**CS 542 Hw14: Probability functions**

For each of the following functions, estimate the answer by doing a million trials.

A. double monotonic(unsigned a, unsigned b);

Consider a sequence of b #'s, where each number is randomly chosen to be an int in [0, a). What is the probability that the sequence is monotonic? (That is, either (non-strictly) monotonically increasing or decreasing.)

For instance, if a were 2 and b were 3, we'd have 2^3 == 8 possibilities:

000 001 010 011 100 101 110 111

m m m m m m

where the 6 monotonic possibilities are marked with m, so we'd expect an answer close to 0.75; your function doesn't do the mathematical calculation, it creates a random 3-bit value 1000000 times and returns the fraction of times the 3-bit value is monotonic.

B. double strictlyMonotonic(unsigned a, unsigned b);

Just like the last problem, but now the sequence must be strictly monotonic (no repeated numbers). (So, we'd expect strictlyMonotonic(2, 3) to return 0.)

C. double okNesting(unsigned n);

Randomly shuffle 2n chars, where n of the chars are ( and n of the chars are ).

What is the probability that the parens will properly nest?

For example, with n == 2 we would have

(()) properly nested

()() properly nested

())( not properly nested

)(() not properly nested

)()( not properly nested

))(( not properly nested

D. double duel(double a, double b);

Mr. A and Mr. B can’t reach an accord on the most pleasing way to format a C++ statement block, so they decide to fight a duel with pistols. The bullets are laced with cyanide, so the first hit will be fatal. Mr. A shoots first, and the duelists courteously take turns shooting. Mr. A hits his target with probability a, and Mr. B hits his target with probability b. They continue until someone is hit.

What is the probability that Mr. A wins the duel?

E. double flip(double p, unsigned n, unsigned k);

We have a crooked coin that comes up heads with probability p. We flip the coin n times; what is the probability that we get exactly k heads?

F. double prettyLady(unsigned gridX, unsigned gridY,

unsigned ladyX, unsigned ladyY);

I need to travel from the origin to the point (gridX, gridY), following the integer grid lines. As usual, I require that my distance be minimum (which is gridX + gridY). Among all possible paths, I pick one at random and follow it.

Unknown to me, there is a pretty lady at the (ladyX, ladyY) intersection. What is the probability that I will pass her intersection on the way my (gridX, gridY) destination?

For example, if (gridX, gridY) were (2, 2) and (ladyX, ladyY) were (1,1), then I have 6 possible paths:

NNEE: no pretty lady

NENE: pretty lady

NEEN: pretty lady

ENNE: pretty lady

ENEN: pretty lady

EENN: no pretty lady

So, I would expect prettyLady(2, 2, 1, 1) to return a number close to 0.667

Write a main that calls the above 6 functions. main’s job is to convince you that your functions are probably right. You decide the details of your main.