CSC 212: Data Structures and Abstractions
Fall 2018
University of Rhode Island
Weekly Problem Set #2

Due Thursday 9/27 at the beginning of class. Please turn in neat, and organized, answers hand-written on standard-sized paper **without any fringe**. At the top of each sheet you hand in, please write your name, and ID.

1. Simplify the following:
   - $\log_2 xy^2 - \log_2 x^2 - 2\log_2 y$
   - $\log_2(16x^2)^{\frac{1}{3}}$
   - $\log_3(9x^4) - \log_3(3x)^2$
2. Solve for x: $\log_2 \frac{x^2}{2} = 5$
3. Evaluate: $\sum\limits_{x=0}^{3}(5 + \sqrt{4^x})$
4. Solve the following: $\sum\limits_{n=0}^{10}(-n)$
5. Prove that: $\sum\limits_{i=1}^{x} i = \frac{(x+1)x}{2}$
6. Rewrite the following expression into its closed form (i.e. without the sigma): $\sum_{i=1}^{4}(2 + i^2)$.
7. Based on the given data, please clasify each of the following as linear, exponential, logarithmic, or none of the above. If it is none of the above, try to reason what type of curve it may be.
   - $f(0) = 4, f(1) = 6, f(2) = 9$
   - $f(0) = 6, f(10) = 8, f(20) = 10$
   - $f(0) = 80, f(0.1) = 60, f(0.2) = 45$
   - $f(1) = 10, f(10) = 20, f(100) = 30$
   - $f(0) = 2, f(3) = 12, f(5) = 240$
8. Rank the following functions by their asymptotic growth rate in ascending order. In your solution, group those functions that are big-Theta of one another (all log functions are base 2):

$$\begin{array}{ccccc}
6 \cdot n \log n & 2^{100} & \log\log n & \log^2 n & 2^{\log n} \\
2^{2^n} & \lceil\sqrt{n}\rceil & n^{0.01} & 1/n & 4n^{3/2} \\
4^n & n^3 & n^2 \log n & 4^{\log n} & \sqrt{\log n}
\end{array}$$

9. For each of the following, give both a big-Oh characterization in terms of n, and an exact characterization (count additions and multiplications):

(a)
```
EX: For the following, the big-Oh characterization is: O(n),
the exact characterization is n.
s = 1
for i = 1 to n do
    s = s * i
```

(b)
```
s = 1
for i = 1 to 4n do
    s = s * i
```

(c)
```
s = 1
for i = 1 to n*n*n do
    s = s * i
```

(d)
```
s = 0
for i = 1 to 4n do
    for j = 1 to i do
        s = s + i
```

(e)
```
s = 0
for i = 1 to n*n do
    for j = 1 to i do
        s = s + i
```

(f)
```
s = 1
for i = 1 to n do
    for j = 1 to n do
        for k = 1 to n do
            s = s * i
```