Average Case

- WC is "pessimistic"
- For most algorithms, still representative, but for some, WC is not representative

Example:

```python
def search(L: List[int], x: int) → bool:
    for item in L:
        if item == x:
            return True
    return False
```

Complication: need to define carefully all possible inputs—specifically, we need a set of inputs that captures all possible behaviours of the algorithm.

Approach: \( P_n = \{\text{all permutations of } [1, 2, \ldots, n]\} \)

(\( P_3 = \{[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]\} \))

For `search`, define \( \mathcal{I}_n = \{ (L, 1) \mid L \in P_n \} \)

(e.g., \( \mathcal{I}_3 = \{(1, 2, 3, 1), (1, 3, 2, 1), \ldots\} \))

\[
AC(n) = \frac{1}{|I_n|} \sum_{(L, 1) \in I_n} RT(L, 1)
\]  

\((\text{Avg}(n) \text{ in course notes})\)
scaling factor for average
add value of expression for each elem. in set $X_n$
running time of algo. on input $(L, 1)$

$|X_n| = |P_n| = n! = n \times (n-1) \times (n-2) \times \cdots \times 2 \times 1$

# possibilities: $[\frac{n}{\mathbf{1}}, \frac{n}{\mathbf{2}}, \frac{n}{\mathbf{3}}, \ldots, \frac{n}{\mathbf{2}}, \frac{n}{\mathbf{1}}]$

$RT(L, 1) = \# \text{iterations of for-loop}$
$= \# \text{times if-statement executes}$
$= 1 + \text{index of value 'i' in L}$

[1, ..., 1, ..., 1, ..., 1, 1, ..., 1]

$AC(n) = \frac{1}{n!} \sum_{(L, i) \in X_n} (1 + \text{index of value 'i' in L})$

idea: introduce new variable $i$ to represent *

$= \frac{1}{n!} \sum_{i=0}^{n-1} \sum_{L \in P_n, L[i]=1} (1+i)$

how many lists in $P_n$ have $L[i]=1$? $(n-1)!$

$= \frac{1}{n!} \sum_{i=0}^{n-1} (n-1)!(1+i)$

$= \frac{(n-1)!}{n!} \sum_{i=1}^{n} i = \frac{1}{n} \frac{n(n+1)}{2} = \frac{n^2+1}{2}$
Choice of $I_n$?
- Does not account for $x \notin L$
- Does not account for repeated elements

Doing this "properly" for algo. search:
1. $I_n = \{(v_1, v_2, \ldots, v_n, x) \mid x = 0, 1, 2, \ldots, n\}$
2. Set up appropriate probability distribution over $I_n$
3. Compute expected running time...

*Test 2 solutions are available — grades tomorrow
*Exam cover page & aid sheet on website

**Graphs**

1. $\forall G = (V, E), |E| \leq \frac{|V|(|V|-1)}{2}$

NOTE: only exception allowed to quantifier expressions introduces 3 related variables $(G, V, E)$.

Proof: every element of $E$ corresponds to a subset of $V$ of size 2 — so by worksheet #10

$|E| \leq \frac{|V|(|V|-1)}{2}$

2. For all $G = (V, E)$ and $u, v \in V$, define
\[ \text{Conn}(G, u, v): \text{"} u \text{ and } v \text{ are connected in } G \text{"} \]
\[ \iff G \text{ contains a path from } u \text{ to } v \]
\[ \iff \exists N_1, \ldots, N_k \in V, (u, N_1) \in E \land (N_1, N_2) \in E \land \cdots \land (N_{k-1}, N_k) \in E \land (N_k, v) \in E \]
\[ \text{\textit{\"G is connected\"}: } \forall u, v \in V, \text{Conn}(G, u, v) \]

3. Study \underline{necessary} and \underline{sufficient} conditions on \(|E|\) for \(G\) to be connected.

\underline{sufficient}: condition \(\implies G\) is connected

\underline{necessary}: \(\neg\)condition \(\implies G\) is not connected

\textit{Necessary:} \(|E| \geq |V|-1 \quad - \text{course notes} \)

\textit{Sufficient:}

\[
\forall G=(V, E), |E| \geq \frac{(|V|-1)(|V|-2)}{2} + 1 \implies G \text{ is connected}
\]

\underline{Proof:} \underline{Trick: introduce a variable to do induction on} \quad - \text{corresponding to the size of graphs}

\[
\forall n \in \mathbb{Z}^+, \forall G=(V, E), |V|=n \Rightarrow \]
\[
P(n): \left( |E| \geq \frac{(n-1)(n-2)}{2} + 1 \implies G \text{ is connected} \right)
\]

\textit{Base Case:} \((n=1)\)

\textit{EXERCISE:} convince yourself \(P(1)\) is vacuously true
Ind. Hyp.: Let \( k \in \mathbb{Z}^+ \) and assume \( P(k) \):
\[
\forall G = (V, E), |V| = k \Rightarrow (|E| \geq \frac{(k-1)(k-2)}{2} + 1 \Rightarrow G \text{ is connected})
\]

Ind. Step: WTS: \( P(k+1) \):
\[
\forall G = (V, E), |V| = k + 1 \Rightarrow (|E| \geq \frac{k(k-1)}{2} + 1 \Rightarrow G \text{ is connected})
\]

Let \( G = (V, E) \). Assume \( |V| = k + 1 \).
Assume \( |E| \geq \frac{k(k-1)}{2} + 1 \).
WTS: \( G \) is connected.

### ROUGH WORK

Consider \( G' = (V', E') \)
\( V' = V - \{v_0\} \)
\( E' = E - \{ (u, v_0) \mid (u, v_0) \in E \} \)

So,

\[
|V'| = k
\]
\[
|S_0| \leq k
\]
\[
|E'| = |E| - |S_0|
\]
\[
\geq \left( \frac{k(k-1)}{2} + 1 \right) - k = \frac{(k-1)(k-2)}{2}
\]

Idea: if there is \( v_0 \in V \) with fewer than \( k \) edges in \( G \), then let \( G' \) be as above and apply ind. hyp.
else, \( G \) contains \( \frac{(k+1)k}{2} \) edges and is connected.