

Lecture 2: Common Inequalities and Proof Strategies

January 28, 2026

Lecturer: D Ellis Hershkowitz

Scribe: Emin Alp Guneri

1 4 Series and 3 Inequalities

1.1 4 Series

The following series are highly relevant for analyzing properties of algorithms. These series appear commonly while analyzing the run-times of numerous algorithms, and will help us derive upper/lower bounds for the run-time complexities of these algorithms.

1.1.1 Triangular Number

The n th triangle number is defined as:

$$T_n = \sum_{i=1}^n i$$

We would like to show that:

$$\sum_{i=1}^n i = \Theta(n^2)$$

We can show this as follows:

$$\sum_{i=1}^n i = 1 + 2 + 3 + \cdots + n - 2 + n - 1 + n$$

$$\sum_{i=1}^n i = 1 + 2 + 3 + \cdots + n - 2 + n - 1 + n$$

We notice that by grouping terms on the left and right together, starting from the outermost elements, we can sum them up to $n + 1$ every time. This way we can get $n/2$ of these $n + 1$ terms.

$$\begin{aligned} \sum_{i=1}^n i &= (n+1) + (n+1) + \cdots + (n+1) \\ \sum_{i=1}^n i &= \sum_{n/2}^n (n+1) \\ \sum_{i=1}^n i &= \frac{n}{2}(n+1) = \frac{n(n+1)}{2} \end{aligned}$$

We also know that $n(n+1)/2 = \Theta(n^2)$, therefore we conclude:

$$\sum_{i=1}^n i = \frac{n}{2}(n+1) = \Theta(n^2)$$

1.1.2 Harmonic Number

The n th harmonic number is defined as:

$$H_n = \sum_{i=1}^n \frac{1}{i}$$

We would like to show that:

$$\sum_{i=1}^n \frac{1}{i} = \Theta(\log n)$$

We can show this by grouping and bounding the terms of the summation as follows:

$$\begin{aligned} H_n &= 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \cdots + \frac{1}{n} \\ H_n &= \underbrace{1}_{\leq 1} + \underbrace{\frac{1}{2} + \frac{1}{3}}_{\leq 1} + \underbrace{\frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7}}_{\leq 1} + \cdots + \frac{1}{n} \\ H_n &\leq 1 + 1 + 1 + \cdots + 1 \end{aligned}$$

Each of the groups are ≤ 1 , which can be shown with the following reasoning:

$$\begin{aligned}
& \frac{1}{k} + \frac{1}{k+1} + \frac{1}{k+2} + \cdots + \frac{1}{2k-1} \\
& \underbrace{\hspace{10em}} \\
& \leq \frac{1}{k} + \frac{1}{k} + \frac{1}{k} + \cdots + \frac{1}{k} \\
& \leq \frac{1}{k} \cdot k \\
& \leq 1
\end{aligned}$$

We notice that the group size (the number of terms in a group) doubles every time we bound a group by ≤ 1 . First group was 1 and had a size of 1. Second group was $\frac{1}{2} + \frac{1}{3}$ and had a group size of 2, and so on. Because we have n terms in total, the sizes of these groups must add up to n . Let x denote the number of groups we have. Then the last group must have 2^{x-1} terms, and we have:

$$\begin{aligned}
1 + 2 + 4 + 8 + \cdots + 2^{x-1} &= n \\
2^x - 1 &= n \\
2^x &= n + 1 \\
x &= \log(n + 1) \\
x &= \Theta(\log n)
\end{aligned}$$

Therefore, we have $\Theta(\log n)$ groups, each ≤ 1 . Based on this:

$$\begin{aligned}
H_n &\leq \sum_{\Theta(\log n)} 1 \\
H_n &\leq \Theta(\log n) \\
H_n &= O(\log n)
\end{aligned}$$

With this we have upper-bounded the sum. To prove that H_n is $\Theta(\log n)$, we need to lower-bound the sum as well. We use the same grouping logic, but we lower-bound each term with $\frac{1}{2}$.

$$\begin{aligned}
H_n &= \underbrace{\frac{1}{1}}_{\geq \frac{1}{2}} + \underbrace{\frac{1}{2} + \frac{1}{3}}_{\geq \frac{1}{2}} + \underbrace{\frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7}}_{\geq \frac{1}{2}} + \cdots + \frac{1}{n} \\
H_n &\geq \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \cdots + \frac{1}{2}
\end{aligned}$$

Once again, there are $\Theta(\log n)$ groups each $\geq \frac{1}{2}$.

$$\begin{aligned}
H_n &\geq \sum_{\Theta(\log n)} \frac{1}{2} \\
H_n &\geq \frac{\Theta(\log n)}{2} \\
H_n &= \Omega(\log n)
\end{aligned}$$

And because H_n is $O(\log n)$ and $\Omega(\log n)$, we conclude that H_n is $\Theta(\log n)$.

