

COL106: Data Structures and Algorithms

Lecture 11

Keerti Choudhary
Department of Computer Science, IIT Delhi

Today's Lecture

Skip Lists: A dictionary based on lists with following guarantees

OPERATION	EXPECTED TIME
Insertion	$O(\log n)$
Deletion	$O(\log n)$
Search	$O(\log n)$

Probability Tool

Union Bound: If X and Y are two events, then

$$\text{Prob}(X \cup Y) \leq \text{Prob}(X) + \text{Prob}(Y)$$

Example: Consider a dice throw experiment. Let X be the event of getting either 1 or 3, and Y be the event of getting a prime number.

$$\text{Prob}(X) = \frac{|\{1,3\}|}{6} = \frac{1}{3}$$

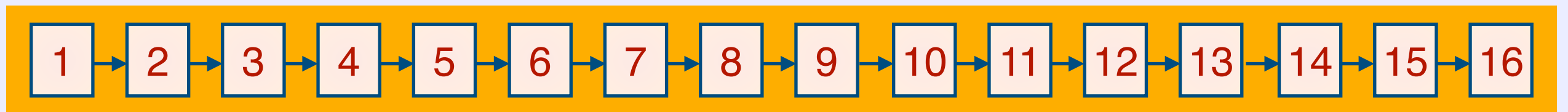
$$\text{Prob}(Y) = \frac{|\{2,3,5\}|}{6} = \frac{1}{2}$$

$$\text{We can verify that - } \text{Prob}(X \cup Y) = \frac{|\{1,2,3,5\}|}{6} = \frac{4}{6} \leq \text{Prob}(X) + \text{Prob}(Y) = \frac{5}{6}$$

Search in Sorted List

Search(x, L_0):

