1. $X : S \to \mathbb{R}$

2. $$E(X) = \sum_{\omega \in S} X(\omega) \Pr(\omega)$$
$$= \sum_{x} x . \Pr(X = x)$$

3. $$E(X+Y) = E(X) + E(Y)$$

$$E\left(\sum_{i=1}^{n} a_i X_i\right) = \sum_{i=1}^{n} a_i . E(X_i)$$

$$E\left(\sum_{i=1}^{\infty} X_i\right) = \sum_{i=1}^{\infty} E(X_i)$$

(provided some convergence conditions hold)

LINEARITY OF EXPECTATION

4. Geometric Distribution $E[X] = 1/p$

5. Binomial Distribution $E[X] = pn$

6. INDICATOR R.V. $E[X] = \Pr[X = 1]$

Example 2: Largest element

Consider the following algo to compute max of n-elements $s_1, s_2, s_3, ----, s_n$.

Algorithm FINDMAX $(s_1, s_2, ----, s_n)$

max := $-\infty$

for $i := 1$ to n do
    if $s_i >$ max then
        max := s_i —— $\boxed{**}$

Return max.

Question: How many times $(**)$ executes, when $s_1, s_2, ---, s_n$ is a random permutation?

Observe: $\boxed{**}$ may execute n-times. (input: $1, 2, 3, ---, n$)

$\boxed{**}$ may execute only once (input: $n, n-1, ---, 3, 2, 1$)

Let $X = \#$ times max changes on an input sequence $s_1, s_2, ----, s_n$

Question: $E(X) = ?$

Define n- indicator r.v. $X_1, X_2, \ldots, X_n$

$$X_i = \begin{cases} 1 & \text{if max changes - i.e line (**) - is executed in the } i^{th} \text{ iteration} \\ 0 & \text{otherwise} \end{cases}$$

Note that $X = X_1 + X_2 + \cdots + X_n$

Hence, $E(X) = E(X_1 + X_2 + \cdots + X_n)$

$= E(X_1) + E(X_2) + \cdots + E(X_n)$

by linearity of ~~xep~~ expectation

Thus $E(X) = \sum_{i=1}^{n} E(X_i) = \sum_{i=1}^{n} Pr(X_i = 1)$.

Let us compute $Pr(X_i = 1) = ?$

When is $X_i = 1$?

This is when in the subsequence $s_1, s_2, \ldots, s_i$ the maximum value is at the last position.

But $s_1, s_2, \ldots, s_i$ is a subsequence of a random permutation - thus max could be in any of i positions.

$Pr\left(\begin{matrix}\text{Max is at the last position} \\ \text{in } s_1, s_2, \ldots, s_i\end{matrix}\right) = \frac{1}{i} = Pr(X_i = 1)$

Hence

$$E(X) = \sum_{i=1}^{n} \Pr(X_i = 1) = \sum_{i=1}^{n} \frac{1}{i}$$

$$= \underbrace{1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \cdots + \frac{1}{n}}_{n\text{th Harmonic-number}}$$

$$\sim 1 + \ln n$$

(see section 6.8.3)

Claim: Let $H_n = \sum_{i=1}^{n} \frac{1}{i} = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$.

$$\boxed{\ln(n+1) \leq H_n \leq 1 + \ln n}$$

Thus max-value changes about $O(\log n)$ times for a random permutation in the algorithm FINDMAX.

Example 3: Insertion Sort

Algorithm INSERTIONSORT (A[1..n])

```
for i=2 to n do
    j:=i;
    while j>1 and A[j] < A[j-1]
        - swap A[j] and A[j-1]
        - j:=j-1
```

Note that at the start of the i^{th} iteration

- the subarray $A[1..i-1]$ is sorted
- algorithm hasn't seen any of the elements in $A[i..n]$.

Question: What is the total # of swaps the algorithm makes?

→ It can be anywhere from 0 to $\binom{n}{2}$

0 for sorted inputs: $1, 2, 3, 4, \ldots, n$

$\binom{n}{2}$ for reverse-sorted inputs: $n, n-1, n-2, \ldots, 1$.

Suppose the array A consists of a uniform random permutation of integers $1, 2, 3, \ldots, n$.

Let X = # of swaps made by INSERTIONSORT on this input.

X is a r.v from permutations of $\{1, 2, 3, \ldots, n\}$ to positive integers.

We are interested in computing $E[X] = ?$

Define a indicator r.v X_{ab} for $1 \leq a < b \leq n$ as follows

$$X_{ab} = \begin{cases} 1 & \text{if } a \,\&\, b \text{ are swapped by the algo} \\ 0 & \text{otherwise.} \end{cases}$$

Note that since $a < b$, the reason that a and b will be swapped iff "b is to left of a in the input permutation".

Thus the event "b is to the left of a" corresponds to $X_{ab} = 1$. Since we have a random permutation, there is an equal chance that ~~to~~ b is to the left of a, or b is to the right of a.

