (mean(dvals))^2
```

(Recall that the **return()** is not ncessary.) What would replace "ED" in the comment in line 6? Your answer should reflect the likely goal of the programmer, and you must use official notation from our course (which is fairly standard in probability books); ED is an example, as is P(D = 3) and so on.

(b) (10) Suppose line 7 were replaced instead by

```
mean(dvals > 18)
```

What would replace "ED" in line 6 in this case?

2. This problem involves the parking space example, Sec. 3.12.3.2.

(a) (15) Find P(D = 12).

(b) (10) Find Var(N).

(c) (10) Good news! I found a parking place just one space away from the destination. Find the probability that I am parked in the same block as the destination.

3. (10) We have two vectors, **x** and **y**. The elements of the latter are either "a", "b" or "c". We want to create a new vector with the following property: For any element in **y** that has the value "a", the new vector's corresponding element will be 100, with the new value being 200 in the case of "b". In the case of "c", the element in the new vector will be the corresponding element of **x**. Write a single line of code that creates and prints out this new vector. You are not allowed to use loops.

Example:

```
> x
[1]  5 12 13  3  4  5
> y
[1] "a" "c" "b" "b" "a" "a"
> # single line of code here,
> # maybe with semicolons
[1] 100  12 200 200 100 100
```

(My grading script will set global variables **x** and **y**. Your code should NOT do this.)

4. (30) The function **rpmf()** below generates **n** random values from a distribution with probability mass function **pmf** and support **supp**. The term *support* for a distribution is just a fancy name for the set of values a random variable with that distribution can take on. The function is is then used to find the approximate probability that in 3 consecutive stops in the bus ridership example (Sec. 2.11), a total of 2 passengers board. Fill in the blanks, without using loops.

```
rpmf <- function(n,pmf,supp) blank (a)

bvals <- rpmf(3000,c(0.5,0.4,0.1),0:2)
m <- matrix( blank (b) )
# the following call sets sums[i]
# to the sum of row i of m
sums <- apply(m,1,sum)
blank (c)
```

Solutions:

1.a Var(D)

1.b P(D > 18)

2.a

$$(1-0.15)^{22}\ 0.15$$

2.b

$$\frac{1-0.15}{0.15^2}$$

2.c

$$\begin{aligned} P(N=12 \mid N=10 \text{ or } N=12) &= \frac{P(N=12)}{PN=10 \text{ or } N=12)} && (1) \\ &= \frac{(1-0.15)^{11}\ 0.15}{(1-0.15)^{9}\ 0.15+(1-0.15)^{11}\ 0.15} && (2) \end{aligned}$$

3.

```
ifelse(y == "b",200,ifelse(y == "a",100,x))
```

4.

```
rpmf <- function(n,pmf,supp) sample(supp,n,prob=pmf,replace=T)

bvals <- rpmf(3000,c(0.5,0.4,0.1),0:2)
m <- matrix(bvals,ncol=3)
# the following call sets sums[i]
# to the sum of row i of m
sums <- apply(m,1,sum)
mean(sums == 2)
```