HW 1

Due 23:59 January 11, 2019

CS ID 1: _____

CS ID 2: _____

Instructions:

1. Do not change the problem statements we are giving you. Simply add your solutions by editing this latex document.

2. Take as much space as you need for each problem. You'll tell us where your solutions are when you submit your paper to gradescope.

3. Export the completed assignment as a PDF file for upload to gradescope.

4. On gradescope, upload only **one** copy per partnership. (Instructions for uploading to gradescope will be posted on the HW1 page of the course website.)

Academic Conduct: I certify that my assignment follows the academic conduct rules for of CPSC 221 as outlined on the course website. As part of those rules, when collaborating with anyone outside my group, (1) I and my collaborators took no record but names away, and (2) after a suitable break, my group created the assignment I am submitting without help from anyone other than the course staff.

1. (2 points) Using 140 characters or less, post a synopsis of your favorite movie to the course piazza space under the "HW1 tell me something!" notice, so that your post is visible to everyone in the class, and tagged by #HW1num1. 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 #HW1num1. (Hint: Check http://support.piazza.com/customer/portal/articles/1774756-code-blocking). Finally, please write the 2 post numbers corresponding to your posts here:

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

2. (16 points) In this problem, you will be a math detective! Some of the symbols and functions may not be familiar to you, but with a little digging, reading, and observing, you'll be able to figure them out. Your task is to simplify each of 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. You may assume that n is an integer greater than 1. Show your work in the space provided, and write your final answer in the box.

   (a)   i. $\displaystyle\sum_{k=1}^{n} k2^{k+1} - \sum_{k=1}^{n} k2^{k}$

Answer for (a.i):	

ii. $\displaystyle\prod_{i=1}^{n-1} \frac{2i}{n-i}$

Answer for (a.ii):	

(b)   i. 15^{333} mod 2

Answer for (b.i):

ii. 32^{333} mod 15

Answer for (b.ii):

(c)   i. $\displaystyle\sum_{r=0}^{n} \left(\frac{1}{n}\right)^{r}$

Answer for (c.i):

ii. $\displaystyle\sum_{r=3}^{\infty} \left(\frac{2}{3}\right)^{r}$

Answer for (c.ii):

(d)   i. $4^{(\log_2 n)/2}$

Answer for (d.i):

ii. $\dfrac{\log_7 1024}{\log_7 32} - \dfrac{\log_3 1024}{\log_3 32}$

Answer for (d.ii):

iii. $\dfrac{\log_2 n}{\log_{32} n}$

Answer for (d.iii):

3. (12 points)

    (a) (3 points) Find a closed form expression for $f(n) = \sum_{j=1}^{n}(2j - 1), \forall n \geq$ _____ (fill in the blank with the smallest reasonable value!), and use induction to prove the formula is correct.

Formula:	

    (b) (2 points) Find a closed form expression for

$$\frac{1 + 3 + 5 + \ldots + (2k - 1)}{(2k + 1) + (2k + 3) + \ldots + (4k - 1)}.$$

Formula:	

(c) Give a rigorous, direct proof that your solution to part (b) is correct.

    i. (2 points) Express the fraction using a new variable that quantifies the number of terms in the numerator and in the denominator. Be sure to bound that variable.

Formula:	

    ii. (5 points) Complete the proof. A concise solution will apply the result from part (a) three times.

4. (8 points) Prove that $f(n) \in O(g(n))$ if and only if $g(n) \in \Omega(f(n))$.

5. (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. Note, if you choose O or Ω, you must also show that the relationship is not Θ. You may use the result from problem 4, if you find it useful.

(a) $f(n) = n \log(n^2 + 1) + n^2 \log n$, and $g(n) = n^2$.

Answer for (a): [   ]

(b) $f(n) = n^2$, and $g(n) = \frac{1}{2}n^3 - 2n^2 - 3n - 17$.

Answer for (b):	

6. (16 points) For each C++ function below, give the tightest asymptotic upper bound
that you can determine for the function's **runtime**, in terms of the input parameter.
Where prompted, also compute the return value of the function. For all function calls,
assume that the input parameter $n \geq 2$.

(a)

```cpp
int raspberryscone(int n) {
    int s = 0;
    for (int q = 0; q < n; q++)
        s = s + q;
    for (int r = s; r > 2; r--)
        s--;
    return r * s;
}
```

Return value for (a): _____

Running time of (a): _____

(b)

```cpp
int peanutbuttercheesecake(int n) {
    int i = 0;
    int j = 4 * n;
    for (int k = j * j; k > 1; k = k / 2) {
        i++;
    }
    return i * i / 4;
}
```

Return value for (b): _____

Running time of (b): _____

(c) ——

```
int almondcaramelbrittle(int n) {
    int c = 0;
    int s;
    for (s = 1; s < n * n * n * n; s = s * n) {
        c++;
    }
    return c * s;
}
```

Return value for (c):	

Running time of (c):	

(d) ——

```
void vanillacustardblueberrytorte(int n) {
    int j = 1;

    for (int k = 0; k < raspberryscone(n/2) * almondcaramelbrittle(n); k++)
        j = 2 * j;

    for (int m = j; m > 1; m = m / 2)
        cout << "I am having so much fun with asymptotics!" << endl;
}
```

Running time of (d):	

(e) ——

```
int pirouline(int n) {
    int j = 0;
    for (int k = 0; k < raspberryscone(n) * n; k++)
        j = j + 2;
    return peanutbuttercheesecake(almondcaramelbrittle(j));
}
```

Running time of (e):	

Blank sheet for extra work.