1.1.3 Inverse squares

We would like to show that:

$$\sum_{i=1}^n \frac{1}{i^2} = \theta(1)$$

We establish two facts before proving this statement:

- **First fact:**

$$\begin{aligned}
\frac{1}{i(i-1)} &= \frac{1-i+i}{i(i-1)} \\
\frac{1}{i(i-1)} &= \frac{i-(i-1)}{i(i-1)} \\
\frac{1}{i(i-1)} &= \frac{1}{i-1} - \frac{1}{i}
\end{aligned}$$

- **Second fact:**

$$\begin{aligned}
\sum_{i=2}^n \frac{1}{i-1} - \frac{1}{i} &= \left(\frac{1}{1} - \frac{1}{2}\right) + \left(\frac{1}{2} - \frac{1}{3}\right) + \cdots + \left(\frac{1}{n-1} - \frac{1}{n}\right) \\
\sum_{i=2}^n \frac{1}{i-1} - \frac{1}{i} &= \left(\frac{1}{1} - \frac{1}{2}\right) + \left(\frac{1}{2} - \frac{1}{3}\right) + \cdots + \left(\frac{1}{n-1} - \frac{1}{n}\right) \\
\sum_{i=2}^n \frac{1}{i-1} - \frac{1}{i} &= \left(\frac{1}{1} - \frac{1}{2}\right) + \left(\frac{1}{2} - \frac{1}{3}\right) + \cdots + \left(\frac{1}{n-1} - \frac{1}{n}\right) \\
\sum_{i=2}^n \frac{1}{i-1} - \frac{1}{i} &= 1 - \frac{1}{n}
\end{aligned}$$

Using these facts, we can prove our original statement. We begin by upper-bounding the sum:

$$\begin{aligned}
\sum_{i=1}^n \frac{1}{i^2} &= 1 + \sum_{i=2}^n \frac{1}{i^2} \\
\sum_{i=1}^n \frac{1}{i^2} &\leq 1 + \sum_{i=2}^n \frac{1}{i(i-1)} \\
\sum_{i=1}^n \frac{1}{i^2} &\leq 1 + \left(1 - \frac{1}{n}\right) \\
\sum_{i=1}^n \frac{1}{i^2} &\leq 2 - \frac{1}{n} \\
\sum_{i=1}^n \frac{1}{i^2} &= O(1)
\end{aligned}$$

We can also lower-bound the sum by simply multiplying the right hand-side with $\frac{1}{2}$.

$$\begin{aligned}
\sum_{i=1}^n \frac{1}{i^2} &\geq \frac{1}{2} + \sum_{i=2}^n \frac{1}{i \cdot 2(i-1)} \\
\sum_{i=1}^n \frac{1}{i^2} &\geq \frac{1}{2} + \frac{1}{2} \sum_{i=2}^n \frac{1}{i(i-1)} \\
\sum_{i=1}^n \frac{1}{i^2} &\geq \frac{1}{2} + \frac{1}{2} \left(1 - \frac{1}{n}\right) \\
\sum_{i=1}^n \frac{1}{i^2} &\geq \frac{1}{2} + \frac{1}{2} - \frac{1}{2n} \\
\sum_{i=1}^n \frac{1}{i^2} &\geq 1 - \frac{1}{2n} \\
\sum_{i=1}^n \frac{1}{i^2} &= \Omega(1)
\end{aligned}$$

And because this sum is both $O(1)$ and $\Omega(1)$ we conclude that it is $\Theta(1)$.

1.1.4 Geometric series

We would like to show that:

$$\sum_{i=0}^{\infty} r^i = O(1) \text{ for } r \in (0, 1)$$

We show this fact by letting S denote the value of this sum, and performing the following steps:

$$\begin{aligned}
 S &= \sum_{i=0}^{\infty} r^i \\
 S &= 1 + r + r^2 + r^3 + \dots \\
 S &= 1 + r(1 + r + r^2 + \dots) \\
 S &= 1 + r \cdot S \\
 (1 - r)S &= 1 \\
 S &= \frac{1}{1 - r}
 \end{aligned}$$

We conclude that $S = \Theta(1)$ because the last expression we get is a constant w.r.t n .

1.2 Big 3 Inequalities

1. AM-GM Inequality

Let $\forall i(x_i \in \mathbb{R}^+)$. Then, the AM-GM inequality tells us:

$$\left(\prod_{i=1}^k x_i \right)^{\frac{1}{k}} \leq \frac{\sum_{i=1}^k x_i}{k}$$

The term on the left hand-side is called the arithmetic mean, whereas the term on the right hand-side is called the geometric mean. AM-GM (Arithmetic Mean - Geometric Mean) inequality tells us that the arithmetic mean is never less than the geometric mean.

