Solution to Problem Set 5

Date: 18.10.2013 Not graded

Problem 1. Since **BubbleSort** swaps neighboring elements starting from a_1, if $a_1, a_2, \ldots, a_k$ are in decreasing order (i.e., $a_1 > a_2 > \ldots > a_k$) then we require the maximum number of computations to sort them. In particular, the number of required operations is given by

$$\sum_{i=1}^{k-1} i = \frac{k(k-1)}{2} = O(k^2).$$

Indeed, at the end of step i, the elements from position $k-i+1$ to position k are already sorted in increasing order by definition of the algorithm. Hence, no more than $k-i-1$ swaps are possible at step $i+1$ and the total number of swaps is upper bounded by $\sum_{i=1}^{k}(k-i) = \sum_{i=0}^{k-1} i = \frac{k(k-1)}{2}$. The upper bound is achieved in the worst case scenario described above.

Problem 2.

a) **Algorithm 1** always terminates. **Algorithm 2** terminates if and only if $m \geq 0$.

b) For **Algorithm 1**, the main loop runs exactly n times, with $j = 1, \ldots, n$. For every j, the internal loop runs $m = 2^{j-1}$ times. Thus, the text gets printed $\sum_{j=1}^{n} 2^{j-1} = 2^n - 1$ times.

Concerning **Algorithm 2**, let us look at the evolution of $\ln m$ (natural logarithm of m). Then, as long as $m > 0$ and if t is initialized to $\ln m$, the algorithm is equivalent to the following:

Algorithm 2 '

Require: t: real number
```
while t > 1 do
    t ← t − ln(3/2)
    print "Hello world"
```

The number of times that Algorithm 2' prints "Hello world" is given by

$$\begin{cases} \left\lceil \dfrac{t-1}{\ln 3/2} \right\rceil & \text{if } t > 1, \\ 0 & \text{otherwise,} \end{cases}$$

which yields for Algorithm 2 (i.e., for the initial problem)

$$\begin{cases} \left\lceil \dfrac{\ln m-1}{\ln 3/2} \right\rceil & \text{if } m > e, \\ 0 & \text{if } 0 \leq m \leq e, \\ \infty & \text{otherwise.} \end{cases}$$

Problem 3.

Lemma. *Let $p(n) = a_k n^k + \cdots + a_1 n + a_0$ be a polynomial with real coefficients and $a_k \neq 0$. Then there are constants $c_1, c_2 > 0$ and $n_0 \in \mathbb{N}$ such that for $n \geq n_0$, $c_1 n^k \leq |p(n)| \leq c_2 n^k$. In other words, $p(n) = \Theta(n^k)$.*

Proof. Pick $c_2 = \sum_{j=0}^{k} |a_j|$. Then for $n \geq 1$ we have

$$|p(n)| = |\sum_{j=0}^{k} a_j n^j| \leq \sum_{j=0}^{k} |a_j n^j| \leq \sum_{j=0}^{k} |a_j n^k| = \sum_{j=0}^{k} |a_j| n^k = c_2 n^k.$$

Let $b = \sum_{j=0}^{k-1} |a_j|$. Then for $n \geq 1$ we have

$$|p(n)| = |\sum_{j=0}^{k} a_j n^j| \geq |a_k n^k| - \left|\sum_{j=0}^{k-1} a_j n^j\right| \geq |a_k| n^k - \sum_{j=0}^{k-1} |a_j| n^j \geq |a_k| n^k - \sum_{j=0}^{k-1} |a_j| n^{k-1} = |a_k| n^k - b n^{k-1}.$$

Pick $c_1 = \frac{|a_k|}{2}$. Then for $n \geq \frac{2b}{|a_k|}$ we have

$$|p(n)| \geq |a_k| n^k - b n^{k-1} = |a_k| n^k - \frac{b n^k}{n} \geq |a_k| n^k - \frac{b |a_k| n^k}{2b} = c_1 n^k.$$

□

a) n^2. Indeed, $f(n) = \sum_{i=0}^{n} 3i + 1 = 3\frac{n(n+1)}{2} + n + 1 = \Theta(n^2)$ by the previous lemma.

b) n^2. Indeed, $g(n) = \sum_{i=0}^{n-1} 2i + 1 = 2\frac{n(n-1)}{2} + n = \Theta(n^2)$ by the previous lemma.

c) n. Indeed, let $p(n) = 3 - 2n^4 - 4n$ and $q(n) = 2n^3 - 3n$. By the above lemma there exist positive constants c_1, c_2, c_1', c_2' and n_0 such that for $n > n_0$

$$c_1 n^4 \leq |p(n)| \leq c_2 n^4 \qquad \text{and} \qquad c_1' n^3 \leq |q(n)| \leq c_2' n^3.$$

Then

$$\frac{c_1}{c_2'} n \leq \left|\frac{p(n)}{q(n)}\right| \leq \frac{c_2}{c_1'} n.$$

d) n^4. Indeed, $h(n) = \sum_{i=0}^{n^2} i = \frac{n^2(n^2+1)}{2} = \Theta(n^4)$ by the previous lemma.

e) n^2. Indeed, $\lceil n+2 \rceil \leq n+3$ and $\lceil n/3 \rceil \leq n/3 + 1$.

f) n^4. Indeed, $3n^4 + \log_2 n^8$ is upper bounded by $4n^4$ for all $n \geq 1$.

Problem 4.

$\frac{1}{n^n}$	$0,0097^n$	$\left(-\frac{1}{2}\right)^n$	n^{-e^4}	$-\frac{1}{\sqrt{n}}$	42	$\log\log\log n$
n	$n \cdot \log\log n$	$n \cdot \log n$	n^{2013}	$n^{\log n}$	$e^{\sqrt{n}}$	2^n
$n \cdot 2^n$	2.0001^n	e^n	$(-3)^n$	e^{2n}	$n!$	

Problem 5.

a) $\left|\frac{3n-8-4n^3}{2n-1}\right| \leq \left|\frac{3n^3+8n^3+4n^3}{2n-n}\right| = 15|n^2|$

b) $1^3 + 2^3 + \cdots + n^3 \leq \underbrace{n^3 + n^3 + \cdots + n^3}_{n \text{ times}} = n^4$

c) $4^n + n \cdot 2^n \leq 4^n + 4^n = 2 \cdot 4^n$, because $n \leq 2^n$ is true for all n.