Project 1
Computational Economics
Due Thursday, September 17

1 Gambler's Ruin

My dad and I love Las Vegas, particularly Craps and Video Poker. Each game in vegas has different payouts. Basic slots return as low as 70 cents for every dollar you put in, while Blackjack and Craps can return over 99.5 cents for every dollar. But video poker is the only game in which you can get a positive expectation. Below is a table of payoffs for a "Full Pay" Dueces Wild machine. (credit Wizard of Odds)

HAND	PAYOFF	COMBINATIONS	PROBABILITY	RETURN
Natural royal flush	800	440202756	0.000022	0.017667
Four deuces	200	4060462824	0.000204	0.040741
Wild royal flush	25	35796957696	0.001796	0.044896
Five of a kind	15	63818309856	0.003202	0.048024
Straight flush	9	83087969280	0.004168	0.037515
Four of a kind	5	1294427430576	0.064938	0.324691
Full house	3	423165297240	0.021229	0.063687
Flush	2	334561280724	0.016784	0.033568
Straight	2	1117664265756	0.056070	0.112141
Three of a kind	1	5674784779512	0.284690	0.284690
Nothing	0	10901423560980	0.546897	0.000000
Total		19933230517200	1.000000	1.007620

(a) Write a While[ ] loop that draws a weighted random number. As a hint, you will want to Accumulate[ ] the probabilities.

(b) Suppose I sit down at this poker machine with $50 and play 200 hands at $1 each. Use your answer from part (a) (or use the weighting option for RandomChoice[ ]) to make a Monte Carlo estimate for the probability I run out of cash at some point during the 200 hands.

(c) What is the probability I end up ahead?

2 Vegas Hotel Projections

You are tasked with forecasting the profits of a Vegas hotel. You come up with the following model:

$$\Pi(t) = (750 + 50t)/(n + 1)$$

where $\Pi(t)$ is the profit in year t, and n is the number of hotels the main competitor owns.

We want to forecast the expected total profit over the next T years, given the following rules:

- You begin in year 1.
- The main competitor currently has no hotels.

- They can only build one hotel at a time.
- Each hotel takes m years to build.

You can set $T = 10,\ m = 2$. To run Monte Carlo simulations of the profit, you will need to randomly generate the number of plants the competitor has each year. An example of a valid construction profile is: {0,1,1,2,2,2,3,3,3,3}. This can be read the following way:

- The competitor begins building on the first day and finishes construction on the second (at that point it counts as a built hotel)
- The competitor starts another immediately after (year 3)
- Finally, they start a third hotel in year 6

To be valid, the profile must be of length T and increment by at most one every other element (for $m = 2$). In the above example, the the total profit over the 10 years is 3850. Find the expected profit. (Note: I will be lenient with how you sample from the space of possible competitor build profiles. Preferably, each possible profile will be equally likely.)

3 Conway's Game of Life

Conway's Game of Life is a "game" with a simple set of rules that yields complex results. It works as follows:

- There is an NxN matrix of alive and dead cells, typically represented by 1's and 0's, respectively.
- Each cell has 8 neighbors surrounding it. Cells on the edge and corners wrap around to the other side.
- Each period, the cells update according to the following rules:
  - If a dead cell has exactly 3 alive neighbors next to it, it becomes alive.
  - If an alive cell has either 2 or 3 alive neighbors next to it, it remains alive.
  - If an alive cell has fewer than 2 alive neighbors next to it, it dies of loneliness.
  - If an alive cell has more than 3 alive neighbors next to it, it dies of overcrowding. Such is life.
- The cells should be updated independently. So, if I am a cell, and my neighbor is currently dead it must be counted as dead to me this period, even if it turns out it will be alive next period.

Program the Game of Life. Run a test by creating a 4x4 matrix with only the 4 corners alive. The four cells should remain stationary. After the test is working, create a 50x50 with the each cell randomly turned on or off. Run it for 100 periods. Plot it using ArrayPlot[ ]. You will likely want to put it in a Do[ ] loop.

Note: Sadly, the code you will write will be embarrassingly slow and ugly. That's okay. You will be rewriting it more efficiently in a few weeks.