2. Cauchy-Schwarz Inequality

Cauchy-Schwarz inequality applies to all n -dimensional real vectors u and v .

$$\langle u, v \rangle \leq \sqrt{\langle u, u \rangle \cdot \langle v, v \rangle}$$

or when written out with sums,

$$\sum_{i=1}^k u_i v_i \leq \sqrt{\sum_{i=1}^k u_i^2 \cdot \sum_{i=1}^k v_i^2}$$

3. Jensen's Inequality

Let $p_1, p_2, \dots, p_k \in [0, 1]$ such that $\sum_i p_i = 1$, Let $a_1, a_2, \dots, a_k \in \mathbb{R}^+$, and let $f : \mathbb{R} \rightarrow \mathbb{R}$ be any convex function.

$$f\left(\sum_{i=1}^k p_i a_i\right) \leq \sum_{i=1}^k p_i f(a_i)$$

To build some intuition about these inequalities we start with two "obvious" facts.

Fact 1: Square Inequality

Given a rectangle with side lengths x_1 and x_2 , we can squish this rectangle into a square with side lengths $(x_1 + x_2)/2$. This square will always have a larger area than the rectangle, and thus we say that a square maximizes area (while preserving total perimeter).

We prove this statement as follows:

$$\begin{aligned} \text{Area(Rectangle)} &\leq \text{Area(Square)} \\ x_1 x_2 &\leq \left(\frac{x_1 + x_2}{2}\right)^2 & (1) \\ 4x_1 x_2 &\leq (x_1 + x_2)^2 \\ 4x_1 x_2 &\leq x_1^2 + 2x_1 x_2 + x_2^2 \\ 0 &\leq x_1^2 - 2x_1 x_2 + x_2^2 \\ 0 &\leq (x_1 - x_2)^2 \end{aligned}$$

which follows from the square of a real number always being greater than or equal to 0. We note that Line (1) matches the form of our stated AM-GM inequality with some manipulation.

$$\begin{aligned} x_1 x_2 &\leq \left(\frac{x_1 + x_2}{2}\right)^2 \\ \sqrt{x_1 x_2} &\leq \left(\frac{x_1 + x_2}{2}\right) \\ \left(\prod_{i=1}^2 x_i\right)^{\frac{1}{2}} &\leq \frac{\sum_{i=1}^2 x_i}{2} \end{aligned}$$

Fact 2: Triangle Inequality

Assume a triangle that has the vertices $u, v, w \in \mathbb{R}^2$, and let Euclidean distance be our notion of distance between two points.

$$d(x, y) := \sqrt{\sum_i (x_i - y_i)^2}$$

We note that Euclidean distance is translation invariant, meaning:

$$d(x + \delta, y + \delta) = d(x, y) \quad \forall x, y, \delta \in \mathbb{R}^2$$

which follows from the difference $(x_i - y_i)$ remaining the same for every term after translation.

From triangle inequality we know that the distance between u and v is always less than or equal to distance between v and w summed with the distance between w and u . This could also be understood as the shortest path between two points being the straight line that connects them. Therefore from this fact, we have:

$$d(u, v) \leq d(u, w) + d(w, v) \quad \forall u, v, w \in \mathbb{R}^2 \quad (2)$$

$$d(0, v - u) \leq d(0, w - u) + d(0, v - w) \quad (3)$$

where we perform translation on each distance term to go from Line (2) to Line (3). We translate the distance terms by $-u$, $-u$, and $-v$ respectively. We now let $\tilde{u} = w - u$ and $\tilde{v} = v - w$. We note that $\tilde{u} + \tilde{v} = v - u$, and substitute the vectors with their tilde variants.

$$\begin{aligned} d(0, v - u) &\leq d(0, w - u) + d(0, v - w) \\ d(0, \tilde{u} + \tilde{v}) &\leq d(0, \tilde{u}) + d(0, \tilde{v}) \end{aligned}$$

We note that $\tilde{u} \in \mathbb{R}^2$. Since for any value of \tilde{u} there exists $u, w \in \mathbb{R}^2$ such that $\tilde{u} = w - u$, and that the inequality above should hold for all $u, w \in \mathbb{R}^2$, the inequality should also hold for all $\tilde{u} \in \mathbb{R}^2$. Same logic applies to \tilde{v} , meaning:

$$d(0, \tilde{u} + \tilde{v}) \leq d(0, \tilde{u}) + d(0, \tilde{v}) \quad \forall \tilde{u}, \tilde{v} \in \mathbb{R}^2$$

We can remove the tildes now, since \tilde{u} and \tilde{v} represent any vector in \mathbb{R}^2 .

