1 More Asymptotic notations

And you thought Big-Oh was enough trouble...

1.1 Big-Omega notation

Now suppose you wanted to say statements of the form “the running time of the algorithm is at least \(O(n^2)\).” Can we say \(f(n) \geq O(g(n))\)? No! This statement is meaningless. To do this, we use a different symbol called “big-Omega”.

**Definition 1.** We say \(g(x) = \Omega(f(x))\) if \(\exists x_0, c > 0\) such that for all \(x \geq x_0\), we have \(g(x) \geq c|f(x)|\).

Big Omega is the opposite of big-Oh. More precisely,

**Theorem 1.** \(f(x) = O(g(x))\) if and only if \(g(x) = \Omega(f(x))\).

**Proof.** We have \(f(x) = O(g(x))\) iff \(\exists x_0, c > 0\) such that \(\forall x \geq x_0, |f(x)| \leq cg(x)\). This holds iff \(\exists x_0, c' > 0\) such that \(\forall x \geq x_0, g(x) \geq c'|f(x)|\), where \(c' = \frac{1}{c}\). Finally, this holds iff \(g(x) = \Omega(f(x))\). \(\square\)

For example, \(x^2 = \Omega(x), 2^x = \Omega(x^2)\), and \(\frac{x}{100} = \Omega(100x + \sqrt{x})\). So if the running time of your algorithm on inputs of size \(n\) is \(T(n)\), and you want to say it is at least quadratic, say \(T(n) = \Omega(n^2)\).

1.2 Theta notation

Sometimes, we want to specify a running time precisely up to constant factors. For example, the running time might be at least quadratic and at most quadratic. For this, we use big-Theta notation.

**Definition 2.** \(f(x) = \Theta(g(x))\) iff \(f(x) = O(g(x))\) and \(f(x) = \Omega(g(x))\). That is to say, iff there exist \(x_0, c_0, c_1 > 0\) such that \(\forall x \geq x_0,\)

\[
c_0|g(x_0)| \leq |f(x)| \leq c_1g(x_0).
\]
For example, \(10x^3 - 20x^2 + 1 = \Theta(x^3)\). Also, \(\frac{x}{\log x} \neq \Theta(x)\). So keep in mind that if your running time \(T(n) = \Theta(n^2)\), then it grows quadratically in \(n\), that is, it is both upper and lower bounded by a quadratic function.

### 1.3 Some others...

Besides \(O\), \(\Omega\), and \(\Theta\), there are two other Greek symbols we use: \(o\) and \(\omega\). Intuitively, \(O\) corresponds to \(\leq\), \(\Omega\) to \(\geq\), and \(\Theta\) to \(=\). Here, \(o\) intuitively corresponds to \(<\), and \(\omega\) to \(>\).

**Definition 3.** \(f(x) = o(g(x))\) if \(\forall c > 0, \exists x_0\) such that \(\forall x \geq x_0, |f(x)| \leq cg(x)\).

The difference here with big-Oh is that here \(g\) grows strictly faster than \(f\) since we quantify over all constants \(c > 0\), whereas for big-Oh, we just require this hold for some constant \(c\), and so in big-Oh \(g\) only grows at least as fast as \(f\).

For example, \(\frac{x}{\log x} = o(x)\). Indeed, for all \(c > 0\), there is an \(x_0\) such that for all \(x \geq x_0\), \(\frac{x}{\log x} \leq cx\). Define \(x_0 = 2^{1/c}\). Then \(\frac{x}{\log x} \leq \frac{x}{\log 2^{1/c}} = cx\).

On the other hand, \(\frac{x}{100} \neq o(x)\). We prove this by contradiction. If \(\frac{x}{100} = o(x)\), then \(\forall c \exists x_0\) such that \(\forall x \geq x_0\), \(\frac{x}{100} \leq cx\).

But if \(c = \frac{1}{200}\), this means \(\forall x \geq x_0, \frac{x}{100} < \frac{x}{200}\), which is impossible.

Finally, we define \(\omega\) as follows.

