

# CS 260 Homework 2

Jason Gallagher

7/7/2016

## 1 Problem 1.11

$$g(1) = \begin{cases} n^2 & \text{even } x \geq 0 \\ n^3 & \text{odd } x \geq 1 \end{cases}$$
$$g(2) = \begin{cases} n & 0 \leq x \leq 100 \\ n^3 & x \geq 100 \end{cases}$$
$$g(3) = \begin{cases} n^2 & x \geq 0 \\ n^3 & x \geq 1 \end{cases}$$

Indicate for each distinct pair  $i$  and  $j$  whether  $g_i(n)$  is  $O(g_j(n))$  and whether  $g_i(n)$  is  $\Omega(g_j(n))$

	$g_1$	$g_2$	$g_3$
$O(g_1)$	t	t	f
$\Omega(g_1)$	t	t	f
$O(g_2)$	t	t	f
$\Omega(g_2)$	t	t	f
$O(g_3)$	f	f	t
$\Omega(g_3)$	f	f	t

## 2 Problem 1.13

Show that the following statements are true.

- 1.

$$17 \leq 17n^0$$

for :  $n = \text{any}$

2.

$$\begin{aligned}n(n-1)/2 &= (n^2 - n)/2 \\(n^2 - n)/2 &\leq 1/2 n^2 \\n^2 - n &\leq n^2 \\when : n &\geq 0\end{aligned}$$

3.

$$\begin{aligned}10n^2 &< 10n^3 \\n^2 &\leq n^3 \\for : n &\geq 10\end{aligned}$$

4.

$$\begin{aligned}1^k + 2^k \dots + n^k &\leq O(n^{(k+1)}) \\1^k + 2^k \dots + n^k &\leq n^k + 1 \\1^k + 2^k \dots + n^k &\leq n * n^k \\(1/n)^k + (2/n)^k \dots n^k - 1 &\leq n^k\end{aligned}$$

### 3 Problem 1.16

1.

$$\frac{3^n}{2}$$

2.

$$\frac{1^n}{3}$$

3.

$$\sqrt{n} \log^2(n)$$

4.

$$\frac{n}{l} \log(n)$$

5.

$$\log^2(n)$$

6.

$$\sqrt{n}$$

7.

$$\log(n)$$

8.

$$\log(\log(n))$$

9.

$$17$$

## 4 Problem 1.18

A.

$$t(j) = \sum_0^{j/2} 2^j$$

B

$$t(j) = O(2^n)$$

## 5 Problem 2.9

Write a procedure to interchange the elements at positions  $p$  and  $\text{NEXT}(p)$  in a singly linked list. The following procedure was intended to remove all occurrences of element  $x$  from list  $L$ . Explain why it doesn't always work and suggest a way to repair the procedure so it performs its intended task.

**Data:** One elementtype and one list as input

**Result:** List with element type removed

procedure delete(elementType  $x$ , list  $l$ )

$p = \text{position}$

    while  $p \neq \text{END}(L)$  do begin

        if  $\text{RETRIEVE}(p, L) = x$  then

$\text{DELETE}(p, L);$

$p = \text{NEXT}(p, L);$

    end

end delete

By removing the element from the list, the list is modified and therefore the indexing of the list changes. This is not accounted for when incrementing through all of the positions for example:

$\text{list} = [12, 32, 13, 13, 46]$

$x = 13$

when  $p = 2$ :

$\text{DELETE}(p, L)$

$p = \text{next}(p, L)$

    \#  $\text{list} = [12, 32, 13, 46]$   $\text{list.len} = 4$

    \#  $p = 3$

$\text{list}[3] = 46$

$\text{list}[2] = 13$

When p is incremented after an element is removed, the following element is skipped. This occurs because the list length is decreased at the same time as the list iterator is being incremented.

Revised Version This fix only increments the list iterator if an element is not

**Data:** One elementtype and one list as input

**Result:** List with element type removed

procedure delete(elementType x, list l)

    p = position

    while p != END(L) do begin

        if RETRIEVE(p,L) = x then

            DELETE(p,L);

        else then

            p = NEXT(p,L);

    end

end delete

removed, this ensures no elements are skipped.

## 6 Problem 2.11

Times each function is hit

FIRST:

$$= \frac{n(n+1)}{2}$$

NEXT:

$$n + \frac{n(n+1)}{2} + \sum_1^n n^2$$

LAST:

$$n + 1$$