$$\begin{aligned} d(0, u + v) &\leq d(0, u) + d(0, v) \quad \forall u, v \in \mathbb{R}^2 \\ \sqrt{\sum_i (u_i + v_i - 0)^2} &\leq \sqrt{\sum_i (u_i - 0)^2} + \sqrt{\sum_i (v_i - 0)^2} \\ \sqrt{\sum_i (u_i + v_i)^2} &\leq \sqrt{\sum_i u_i^2} + \sqrt{\sum_i v_i^2} \\ \sqrt{(u_1 + v_1)^2 + (u_2 + v_2)^2} &\leq \sqrt{u_1^2 + u_2^2} + \sqrt{v_1^2 + v_2^2} \\ (u_1 + v_1)^2 + (u_2 + v_2)^2 &\leq (u_1^2 + u_2^2) + 2\sqrt{(u_1^2 + u_2^2)(v_1^2 + v_2^2)} + (v_1^2 + v_2^2) \\ u_1^2 + 2u_1v_1 + v_1^2 + u_2^2 + 2u_2v_2 + v_2^2 &\leq u_1^2 + u_2^2 + 2\sqrt{(u_1^2 + u_2^2)(v_1^2 + v_2^2)} + v_1^2 + v_2^2 \end{aligned}$$

$$\begin{aligned}
2u_1v_1 + 2u_2v_2 &\leq 2\sqrt{(u_1^2 + u_2^2)(v_1^2 + v_2^2)} \\
u_1v_1 + u_2v_2 &\leq \sqrt{(u_1^2 + u_2^2)(v_1^2 + v_2^2)} \\
(u_1v_1 + u_2v_2)^2 &\leq (u_1^2 + u_2^2)(v_1^2 + v_2^2) \\
u_1^2v_1^2 + 2u_1v_1u_2v_2 + u_2^2v_2^2 &\leq u_1^2v_1^2 + u_2^2v_1^2 + u_1^2v_2^2 + u_2^2v_2^2 \\
2u_1v_1u_2v_2 &\leq u_2^2v_1^2 + u_1^2v_2^2 \\
0 &\leq u_2^2v_1^2 - 2u_1v_1u_2v_2 + u_1^2v_2^2 \\
0 &\leq (u_2v_1 - u_1v_2)^2
\end{aligned} \tag{4}$$

which once again follows from the square of a real number always being greater than or equal to 0. We note that Line (4) matches the form of our stated Cauchy-Schwarz inequality with some manipulation.

$$\begin{aligned}
u_1v_1 + u_2v_2 &\leq \sqrt{(u_1^2 + u_2^2)(v_1^2 + v_2^2)} \\
\sum_{i=1}^2 u_i v_i &\leq \sqrt{\sum_{i=1}^2 u_i^2 \sum_{i=1}^2 v_i^2} \\
\langle u, v \rangle &\leq \sqrt{\langle u, u \rangle \langle v, v \rangle}
\end{aligned}$$

Proofs of Big 3 Inequalities

We used the square inequality to verify the correctness of AM-GM inequality in two dimension, and we used the triangle inequality to verify the correctness of Cauchy-Schwarz inequality in two dimensions.

However, AM-GM and Cauchy-Schwarz inequalities are applicable not only to two dimension but to k -dimensions, where $k \in \mathbb{Z}^+$.

For our proofs, we first prove Cauchy-Schwarz with induction. We then prove Jensen's inequality, and use Jensen's inequality to prove AM-GM inequality.

1.2.1 Proof of Cauchy-Schwarz Inequality

We prove Cauchy-Schwarz Inequality using strong induction.

Base Case: $n = 2$

We have already proven that Cauchy-Schwarz inequality holds in two dimensions.

Induction Hypothesis: $n \leq k$

Assume that Cauchy-Schwarz inequality holds in $n \leq k$ dimensions. This means we assume:

$$\sum_{i=1}^n u_i v_i \leq \sqrt{\sum_{i=1}^n u_i^2 \sum_{i=1}^n v_i^2}$$

$$u_1 v_1 + \cdots + u_n v_n \leq \sqrt{\sum_{i=1}^n u_i^2 \sum_{i=1}^n v_i^2}$$

for all $n \in [k] \setminus 1$

Inductive Step: $n = k + 1$

$$\sum_{i=1}^{k+1} u_i v_i \stackrel{?}{\leq} \sqrt{\sum_{i=1}^{k+1} u_i^2 \sum_{i=1}^{k+1} v_i^2}$$

$$(u_1 v_1 + \cdots + u_k v_k) + u_{k+1} v_{k+1} \leq \left(\sqrt{\sum_{i=1}^k u_i^2 \sum_{i=1}^k v_i^2} \right) + u_{k+1} v_{k+1} \quad (5)$$

$$(u_1 v_1 + \cdots + u_k v_k) + u_{k+1} v_{k+1} \leq \sqrt{\sum_{i=1}^k u_i^2} \sqrt{\sum_{i=1}^k v_i^2} + u_{k+1} v_{k+1}$$

where Line (5) follows from our Induction Hypothesis, simply by adding $u_{k+1} v_{k+1}$ to both sides. We now define two vectors \bar{u} and \bar{v} .