**Definition 4.** \(g(x) = \omega(f(x))\) if and only if \(f(x) = o(g(x))\).

The \(\omega\) symbol is not used as commonly as the other 4 symbols. As with the \(\sim\) notation, sometimes you’ll see an expression like

\[H_n = \ln n + \gamma + O\left(\frac{1}{n}\right),\]

which means

\[H_n - \ln n - \gamma = O\left(\frac{1}{n}\right)\]

As \(n\) gets large, the error term is at most a constant times \(1/n\).
Problem 1. Which of these symbols

\[ \Theta \quad O \quad \Omega \quad o \quad \omega \]

can go in these boxes?

\[
2n + \log n = \boxed{\Theta, O, \Omega} (n)
\]

\[
\log n = \boxed{O, o} (n)
\]

\[
\sqrt{n} = \boxed{\Omega, \omega} (\log^{300} n)
\]

\[
n2^n = \boxed{\Omega, \omega} (n)
\]

\[
n^7 = \boxed{O, o} (1.01^n)
\]
Problem 2. An explorer is trying to reach the Holy Grail, which she believes is located in a desert shrine $d$ days walk from the nearest oasis.\footnote{She’s right about the location, but doesn’t realize that the Holy Grail is actually just the Beneš network.} In the desert heat, the explorer must drink continuously. She can carry at most 1 gallon of water, which is enough for 1 day. However, she is free to create water caches out in the desert.

For example, if the shrine were $2/3$ of a day’s walk into the desert, then she could recover the Holy Grail with the following strategy. She leaves the oasis with 1 gallon of water, travels $1/3$ day into the desert, caches $1/3$ gallon, and then walks back to the oasis—arriving just as her water supply runs out. Then she picks up another gallon of water at the oasis, walks $1/3$ day into the desert, tops off her water supply by taking the $1/3$ gallon in her cache, walks the remaining $1/3$ day to the shrine, grabs the Holy Grail, and then walks for $2/3$ of a day back to the oasis—again arriving with no water to spare.

But what if the shrine were located farther away?

(a) What is the most distant point that the explorer can reach and return from if she takes only 1 gallon from the oasis?

Solution. At best she can walk $1/2$ day into the desert and then walk back.

(b) What is the most distant point the explorer can reach and return form if she takes only 2 gallons from the oasis? No proof is required; just do the best you can.

Solution. The explorer walks $1/4$ day into the desert, drops $1/2$ gallon, then walks home. Next, she walks $1/4$ day into the desert, picks up $1/4$ gallon from her cache, walks an additional $1/2$ day out and back, then picks up another $1/4$ gallon from her cache and walks home. Thus, her maximum distance from the oasis is $3/4$ of a day’s walk.

(c) What about 3 gallons? (Hint: First, try to establish a cache of 2 gallons plus enough water for the walk home as far into the desert as possible. Then use this cache as a springboard for your solution to the previous part.)

Solution. Suppose the explorer makes three trips $1/6$ day into the desert, dropping $2/3$ gallon off units each time. On the third trip, the cache has 2 gallons of water, and the explorer still has $1/6$ gallon for the trip back home. So, instead of returning immediately, she uses the solution described above to advance another $3/4$ day into the desert and then returns home. Thus, she reaches

$$\frac{1}{6} + \frac{1}{4} + \frac{1}{2} = \frac{11}{12}$$

days’ walk into the desert.

(d) How can the explorer go as far as possible is she withdraws $n$ gallons of water? Express your answer in terms of the Harmonic number $H_n$, defined by:

$$H_n = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}$$
Solution. With \( n \) gallons of water, the explorer can reach a point \( H_n/2 \) days into the desert.

Suppose she makes \( n \) trips \( 1/(2n) \) days into the desert, dropping of \((n-1)/n\) gallons each time. Before she leaves the cache for the last time, she has \( n - 1 \) gallons plus enough for the walk home. So she applies her \((n-1)\)-day strategy to go an additional \( H_{n-1}/2 \) days into the desert and then returns home. Her maximum distance from the oasis is then:

\[
\frac{1}{2n} + \frac{H_{n-1}}{2} = \frac{H_n}{2}
\]

(e) Use the fact that 

\[
H_n \sim \ln n
\]

to approximate your previous answer in terms of logarithms.

Solution. An approximate answer is \( \ln n/2 \).

(f) Suppose that the shrine is \( d = 10 \) days walk into the desert. Relying on your approximate answer, how many days must the explorer travel to recover the Holy Grail?

Solution. She can obtains the Grail when:

\[
\frac{H_n}{2} \approx \frac{\ln n}{2} \geq 10
\]

This requires about \( n \geq e^{20} = 4.8 \cdot 10^8 \) days.