Search x in L_0 , if not found then find Predecessor and Successor of x in L_0

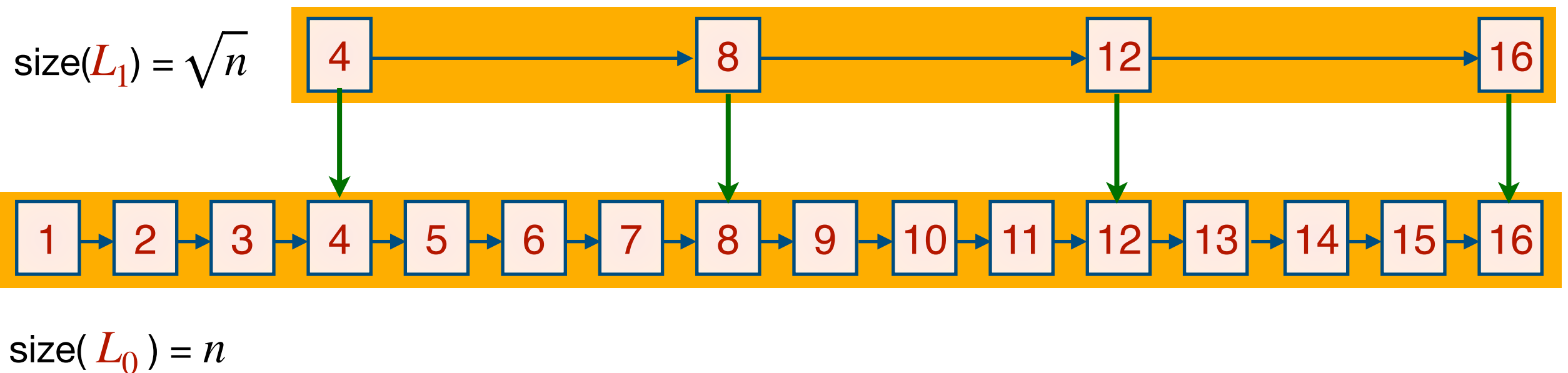


List L_0 of size n

Search Time (in a sorted list of size n) = $O(n)$

Search in Sorted List

List L_1 — Contains elements whose indices are of type $(\sqrt{n} \times i)$



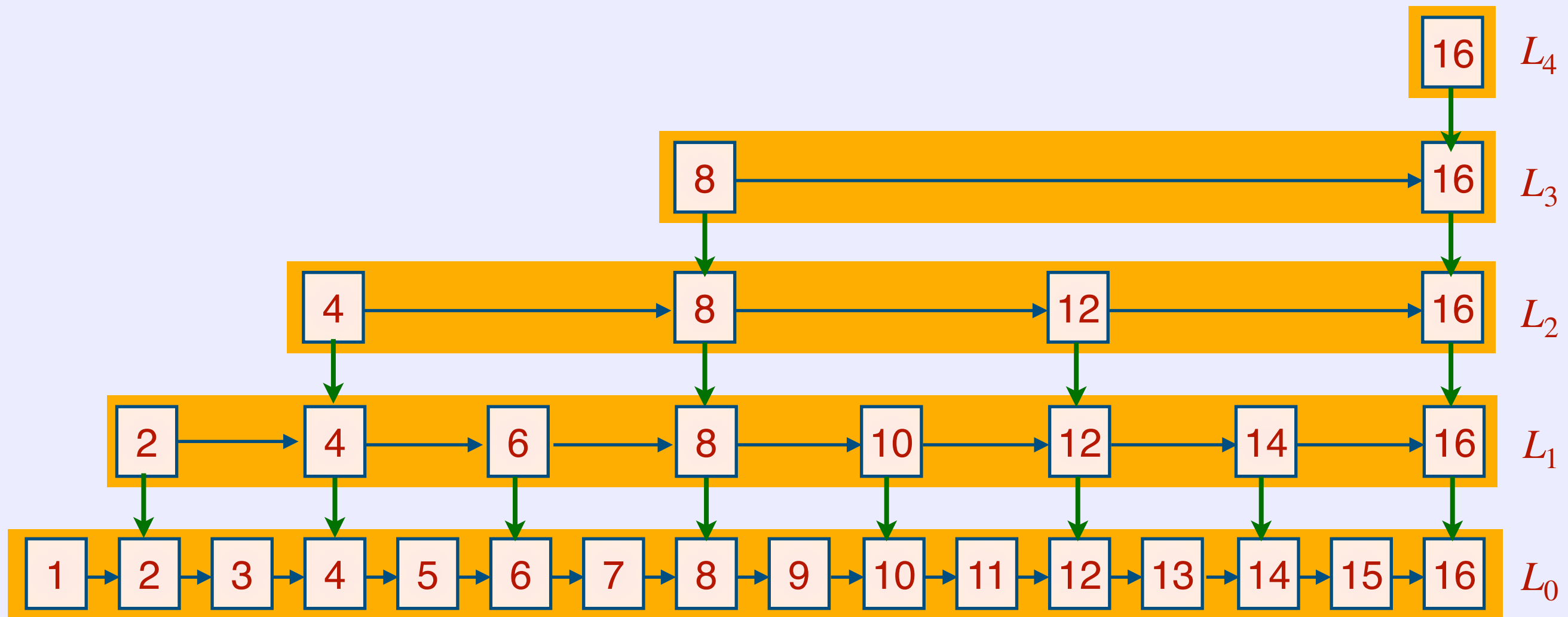
Search Time in list $L_1 = O(\sqrt{n})$

Search Time in L_0 using $L_1 = O(\sqrt{n})$

Search in Sorted List

Generation Rule

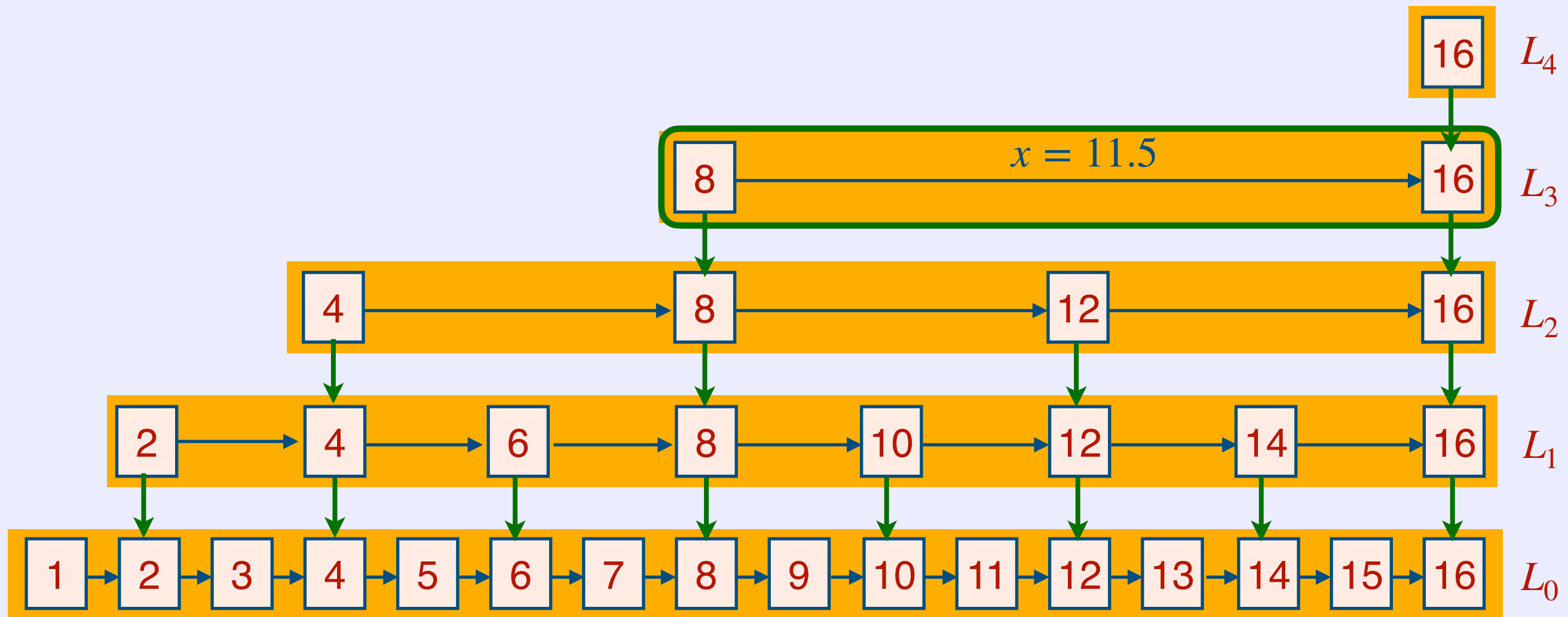
L_{i+1} contains alternate elements of L_i