$$\bar{u}_1 := \sqrt{\sum_{i=1}^k u_i^2}$$

$$\bar{v}_1 := \sqrt{\sum_{i=1}^k v_i^2}$$

$$\bar{u}_2 := u_{k+1}$$

$$\bar{v}_2 := v_{k+1}$$

$$\bar{u} := (\bar{u}_1 \ \bar{u}_2)$$

$$\bar{v} := (\bar{v}_1 \ \bar{v}_2)$$

$$\langle \bar{u}, \bar{v} \rangle = \bar{u}_1 \bar{v}_1 + \bar{u}_2 \bar{v}_2$$

$$\langle \bar{u}, \bar{v} \rangle = \sqrt{\sum_{i=1}^k u_i^2} \sqrt{\sum_{i=1}^k v_i^2} + u_{k+1} v_{k+1}$$

From the last line we recognize the significance of defining \bar{u} and \bar{v} , as it is now evident that:

$$\begin{aligned}\langle u, v \rangle &\leq \sqrt{\sum_{i=1}^k u_i^2} \sqrt{\sum_{i=1}^k v_i^2} + u_{k+1}v_{k+1} \\ \langle u, v \rangle &\leq \langle \bar{u}, \bar{v} \rangle\end{aligned}$$

Because \bar{u} and \bar{v} are two dimensional vectors, by our induction hypothesis, we know that Cauchy-Schwarz inequality also applies to these vectors.

$$\begin{aligned}\langle \bar{u}, \bar{v} \rangle &\leq \sqrt{(\bar{u}_1^2 + \bar{u}_2^2)(\bar{v}_1^2 + \bar{v}_2^2)} \\ \langle \bar{u}, \bar{v} \rangle &\leq \sqrt{\left(\sum_{i=1}^k u_i^2 + u_{k+1}^2\right)\left(\sum_{i=1}^k v_i^2 + v_{k+1}^2\right)} \\ \langle \bar{u}, \bar{v} \rangle &\leq \sqrt{\left(\sum_{i=1}^{k+1} u_i^2\right)\left(\sum_{i=1}^{k+1} v_i^2\right)} \\ \langle u, v \rangle &\leq \langle \bar{u}, \bar{v} \rangle \leq \sqrt{\left(\sum_{i=1}^{k+1} u_i^2\right)\left(\sum_{i=1}^{k+1} v_i^2\right)} \\ \langle u, v \rangle &\leq \sqrt{\left(\sum_{i=1}^{k+1} u_i^2\right)\left(\sum_{i=1}^{k+1} v_i^2\right)} \\ \langle u, v \rangle &\leq \sqrt{\langle u, u \rangle \langle v, v \rangle}\end{aligned}$$

which is what we wanted to prove. Now that we have proven that Cauchy-Schwarz inequality holds in $k + 1$ dimensions given that it holds for $\leq k$ dimensions, and we have proven that it holds for $n = 2$ dimensions, by induction we conclude that Cauchy-Schwarz inequality must hold for all dimensions, $n \geq 2$.

1.2.2 Proof of Jensen's Inequality

Firstly, we define what a convex function is. A function $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex if $\forall p \in [0, 1], \forall a_1, a_2 \in \mathbb{R}$ the following inequality holds:

$$f(pa_1 + (1-p)a_2) \leq pf(a_1) + (1-p)f(a_2)$$

Geometrically, this can be understood by choosing two arbitrary points a_1 and a_2 , connecting the points $(a_1, f(a_1))$ and $(a_2, f(a_2))$ with a line, and checking that this line never goes below f . If the line never goes below f , for any choice of a_1 and a_2 , then the function is convex. A simple example for a convex function is $f(x) = x^2$.

We prove Jensen's inequality by strong induction similar to our proof for Cauchy-Schwarz inequality. Let n denote the number of a and p terms in the inequality.

Base case: $n = 2$

Jensen's inequality for two terms is:

$$\begin{aligned} f\left(\sum_{i=1}^2 p_i a_i\right) &\leq \sum_{i=1}^2 p_i f(a_i) \\ f(p_1 a_1 + p_2 a_2) &\leq p_1 f(a_1) + p_2 f(a_2) \\ p_1 + p_2 = 1 &\implies p_2 = 1 - p_1 \\ \therefore f(p_1 a_1 + (1 - p_1) a_2) &\leq p_1 f(a_1) + (1 - p_1) f(a_2) \end{aligned}$$

Thus, the base case with two terms holds because it is equivalent to the definition of convexity of f , and Jensen's inequality requires f to be convex.

