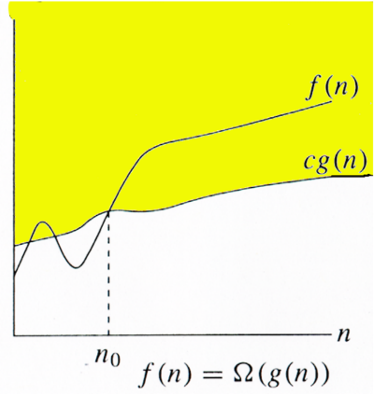
Big Omega



Definition Of Big Omega

- Let $f(n)$ and $g(n)$ be functions defined over positive integers.
- $f(n)$ is $\Omega(g(n))$, if $\exists c > 0$, and $n_0 > 1$ such that

$$f(n) \geq c.g(n)$$

for all values of $n \geq n_0$.

Big Omega

Theorem

Prove $f(n) = n^3 + 20n$ is $\Omega(n^2)$

Proof

- Find $c > 0$, and $n_0 > 0$ such that $n^3 + 20n \geq c.n^2$
- Or, $c \leq n + \frac{20}{n}$.
- RHS of above expression is minimum, when $n = \sqrt{20}$
- So, with $n_0 = 5$ and $c \leq 9$ $f(n) \geq c.n^2$ for $n \geq n_0$.
- Note this is same as saying n^2 is O($n^3 + 20n$).

Big Omega

Theorem

Prove $f(n) = n^3 + 20n$ is $\Omega(n^3)$

Proof

- Find $c > 0$, and $n_0 > 0$ such that $n^3 + 20n \geq c.n^3$
- I.e., $c \leq 1 + \frac{20}{n^2}$,
- Let $c = 1$ and $n_0 = 1$, then $f(n) \geq c.n^3$ for $n \geq n_0$.

Big Omega

Theorem

Prove that $f(n)$ is $\Omega(g(n))$ iff $g(n) = O(f(n))$.

Proof

- If $f(n) = \Omega(g(n))$ then $\exists c > 0$ and $n_0 \geq 1$ such that $f(n) \geq c.g(n)$.
- It implies $g(n) \leq \frac{1}{c} f(n)$.
- Let $\frac{1}{c} = c_1$. Since $c > 0$, $c_1 > 0$.
- So, we have $g(n) \leq c_1 f(n)$ for a $c_1 > 0$, and $n > n_0 \geq 1$,
- Converse part can be proved likewise.

Big Omega

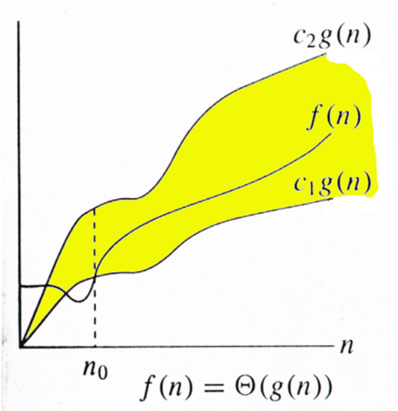
Example

Prove that $n^2 - 2n + 1$ is $\Omega(n^2)$

Proof

- Eliminate lowest order term $1 > 0$, $f(n) > n^2 - 2n$
- If $n > 10$, then $-10 > -n$, implies $-2 > 0.2n$
- Now $-2 > -0.2n$ implies $-2n > -0.2n^2$
- So, $n^2 - 2n > n^2 - 0.2n^2 = 0.8n^2$
- Furthermore, $n > 10$ implies $.8n^2 > n^2/2$
- Therefore, $n^2 - 2n + 1 > n^2/2$ for $n > n_0 = 10$.

Big Theta



Definition Of Big Theta

- Let $f(n)$ and $g(n)$ be functions defined over positive integers.
- $f(n)$ is $\Theta(g(n))$, if $\exists c_1 > 0$, $c_2 > 0$ and $n_0 > 1$ such that

$$c_1.g(n) \leq f(n) \geq c_2.g(n)$$

for all values of $n \geq n_0$.

Big Theta

Example

Show that $f(n) = 3n^2 + 8n \log n$ is $\Theta(n^2)$.

Proof

- For $n > 1$, since $0 \leq 8n \log n \leq 8n^2$, we have $3n^2 + 8n \log n \leq 11n^2$
- Also n^2 is O($3n^2 + 8n \log n$).
- Hence, $3n^2 + 8n \log n = \Theta(n^2)$.

Use of Limits

- A quick way to determine if $f(n)$ is O($g(n)$) is to find if

$$\lim_{n\to\infty} \frac{|f(n)|}{|g(n)|}$$

exists and finite.

- Similarly, if the above limit is not equal to zero. then $f(n)$ is $\Theta(g(n))$.
- If above limit is some constant c, where $0 < c < \infty$ then $f(n)$ is $\Omega(g(n))$.

Little oh and Little omega

- There are two other asymptotic bounds called little ω and little o.
- These bounds are loose bounds.
- If $\lim_{n\to\infty} \frac{|f(n)|}{|g(n)|} = 0$ then $f(n)$ is $o(g(n))$
- If $\lim_{n\to\infty} \frac{|f(n)|}{|g(n)|} = \infty$ then $f(n)$ is $\omega(g(n))$

Use of Limits

Exercise

Prove $f(n) = 7n + 8$, is $o(n^2)$.

Proof

$$\lim_{n\to\infty} \frac{7n+8}{n^2} = \lim_{n\to\infty} \frac{7}{n}, \text{ by l'Hospital}$$
$$= 0$$

Upper & Lower Bounds

- Upper and lower bounds give only incomplete information.
- Bounds are important when we have incomplete knowledge of execution time.
- Upper (or lower) bound is not the same as the worst (the best) case input size.
- The best and the worst cases are not tied to input sizes.
  - They express the distribution of the input elements, so that for a given size what would be the maximum (or the minimum) execution time.

Upper & Lower Bounds

Upper bound

Lowered by better algorithm

UB implies we know the best algorithmic solution.

LB represent the best theoretical solution

Lower bound

Raised by better proof

Upper & Lower Bounds

Upper bound **Closed problems have identical bounds**

Lower bound

Upper bound

LB & UB differ: Unknown space

Lower bound

- For closed problems, better algorithms are possible: it does not change big-Oh but reduces hidden constant.

Tractable and Intractable Problems

	Problems	Algorithms
Polynomial	Tractable	Reasonable
Exponential	Intractable	Unreasonable

Definition (Tractable)

If upper and lower bounds have only polynomial factors.

Definition (Intractable)

If both upper and lower bounds have an exponential factor.

Assignment on Running Time

It will be a theoretical assignment which will be posted soon.
Due date for the assignment will be as indicated in the sheet.

Summary

Computational Concepts

- Introduced theoretical models of computation: TM and RAM
- Notion of running time
- Big Oh, Big Omega, Big Theta, little oh and little omega.
- Some worked out examples.
- Upper bound and lower bounds.