1. (3 points) Using 140 characters or less, post a synopsis of your favorite movie to the course piazza space under the "HW0 tell me something!" notice, so that your post is visible to everyone in the class, and tagged by #HW0num1. Also, use Piazza's code-formatting tools to write a *private* post to course staff that includes at least 5 lines of code. It can be code of your own or from a favorite project–it doesn't even have to be syntactically correct–but it must be formatted as a code block in your post, and also include the tag #HW0num1. (Hint: Check https://piazza.com/product/features). Finally, please write the 2 post numbers corresponding to your posts here:

Favorite Movie Post (Public) number:	639
Formatted Code Post (Private) number:	656

2. (12 points) Simplify the following expressions as much as possible, **without using an calculator (either hardware or software)**. Do not approximate. Express all rational numbers as improper fractions. Show your work in the space provided, and write your answer in the box provided.

(a) $\prod_{k=2}^{n}(1-\frac{1}{k^2}) = \prod_{K=2}^{n}\left(\frac{K^2-1}{K^2}\right) = \frac{3}{4}\ \frac{8}{9}\ \frac{15}{16}\ \frac{24}{25} \cdots \frac{(n-1)^2-1}{(n-1)^2}\ \frac{n^2-1}{n^2}$

$\frac{3}{4}\ \frac{2}{3}\ \frac{5}{8}\ \frac{3}{5}$

$\frac{3}{4}\ \frac{4}{6}\ \frac{5}{8}\ \frac{6}{10}$

numerator $= n+1$

denominator $= 2n$

Answer for (a):	$n+1/2n = \prod_{K=2}^{n}(1-\frac{1}{K^2})$

(b) $3^{1000} \bmod 7 = (3^4)^{250} \bmod 7$

$= 4^{250} \bmod 7$

$= 4^{(2)^{5}+10} \bmod 7$

$= 2^{15} + 4^{10} \bmod 7$

$= 2^{15} + 2^{5} \bmod 7$

$= 2^{20} \bmod 7$

$= 4^{4} \bmod 7$

$= 2 \cdot 2 \bmod 7$

$= \boxed{4}$

$3^4 \bmod 7 = 4$

$4^2 \bmod 7 = 2$

$4^{10} \bmod 7 = (4^2)^5 \bmod 7 = 2^5 \bmod 7$

$2^5 \bmod 7 = 4$

Answer for (b):	4

(c) $\sum_{r=1}^{\infty}(\frac{1}{2})^r = \frac{1}{1-\frac{1}{2}} = \frac{1}{\frac{1}{2}} = 2$

Answer for (c):	2

(d) $\frac{\log_7 81}{\log_7 9}$?

Answer for (d):	2

(e) $\log_2 4^{2n} = 2n \log_2 4$
$= 2n(2)$
$= 4n$

Answer for (e):	4n

(f) $\log_{17} 221 - \log_{17} 13 = \frac{\log_{17} 221}{\log_{17} 13}$

$= 17$

Answer for (f):	17

3. (8 points) Find the formula for $1+\sum_{j=1}^{n} j!j$, and show work proving the formula is correct using induction.

$1+\sum_{j=1}^{n} j!j = 1+1+4+18+96+600+\dots+n!n$

$=(n+1)!$

Formula:	$(n+1)!$

$n=1 \dots 2$
$n=2 \dots 6$
$n=3 \dots 24$
$n=4 \dots 120$
$n=5 \dots 720$

Base Case $n=1$

$1+\sum_{j=1}^{1} j!j = 1+1=2$ ✓ $(1+1)!=2$ ✓

Inductive Hypothesis

Suppose we have a set $n=\{1,2,\dots,K\}$ for which the relation $1+\sum_{j=1}^{n} j!j=(n+1)!$ holds true. Then $K+1$ must also hold true.

$((K+1)+1)!=(K+2)!$
$=(K+1)!(K+2)$ ✓

$1+\sum_{j=1}^{K+1} j!j = 1+\sum_{j=1}^{K} j!j+(K+1)!(K+1)$
$=(K+1)!+(K+1)!(K+1)$
$=(K+1)!(1+(K+1))$
$=(K+1)!(K+2)$ ✓ Q.E.D.

4. (8 points) Indicate for each of the following pairs of expressions $(f(n), g(n))$, whether $f(n)$ is O, Ω, or Θ of $g(n)$. Prove your answers to the first two items, but just GIVE an answer to the last two.

(a) $f(n)=4^{\log_4 n}$ and $g(n)=2n+1$.
$=n$

$n \le 2n+1$ for $n<0$

Answer for (a):	$f(n) = O\ g(n)$

(b) $f(n)=n^2$ and $g(n)=(\sqrt{2})^{\log_2 n}$.

