Note: Homework is due **5pm** on the due date. Please submit your homework through the dropbox in the Siebel Center basement. Make sure to include your name and **netid** in your homework.

Problem 1 [10pt] This problem will involve simulating a random walk with a Monte Carlo simulation. Suppose a drunkard leaves a bar located at a point (0,0) in a two-dimensional coordinate system. His home is located at the point (3,4). Assume he steps either North, South, East, or West, each with equal probability. What is the probability that at any point in taking 25 steps he ends up at home? Write a Python program to simulate this problem. To determine the probability repeat the walk 30000 times.

Turn in the following:

- Your source code for your Python program
- The output of your program (the probability determined).

Problem 2 [16pt] We wish to perform the DFT on the following signal:

```
\begin{bmatrix} 8 \\ 0 \\ 2 \\ 6 \\ 4 \\ 1 \\ 5 \\ 3 \end{bmatrix}
```

(a) [4pt] Compute the n^{th} roots of unity for n = 8, i.e. find $\omega_8^0, \dots, \omega_8^7$.

(b) [6pt] Construct the matrix F_8 corresponding to the 8-point DFT. You may use Python to construct the matrix if you would like or you may do it by hand. Please show three decimal digits. If you use Python, you must also turn in your source code.

(c) [4pt] Construct the matrix F_8^{-1} corresponding to the 8-point IDFT. You may use Python to construct the matrix if you would like or you may do it by hand. If you use Python, you cannot simply use numpy to get the inverse of the matrix. Please show three decimal digits. If you use Python, you must also turn in your source code.

(d) [2pt] Use Python to compute the DFT of the above signal, x, by computing F_8x using your result for F_8 from part (b). Turn in your source code and the resulting DFT of x.

Problem 3 [4pt] True/False questions

(a) (True/False) The FFT is an algorithm for computing the DFT, not its own transform.

(b) **(True/False)** The cost of computing the FFT is $\mathcal{O}(n \log_2 n)$ and the cost of computing the DFT is $\mathcal{O}(n^3)$.

(c) $(\mathbf{True}/\mathbf{False})$ The FFT is always done recursively.

(d) **(True/False)** For any real sequence, x, let y be the DFT of x. Then, the first component of y, y_0 , is always real and is the sum of the components of x.