Favorite Movie Post (Public) number:	@1074
Summer Travel Post (Private) number:	@ 1076

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) $\prod_{k=2}^{n}(1-\frac{1}{k^2})$

$= \prod_{k=2}^{n} \frac{k^2-1}{k^2}$

$(1-\frac{1}{4})\cdot(1-\frac{1}{9})\cdot(1-\frac{1}{16})$

$\frac{3}{4}\cdot\frac{8}{9}\cdot\frac{15}{16}$

$\prod_{k=2}^{n} \frac{(k+1)(k-1)}{k^2} = \frac{n+1}{2n}$

Answer for (a):	$\frac{n+1}{2n}$

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

$3^0 \bmod 7 = 1$

$3^1 \bmod 7 = 3$

$3^2 \bmod 7 = 2$

$3^3 \bmod 7 = 6$

$3^4 \bmod 7 = 4$

$3^5 \bmod 7 = 5$

$3^6 \bmod 7 = 1$

Every power that's a multiple of 6 = 1.

Closest multiple of 6 to 1000 → 996

$3^{996} \bmod 7 = 1$ $\quad$ $3^{998} \bmod 7 = 2$

$3^{997} \bmod 7 = 3$ $\quad$ $3^{999} \bmod 7 = 6$

$3^{1000} \bmod 7 = 4$

Answer for (b):	4

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

$\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}+\frac{1}{32}+\frac{1}{64}+\frac{1}{128}+\frac{1}{\ }=$

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

Answer for (c):	1

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

$$\frac{\log_7(3^4)}{\log_7(3^2)} = \frac{4\cancel{\log_7(3)}}{2\cancel{\log_7(3)}} = 2$$

Answer for (d):	2

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

$2n \log_2 4$

$2^x = 4^{2n}$

$2^x = (2^2)^{2n}$

$x = 4n$

Answer for (e):	4n

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

$\log_{17}\left(\frac{221}{13}\right) = \log_{17} 17$

Answer for (f):	1

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

Formula:	$(n+1)!$

$1 + 1 + (1\cdot 2)2 + (1\cdot 2\cdot 3)3 + (1\cdot 2\cdot 3\cdot 4)4 = 1+1+4+18+96 = 120$

$\sum_{j=1}^{n} j!j = (n+1)! - 1$

$1 + \sum_{j=1}^{m} j!j$

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 $g(n) = 2n+1$.

Big Θ notation

$n = 16$

$4^{\log_4 16} = 16$

$f(n) = n \qquad g(n) = 2n+1$

$n \le 2n+1$

$C_1 g(n) \le f(n) \le C_2 g(n)$

$f(n) \in \Theta(g(n))$

Answer for (a):	$f(n) \in \Theta(g(n))$

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

$2^{\frac{1}{2}\log_2 n}$

$n = 8$

$f(n) = 64 \qquad g(n) = \sqrt{n}$

n^2 grows quicker than $\sqrt{n}$

$f(n) \ge g(n)$

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

$g(n) \le f(n) \le g(n)$

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

$\log_2 n!$ vs $n \log n$

$C_1 \leq f(n) \leq C_2 g(n)$

Answer for (c):	$f(n) \in \Theta(g(n))$

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

$f(n) = n^k \quad g(n) = c^n$

$n^k < c^n$

$n^k < c^n$

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

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.

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

$T(2) = 1 + 5 = 6$

$T(\frac{n}{2}) = T(2) = 6$, $n = 4$

$T(4) = 6 + 5 = 11$

$T(\frac{n}{2}) = T(4) = 11$, $n = 8$

$T(8) = 11 + 5 = 16$

$T(1) = 1$, $T(2) = 6$, $T(8) = 16$, $T(16) = 21$

$5 \log_2 n + 1$

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

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

$T(1) = T(0) + \frac{1}{1} = 1$

$T(2) = 1 + \frac{1}{2} = 1.5 = 3/2$

$T(3) = 1.5 + \frac{1}{3} = 11/6$

$T(4) = \frac{11}{6} + \frac{1}{4} = \frac{25}{12}$

$T(5) = \frac{25}{12} + \frac{1}{5} = \frac{137}{60}$

$T(6) = \frac{49}{20}$

$T(1) = 1$, $T(2) = 1.5$

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

Harmonic Series

$T(n) = \sum_{k=1}^{n} 1/k$

Answer for (b):	$T(n) = \sum_{k=1}^{n} 1/k$

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

$\sum_{k=1}^{n} 1/k$ $T(1) = \frac{1}{1} = 1$

holds true

$T(n-1) \sum_{k=1}^{n-1} \frac{1}{k}$ equal → proves that

$= T((n+1)-1) + \frac{1}{n+1} = \sum_{k=1}^{n} \frac{1}{k} + \frac{1}{n+1} = \sum_{k=1}^{n+1} \frac{1}{k}$ $T(n) = \sum_{k=1}^{n} \frac{1}{k}$

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

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.

$T(n) = T\left(\frac{n}{2^k}\right) + kd$

$T(1) = C_0$

$\frac{n}{2^k} = 1 \Rightarrow n = 2^k$

$k = \log_2 n = \lg(n)$

$= \Theta(\lg(n))$

Recurrence:	$T(n) = T(\frac{n}{2}) + d$
Base case:	$T(1) = C_0$
Recurrence Solution:	$T(n) = O(\lg(n))$

(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.

for $k=1$ $T(n) = 2^k T\left(\frac{n}{2^k}\right) + kn = 2T\left(\frac{n}{2}\right) + n$

$T(n) = 2^{k-1} T\left(\frac{n}{2^{k-1}}\right) + (k-1)n$

$= 2^k T\left(\frac{n}{2^k}\right) + kn$

when $\frac{n}{2^k} = 1$ or $n = 2^k$
$k = \log_2 n$

$T(n) = nT(1) + n\log_2 n =$

$= n\log_2 n + n$

Recurrence:	$T(n) = 2T(\frac{n}{2}) + kn$
Base case:	$T(1) = C_0$
Running Time:	$T(n) = O(n\lg(n))$

O's $\Rightarrow T(n) = O(n\lg n)$

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);
```

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

derp(4, 6) → x=16 n=3

~~2 * derp(4, 6)~~

16 (derp(16², 1))

16(16²(derp((16²)², 0)))

4096

Answer for (a):	4096

(b) Briefly describe what this function is doing.

x^n

(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?]

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

$T(0) = c$

$T(1) = d$

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

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

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

$T(n) = T((\frac{n}{2^k})) + ck$

$T(n) = d + c\log_2 n$

$T(n) = o(\log_2 n)$