Q: What is the complexity of a randomized algorithm?

A: A randomized algorithm is one that can produce different results even when run multiple times with the same inputs.

In the case of the algorithm described:

- The expected running time is $O(n \log n)$.
- The probability of exceeding $O(n \log n)$ is less than $\frac{1}{n}$. Therefore, the algorithm is considered efficient.

This is because the algorithm runs in time $T(n)$ with high probability if $p(n) = \frac{1}{n}$, suggesting that the algorithm performs well in practice.
Q: do same?

alg: quickest-update

while

- run best quicksort(a) for 2(C(n) log n)

- keep track of best

- after O(n log n) steps

- expected running

- Euclidean case

if b is sorted, copy b

check

no need to check if sorted

prop: quickest-update always correct [claim 2]

prop: quickest-update runs in O(C(n) log n) time with probability 1 - \( Y_n \) & is 1 - \( Y_n \)

pl: check if b is sorted is O(n) time & single pass, take O(n) vs.

L be number of pics

claim: runtime \( = \binom{L + 1}{2} C(n) + O(n) \)

\( X_i \) = 

2 \( \text{in change of quicksort successive} \)

2(C(n) log n) \

claim: \( P(X_i = 0) = \frac{1}{2^v} \text{quickest update next} = 2 \text{Euclidean} \)

\( L \geq Y_n \)

\( \forall \text{runtime} \geq 2(C(n) \log n) \)

V: quickest-update runs in \( O(\log n + n) \) time w/ 1 - \( Y_n \) & why?

\[ \begin{align*}
\text{compute to find expected running time} & = 0(n \log n) \text{ time up to } 1 - \gamma_n, \text{any } \delta(\gamma)
\end{align*} \]

\[ \begin{align*}
\text{compute running time & height of } \gamma
\end{align*} \]

Q: avoid pending testing expected runtime & why runtime? A better analysis of the [claim 2]

pl: quickest-update runs in \( O(\gamma_0) \log n \) time with probability 1 - \( Y_n \), any \( \delta(\gamma_0) \)

\( n^2/10 \) time & \( \gamma \leq \gamma_0 \) [claim 2]

den

Q: it a bunch of total brain?

Q: what is experience?

\( \begin{align*}
\text{distribution of human height, peak } \delta \text{ vs. typical } 2
\end{align*} \)

\( \begin{align*}
\text{ms} \text{. central limit theorem } \rightarrow
\end{align*} \)

\( \begin{align*}
X_1, X_2, \ldots \text{ random for each IR, identical and independent, } \sum \text{ then } \gamma_n \rightarrow \gamma \text{ gaussian distribution}
\end{align*} \)

Q: given what case? Crude

\[ \begin{align*}
\text{place } 1/E(\gamma) \text{ in model 2}
\end{align*} \]
The Chernoff bound:

$$X = X_1 + \cdots + X_n \geq \xi \quad \text{with probability } \geq 1 - \frac{1}{e^{2\xi^2}}$$

Pr[\text{average}] = \frac{e^{\xi^2/2}}{e^{\xi^2}} \geq 2 - e^{-2\xi^2/2}

The more general case exists.

Let \( \bar{X} \) be the average of independent random variables.

If \( \bar{X} \) is normal, then

$$\Pr[\bar{X} \geq \xi] \geq 1 - e^{-\xi^2/2}$$

With \( n = \Theta(n) \),

$$\Pr[\text{max } Y_i \geq c \text{ for some } i] \leq n \cdot e^{-\Theta(c \ln c)}$$

Choose \( c = k \cdot \ln \ln n \).

$$\Pr[\text{max } Y_i \geq c] = \Pr[\text{max } Y_i \geq c] \leq \frac{1}{\ln \ln n}$$

$$n \cdot e^{-\Theta(c \ln c)}$$

Each bin has \( E = 1 \) individually, but the max load expected to be higher.
\[
\begin{align*}
\text{Then [Chengu7]: } & \quad X_i, \ldots, X_n \in \mathcal{E}(\mathbb{N}) \quad \text{independent, } \quad \mathbb{E}[X_i] = p \cdot c \cdot t \\
\chi & = X_1 + \cdots + X_n \text{. Here, } r \cdot \mathbb{E}[X] = (1 + t)^r \mathbb{E} \left( \frac{c^r}{r!} \chi \right) \\
p & \quad \text{set: } \quad M := \text{Markov's inequality} \\
\end{align*}
\]

\text{\textbf{t > 0 \quad powerful \quad 1: } } x \rightarrow e^{tx} \quad \text{is strictly increasing function}

\begin{align*}
\mathbb{P}[X = (1+\delta)^n] = & \quad \mathbb{P}[X > e^{t(1+\delta)/n}] \\
\mathbb{E}[X] \leq & \quad \mathbb{E}[e^{tX}] / e \\
\mathbb{E}[e^{tX}] = & \quad \mathbb{E}[e^{(X_1 + \cdots + X_n)}] \\
\mathbb{E}[e^{tX}] & = \mathbb{E}[e^{tX_1}] \cdots \mathbb{E}[e^{tX_n}] \\
\mathbb{E}[e^{tX_i}] & = e^{t \cdot p} \mathbb{E}[X_i] = 1 + (e^{t} - 1) = e^{t} \\
\end{align*}

\text{choose } t = \ln(1+\delta)

\begin{align*}
\mathbb{E}[X] = & \quad \frac{e^{\mu}(e^{\delta - 1})}{e^{\mu}(1+\delta)} \\
\text{today: } \quad \text{randomized CDFs} \quad \text{complexity of randomized algo.} \quad \text{expected \& worst \ cases} \\
\text{Chernoff \& Markov} \quad \text{tail bounds} \quad \text{Chernoff-Chebyshev} \\
\text{more complicated} \quad \text{distribution for } k \\
\end{align*}

\text{\textbf{HW:} \quad \text{randomized CDFs} }