Induction Hypothesis: $n \leq k$

Assume that Jensen's inequality is true for all $n \leq k$.

$$f\left(\sum_{i=1}^n p_i a_i\right) \leq \sum_{i=1}^n p_i f(a_i)$$

Inductive Step: $n = k + 1$

Now we prove that Jensen's inequality holds for $n = k + 1$ terms. Let $p = \sum_{i=1}^k p_i$. We also know that

$\sum_{i=1}^{k+1} p_i = 1$, therefore $p_{k+1} = 1 - p$. We use these terms to break Jensen's inequality into smaller pieces.

$$\begin{aligned} f\left(\sum_{i=1}^{k+1} p_i a_i\right) &= f(p_1 a_1 + \cdots + p_k a_k + p_{k+1} a_{k+1}) \\ f\left(\sum_{i=1}^{k+1} p_i a_i\right) &= f\left(p \left[\frac{p_1}{p} a_1 + \frac{p_2}{p} a_2 + \cdots + \frac{p_k}{p} a_k\right] + p_{k+1} a_{k+1}\right) \\ f\left(\sum_{i=1}^{k+1} p_i a_i\right) &= f(p \tilde{a} + (1 - p) a_{k+1}), \text{ where } \tilde{a} := \left[\frac{p_1}{p} a_1 + \frac{p_2}{p} a_2 + \cdots + \frac{p_k}{p} a_k\right] \\ f\left(\sum_{i=1}^{k+1} p_i a_i\right) &\leq p f(\tilde{a}) + (1 - p) f(a_{k+1}) \end{aligned} \tag{6}$$

where Line (6) follows from the correctness of our base case with $n = 2$. We now focus on bounding $f(\tilde{a})$. We note that Jensen's inequality also applies to $f(\tilde{a})$, because f is convex, the constants used a_1, a_2, \dots, a_k are real valued constants, and the p coefficients add up to 1: $\sum_i \frac{p_i}{p} = \frac{1}{p} \sum_i p_i = \frac{1}{p} \cdot p = 1$.

$$\begin{aligned}
f(\tilde{a}) &= f\left(\frac{p_1}{p}a_1 + \frac{p_2}{p}a_2 + \cdots + \frac{p_k}{p}a_k\right) \\
f(\tilde{a}) &\leq \frac{p_1}{p}f(a_1) + \frac{p_2}{p}f(a_2) + \cdots + \frac{p_k}{p}f(a_k) \\
pf(\tilde{a}) &\leq p_1f(a_1) + p_2f(a_2) + \cdots + p_kf(a_k) \\
pf(\tilde{a}) + (1-p)f(a_{k+1}) &\leq p_1f(a_1) + p_2f(a_2) + \cdots + p_kf(a_k) + p_{k+1}f(a_{k+1}) \\
pf(\tilde{a}) + (1-p)f(a_{k+1}) &\leq \sum_{i=1}^{k+1} p_i f(a_i)
\end{aligned} \tag{7}$$

where Line (7) once again follows from our Induction Hypothesis, with $n = k$ terms. We substitute this back into our original inequality.

$$\begin{aligned}
f\left(\sum_{i=1}^{k+1} p_i a_i\right) &\leq pf(\tilde{a}) + (1-p)f(a_{k+1}) \leq \sum_{i=1}^{k+1} p_i f(a_i) \\
f\left(\sum_{i=1}^{k+1} p_i a_i\right) &\leq \sum_{i=1}^{k+1} p_i f(a_i)
\end{aligned}$$

which is what we wanted to prove. Now that we have proven that Jensen's inequality holds with $k + 1$ terms given that it holds with any $n \leq k$ terms, and we have proven that it holds for $n = 2$ terms, by induction we conclude that Jensen's inequality must hold for all $n \geq 2$.

1.2.3 Proof of AM-GM Inequality

We use Jensen's inequality to prove AM-GM. To this end, we:

1. Let $a_i = \ln x_i$
2. Let $f = e^x$ be our convex function of choice
3. Let $p_i = \frac{1}{k}$. Since there are k total terms, $\sum_{i=1}^k p_i = \sum_{i=1}^k \frac{1}{k} = k \cdot \frac{1}{k} = 1$, as required by Jensen's inequality.

Then, by Jensen's inequality we have:

$$\begin{aligned}
f\left(\sum_{i=1}^k p_i a_i\right) &\leq \sum_{i=1}^k p_i f(a_i) \\
\exp\left(\sum_{i=1}^k p_i a_i\right) &\leq \sum_{i=1}^k p_i \exp(a_i) \\
\exp\left(\sum_{i=1}^k \frac{1}{k} \ln(x_i)\right) &\leq \sum_{i=1}^k \frac{1}{k} \exp(\ln(x_i)) \\
\exp\left(\sum_{i=1}^k \frac{1}{k} \ln(x_i)\right) &\leq \sum_{i=1}^k \frac{1}{k} x_i \\
\exp\left(\sum_{i=1}^k \frac{1}{k} \ln(x_i)\right) &\leq \frac{\sum_{i=1}^k x_i}{k} \\
\exp\left(\frac{1}{k} \cdot \sum_{i=1}^k \ln(x_i)\right) &\leq \frac{\sum_{i=1}^k x_i}{k} \\
\exp\left(\sum_{i=1}^k \ln(x_i)\right)^{1/k} &\leq \frac{\sum_{i=1}^k x_i}{k} \\
\left(\prod_{i=1}^k \exp(\ln(x_i))\right)^{1/k} &\leq \frac{\sum_{i=1}^k x_i}{k} \\
\left(\prod_{i=1}^k x_i\right)^{1/k} &\leq \frac{\sum_{i=1}^k x_i}{k}
\end{aligned}$$

which is what we wanted to prove, showing that the geometric mean is never greater than the arithmetic mean.