$n^2 \ge (\sqrt{2})^{\log_2(n)}$

Answer for (b):	$f(n) = \Omega\ g(n)$

(c) $f(n) = \log_2 n!$ and $g(n) = n \log_2 n$.

Answer for (c):	$f(n) = \Omega\ g(n)$

(d) $f(n) = n^k$ and $g(n) = c^n$ where k, c are constants and c is >1.

Answer for (d):	$f(n) = O\ g(n)$

5. (9 points) Solve the following recurrence relations for integer n. If no solution exists, please explain the result.

(a) $T(n) = T(\frac{n}{2}) + 5$, $T(1) = 1$, assume n is a power of 2.

$= T(\frac{n}{4}) + 5 + 5$

$= T(\frac{n}{8}) + 5 + 5 + 5$

$= T(\frac{n}{2^K}) + K(5)$

$= 1 + \log_2 n(5)$

$\frac{n}{2^K} = 1$

$n = 2^K$

$\log_2 n = K$

Answer for (a):	$1 + 5 \log_2 n$

(b) $T(n) = T(n-1) + \frac{1}{n}$, $T(0) = 0$.

$= T(n-2) + \frac{1}{n} + \frac{1}{n}$

$= T(n-3) + \frac{1}{n} + \frac{1}{n} + \frac{1}{n}$

$= T(n-K) + \frac{K}{n}$

$= T(n-n) + \frac{n}{n}$

$= T(0) + 1$

$= 0 + 1 = \boxed{1}$

$n - K = 0$

$n = K$

Answer for (b):	1

(c) Prove that your answer to part (a) is correct using induction.

base Case $n=2$

$T(2) = T(\frac{2}{2}) + 5 = 6$ ✓

$1 + 5\log_2 2 = 6$ ✓

Suppose given a set $n = \{1, 2, 4, 8, \ldots K\}$ the relation $T(\frac{n}{2}) + 5 = 1 + 5\log_2 n$ given $T(1)=1$ holds true.

Then $2K$ must also hold true $1 + 5\log_2(2K)$ ✓

Test $K+1$: $T(2K) = T(\frac{2K}{2}) + 5$

$= T(K) + 5$

$= 1 + 5\log_2 K + 5$

$= 1 + 5\log_2(2K)$ ✓ Q.E.D.

6. (10 points) Suppose function call parameter passing costs constant time, independent of the size of the structure being passed.

(a) Give a recurrence for worst case running time of the recursive Binary Search function in terms of n, the size of the search array. Assume n is a power of 2. Solve the recurrence.

$T(n) = T(\frac{n}{2}) + X$, $T(1) = C$

where X is the amount of time it takes to do a single comparison of searched item to desired.

and C is the constant costs of running the algorythum.

Recurrence:	$T(n) = T(\frac{n}{2}) + X$, $T(1) = C$
Base case:	$\frac{n}{2}$
Recurrence Solution:	$C + X\log_2 n$

start in the middle

(b) Give a recurrence for worst case running time of the recursive Merge Sort function in terms of n, the size of the array being sorted. Solve the recurrence.

$T(n) = T(\frac{n}{2}) + Xn$, $T(1) = C$

Recurrence:	$T(n) = T(\frac{n}{2}) + Xn$, $T(1) = C$
Base case:	1
Running Time:	$O(n \log n)$

7. **(10 points) Consider the pseudocode function below.**

```
derp( x, n )
    if( n == 0 )
        return 1;
    if( n % 2 == 0 )
        return derp( x^2, n/2 );
    return x * derp( x^2, (n - 1) / 2);
```

$16 * 256 * 1$

(a) What is the output when passed the following parameters: $x = 2$, $n = 12$. Show your work (activation diagram or similar).

$x=2, n=12$

$4, 6$

$16*$ $16, 3$

$256*$ $256, 1$

$256^2, 0$

1

Answer for (a):	4096

(b) Briefly describe what this function is doing.

The function runs $\lceil \log_2 n \rceil + 1$ times if $\log_2 n \leq 2$ the output will be $(8^2)^2)^2)\cdots$ $n-3$ times.

(c) Write a recurrence that models the running time of this function. Assume checks, returns, and arithmetic are constant time, but be sure to evaluate all function calls. [Hint: what is the *most* n could be at each level of the recurrence?]

for my algorithm no matter what n is it gets dropped to the nearest power of two < n.

$T(n) = T(\frac{n}{2})^2$, $T(4) = 8$

(d) Solve the above recurrence for the running time of this function.

$O(\log n)$