CSM61B

Asymptotics Practice Problems

Alex Kazorian

March 30, 2017

1. Order the following functions so that $f_i \in O(f_j) \iff i < j$:
   (a) $f_1(n) = 3^n$
   (b) $f_2(n) = \frac{n}{2}$
   (c) $f_3(n) = 2^{10000}$
   (d) $f_4(n) = 2^{\log_2 n}$
   (e) $f_5(n) = \log n$
   (f) $f_6(n) = n + n^2 \log n$
   (g) $f_7(n) = n!$
   (h) $f_8(n) = 1.001^n + n^3$
2. Provide the tightest bound on $f(n)$ in terms of $g(n)$ by saying $f = O(g)$, $f = \Omega(g)$, or $f = \Theta(g)$.
   (a) $f(n) = \log_2 n$
   $g(n) = \log_3 n$
   (b) $f(n) = \log n^2$
   $g(n) = \log n$
   (c) $f(n) = n - 100$
   $g(n) = n + 10000$
   (d) $f(n) = 2^{1.1n}$
   $g(n) = \sum_i^n i^2$
3. Find the tightest bound of the function $f(n) = x \sin x^2$ in both the worst and best case.
4. Provide the worst and best runtime for the following methods in terms of N. However, if the notion of worst and best case runtime does not make sense for some program, then provide the single best asymptotic bound on the function.
   (a) Here consider N to be the length of the array and p to be initialized to N.

```
public int foo(int[] fighters, int p) {
  if (p != 0) {
    return fighters[p];
  }
  if (p % 2 == 0) {
    return foo(fighters, p / 2);
  }
  System.arraycopy(fighters, 0, fighters, 0, p + 1);
  fighters[p] = p;
  return foo(fighters, p + 1);
}
```

(b) Here consider N to be the length of the array and assume k, p are initialized to 0 and `seen` is an N x N matrix that is globally accessible.

```
public boolean jumpMan(boolean[] arr, int p, int k) {
  if (k >= arr.length) {
    seen[p][k] = 1;
  } else if (arr[p] == false) {
    seen[p][k] = 0
  }
  seen[p][k] = jumpMan(arr, p + k, k)
             || jumpMan(arr, p + k - 1, k - 1)
             || jumpMan(arr, p + k + 1, k + 1);
  return seen[p][k];
}
```