2 Proof Strategies

We can prove run-time complexities of algorithms using various strategies. The following strategies are common and powerful tools for establishing upper bounds for run-time complexities.

2.1 Problem 1: Ant on a Log

Consider the following problem. There is an ant on a log, starting at point 0. At each step, the ant is able to move left or right by 1 unit, which corresponds to moving from point p to the point $p - 1$

or $p + 1$ respectively. The ant is looking for a cookie that is located either at point x or point $-x$ (unknown to the ant). How can the ant get to the cookie while minimizing the distance it travels (the number of steps it takes)?

Case 1: x is known

If the ant knows x , then they can choose to simply travel to x , and if the cookie wasn't there, it travels to $-x$. If the cookie is at x , ant takes x steps, and if the cookie is at $-x$, the ant takes $3x$ steps. In either case, the distance traveled by the ant is $\leq 3x$.

Case 2: x is not known

When x is not known, getting to the cookie is more tricky. To analyze this case, we use two proof strategies, guessing and charging.

2.1.1 Strategy 1: Guessing

Algorithmic problems are easier when parameters are known. If the parameters are unknown, we can simply guess them.

With this approach, we can solve the problem (naively) by brute-force guessing the value of x . We start our guess, \hat{x} , from 1 and increment it by one every time we fail. This gives us the following algorithm:

Algorithm 1 Naive Guessing Algorithm

```

while Cookie not found do
  for  $\hat{x} = 1, 2, 3, \dots$  do
    Go to  $\hat{x}$ 
    Go to 0
    Go to  $-\hat{x}$ 
    Go to 0
  end for
end while

```

At each guess, \hat{x} , the ant travels to \hat{x} , then to 0, then to $-\hat{x}$, and then to 0 again. Each of these travels take \hat{x} steps, and therefore for each guess, the ant takes $4\hat{x}$ steps, until \hat{x} reaches x . When \hat{x} reaches x , the ant will either take x steps or $3x$ steps, if the cookie is located at x or $-x$ respectively. In either case, we can upper-bound the number the number of steps in this iteration with $4x$ as well. Therefore the total number of steps will be upper-bounded by:

$$\begin{aligned}
 \# \text{ steps} &\leq 4 \cdot 1 + 4 \cdot 2 + \dots + 4 \cdot x \\
 \# \text{ steps} &\leq 4 \sum_{i=1}^x i \\
 \# \text{ steps} &= O(x^2) \tag{8}
 \end{aligned}$$

where Line (8) follows from the first series we analyzed, the n th triangular number.

2.1.2 Strategy 2: Charging

To upper-bound a quantity y with z , we can create z dollars available for our algorithm to use. If we pay for each part of y with the available z dollars, that proves $y \leq z$.

For our ant on a log problem, we deposit $2x$ dollars at each point between $-x$ and x inclusive, meaning we create $4x^2 + 2x$ dollars in total. Each time that the ant takes a step from point i to point $i \pm 1$, we reduce the number of dollars at point i by 1 (and we fail if the number of dollars at i was 0).

With this charging scheme, the ant will be able to reach the cookie. This is because our algorithm will run for $\leq x$ iterations and at each iteration, every point is visited at most 2 times. Therefore, at each iteration the number of dollars at any point is decreased by at most 2. Due to this, no point could have more than $2x$ dollars removed, and therefore we will be able to pay for each step with the $4x^2 + 2x$ dollars we initially created. Therefore we conclude:

$$\begin{aligned} \# \text{ steps} &\leq \text{total dollars} = 4x^2 + 2x \\ \# \text{ steps} &= O(x^2) \end{aligned}$$

We note that we can improve the efficiency of our algorithm by improving our guessing. Instead of incrementing our guesses by 1, we can double our guesses, and therefore within $1 + \log x$ guesses, \hat{x} will be greater than or equal to x .

Algorithm 2 Efficient Guessing Algorithm

```

while Cookie not found do
  for  $\hat{x} = 1, 2, 4, 8 \dots$  do
    Go to  $\hat{x}$ 
    Go to 0
    Go to  $-\hat{x}$ 
    Go to 0
  end for
end while

```

In this case, the distance that the ant travels will be upper-bounded by:

$$\begin{aligned} \# \text{ steps} &\leq 4 \cdot 1 + 4 \cdot 2 + 4 \cdot 4 + \dots + 4 \cdot x \\ \# \text{ steps} &\leq 4 \sum_{i=0}^{\log x} 2^i \\ \# \text{ steps} &\leq 4 \sum_{i=0}^{\log x} \frac{x}{2^i} \\ \# \text{ steps} &\leq 4x \sum_{i=0}^{\log x} \frac{1}{2^i} = O(x) \end{aligned} \tag{9}$$

where Line (9) follows from our bound for geometric series. In this case, this sum is a geometric series with $r = \frac{1}{2}$, and thus the sum is $O(1)$.