Thus

$$E[X_{ab}] = Pr[X_{ab} = 1] = 1/2 .$$

Note that

$$X = \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} X_{ab}$$

(as each pair a, b can be swapped at most once by the algorithm)

Thus

$$E[X] = E\left[\sum_{a=1}^{n-1} \sum_{b=a+1}^{n} X_{ab}\right]$$

Linearity of expectation

$$= \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} E[X_{ab}]$$

Indicator r.v

$$= \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} Pr[X_{ab} = 1]$$

$$= \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} \frac{1}{2}$$

$$= \frac{1}{2}\binom{n}{2} \blacksquare$$

Example 4: Quicksort

Input: An array of n-distinct numbers, and it is random permutation of $\{1..n\}$.
Output: Same array, elements are in sorted order.

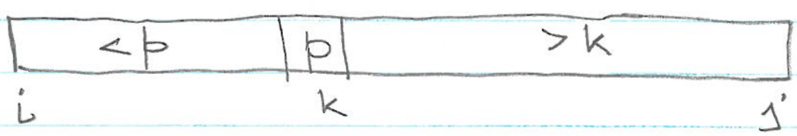
Generic call to QUICKSORT(A, i, j) takes two indices $i < j$, and sorts the subarray $A[i..j]$.

Algorithm QUICKSORT(A, i, j):

if $i < j$ then

$p :=$ uniformly random element in $A[i...j]$;

Compare p with all elements in $A[i..j]$ and rearrange the elements so that the following holds



QUICKSORT$(A, i, k-1)$;
QUICKSORT$(A, k+1, j)$;

As we saw in the very first class, p is called the pivot.

The worst case running time of Quicksort is $\Theta(n^2)$.

We will show that the expected running time of Quicksort $(A, 1, n)$ is $O(n \log n)$.

Let X be the r.v. denoting

X = total # of comparisons performed in algorithm QUICKSORT $(A, 1, n)$.

We will show that

$$E(X) = O(n \log n).$$

For each a and b, where $1 \le a < b \le n$, define

$$X_{ab} = \begin{cases} 1 & \text{if } a \text{ \& } b \text{ are compared in Quicksort}(A,1,n) \\ 0 & \text{otherwise.} \end{cases}$$

Since each pair of input elements are compared once,

$$X = \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} X_{ab}$$

Thus,
$$\begin{aligned} E(X) &= E\left(\sum\sum X_{ab}\right) \\ &= \sum\sum X_{ab} \\ &= \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} \Pr(X_{ab} = 1). \end{aligned}$$

What is $Pr(X_{ab} = 1) = ?$

What is the probability that the elements a & b are compared to each other when running QUICKSORT(A, 1, n)?

Define the set $S_{ab} = \{a, a+1, ---, b\}$.

At the start of the algorithm, $S_{ab} \subseteq S$ and thus all elements of S_{ab} are part of input to Quicksort(A, 1, n).

Consider the first pivot p that is chosen.

Consider the following 2 cases:

(1) $p \notin S_{ab}$

- If $p < a$, after the algorithm has rearranged the input array, all elements of S_{ab} are to the right of p.

  All elements of S_{ab} are part of the input to recursive call QUICKSORT(A, k+1, n).

  During rearrangement, a & b were not compared to each other, but in the recursive call, they may get compared.

- If $p > b$, then by similar reasoning, all elements of S_{ab} are part of recursive call QUICKSORT(A, 1, k-1).

(2) $p \in S_{ab}$

- if $p \neq a$ and $p \neq b$, then after the algorithm has rearranged, a is to left of p, b is to right of p, and a & b are "separated". They are not compared to each other during the rearranging. Also, they will not be compared ever after, as they are seperated. Thus $X_{ab} = 0$

- if $p = a$ or $p = b$, then during the rearranging a & b are compared to each other. In future recursive calls, they are never compared again. Thus $X_{ab} = 1$.

Conclusions so far:

Whether or not a and b are compared to each other is determined completely by the first element p that is chosen as pivot in the set S_{ab}.

Moreover, only if $p=a$ or $p=b$, a & b are compared to each other.

$$X_{ab} = \begin{cases} 1 & \text{only if } p=a \text{ or } p=b \\ 0 & \text{otherwise.} \end{cases}$$

Since, p is chosen uniformly at random, $Pr(p=a \text{ or } p=b) = \dfrac{2}{b-a+1}$ as there are $b-a+1$ elements in S_{ab}, each of them has equal chance of being chosen as p, and $Pr(p=a \text{ or } p=b) = \dfrac{2}{b-a+1}$.

Hence $Pr(X_{ab}=1) = Pr(p=a \text{ or } p=b$ in S_{ab} where p is the first pivot element from $S_{ab})$

Thus,

$$E(X) = \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} P_r(X_{ab} = 1)$$

$$= \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} \frac{2}{b-a+1}$$

$$= 2 \sum_{a=1}^{n-1} \left(\frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n-a+1} \right)$$

$$\leq 2 \sum_{a=1}^{n-1} \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n-a+1} + \cdots + \frac{1}{n}$$

$$= 2 \sum_{a=1}^{n-1} (H_n - 1)$$

$$= 2(n-1)(H_n - 1)$$

$$\leq 2n(H_n - 1)$$

Thus $E(X) \leq 2n \ln n$, as $H_n \sim 1 + \ln n$.

Expected # comparisons made by quicksort on a random permutation of input elements is $\sim 2n \ln n$. ∎