2.2 Problem 2: Ant on a Tree

We shift our attention to a different problem to demonstrate the other proof strategies. First, we define two types of trees: (1) a rooted tree is d -ary if all non-leaf nodes have d children, and (2) a rooted tree is d -perfect if it is d -ary and all of its leaf nodes are at the same depth.

Case 1: d -perfect tree

Assume that an ant starts at the root of a d -perfect tree, and that all the leaves have cookies. If the ant wants to reach a cookie, it can simply choose an arbitrary child node to go to until it reaches a cookie.

Algorithm 3 Perfect Tree Cookie Algorithm

```
while Cookie not found do  
    Go to arbitrary child  
end while
```

We can analyze the number of steps that the ant takes to reach a cookie with the help of doubling and halving arguments.

2.2.1 Strategy 3: Doubling Argument

If

1. $y \geq 1$ initially
2. y only increases over time
3. $y \leq B$ always

then, the number of times that y scales by d , $y \leftarrow y \cdot d$, is $\leq \log_d B$.

We use this argument to bound the number of steps the ant takes with Algorithm 3.

Let $y := \#$ of nodes at ant's level.

1. $y \geq 1$ initially, there is 1 node at root level (being the root).
2. For $d > 1$, y never decreases. Instead, at each step $y \leftarrow y \cdot d$, because each node has d children.
3. $y \leq n$ always, because there are n total nodes at the tree, thus the number of nodes at a certain level cannot exceed this.

By our doubling argument, we know that the number of times $y \leftarrow y \cdot d$ is $\leq \log_d B$. Thus, $\#$ of steps $\leq \log_d n$.

2.2.2 Strategy 4: Halving Argument

If

1. $y \leq B$ initially
2. y only decreases over time
3. $y \geq 1$ always

then, the number of times that y shrinks by d , $y \leftarrow \frac{y}{d}$, is $\leq \log_d B$.

We use this argument to bound the number of steps ant takes the ant takes with Algorithm 3 in a very similar fashion to the doubling argument.

Let $y := \#$ of nodes in ant's subtree.

1. $y \leq n$ initially, because $y = n$. The subtree rooted at the root node contains all the nodes of the tree, and therefore has n nodes.
2. $y \geq 1$ always, because a subtree always has a root, and therefore at least one node.
3. After each step $y \leftarrow \frac{y}{d}$. Because the tree is d -perfect, all child nodes will have equal subtree sizes. Since the root of the subtree has y nodes under it, splitting these nodes (except the root of the subtree) among d children gives $\frac{y-1}{d} \approx \frac{y}{d}$ node per child subtree.

Then, by our halving strategy, we conclude that y shrinks by d at most $\log_d n$ times, meaning $\#$ steps $\leq \log_d n$.

With our proofs using the doubling and halving strategies, we conclude that the depth of a d -perfect tree is $\leq \log_d n$. For different values of d we have different depth upper-bounds:

d	2	$\log n$	$2^{\sqrt{\log n}}$	\sqrt{n}	n^ϵ
Depth	$\log n$	$\frac{\log n}{\log(\log n)}$	$\sqrt{\log n}$	2	$\frac{1}{\epsilon}$

Case 2: d -ary tree

Now we shift our attention to what would happen if the ant was on a d -ary tree instead. Going to an arbitrary child would eventually lead to a cookie, however in a very pathological case, the ant could get unlucky and end up taking a lot of steps. As an example consider a 2-ary (binary) tree where the left child of a node is always a leaf. For this to be possible, the right child of the root must contain $n - 2$ nodes in its subtree. As long as the ant keeps choosing the right child, the number of nodes in the subtree will decrease by 2 each step, leading to $\approx \frac{n}{2}$ steps, and therefore a linear runtime.

We instead opt for a more greedy approach, and choose the child with the minimum number of nodes in its subtree. We analyze the number of steps that the ant will take with this approach using the averaging argument.

Algorithm 4 *d*-ary Tree Cookie Algorithm

while Cookie not found **do**

 Go to child *j*, minimizing y_j (size of node *j*'s subtree)

end while

2.2.3 Strategy 5: Averaging Argument

Given a collection of real numbers $y_1, y_2, \dots, y_d \in \mathbb{R}$, there always exists an element in the collection with a value less than or equal to the average, namely the minimum of the collection.

$$\exists y_i \leq \frac{\sum_i y_i}{d}$$

From this averaging fact, we know that the choosing the child *j* with minimum subtree size will give us:

$$y_j \leq \frac{\sum_i y_i}{d}$$
$$y_j \leq \frac{y}{d}$$

And once again, by halving argument, we note that when each step shrinks our subtree size by a factor of *d*, the total number of steps will be upper-bounded by $\log_d n$. Therefore, this algorithm will have a run-time of $O(\log_d n)$ in a *d*-ary tree.