Search in Sorted List

Generation Rule

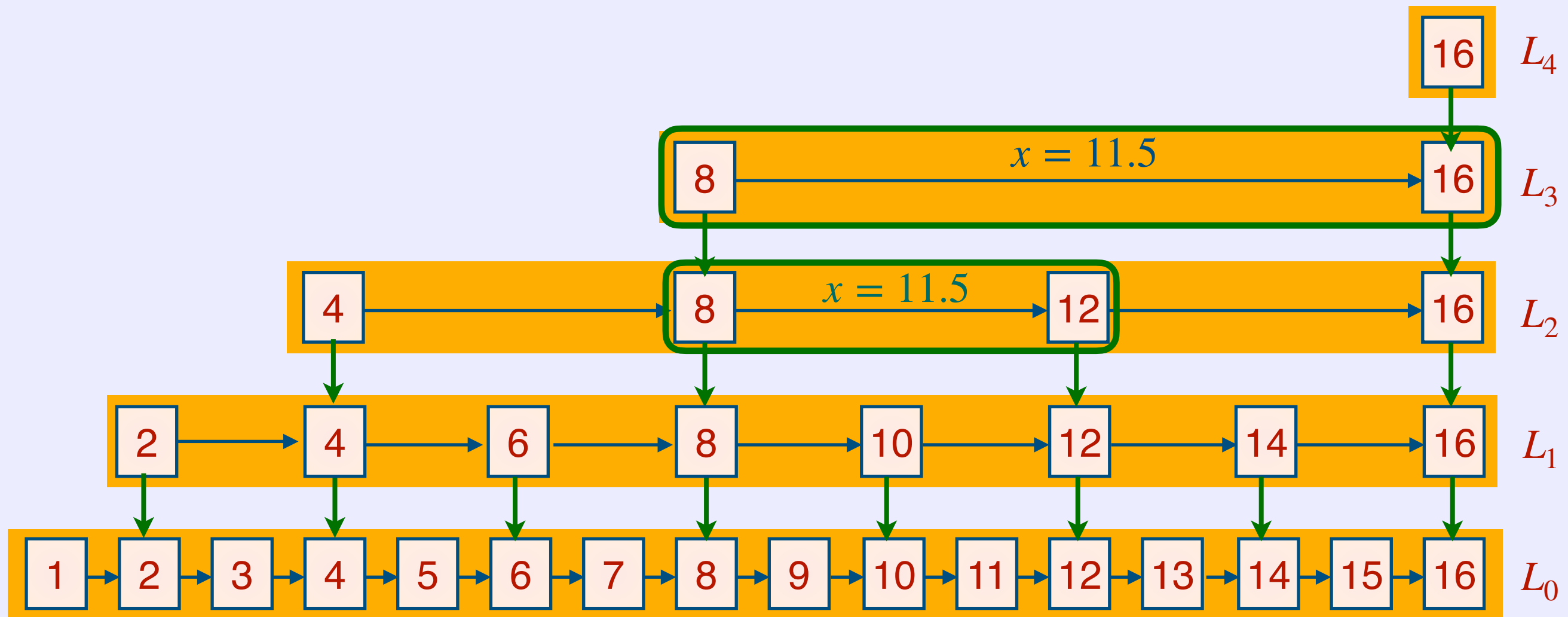
L_{i+1} contains alternate elements of L_i



Search in Sorted List

Generation Rule

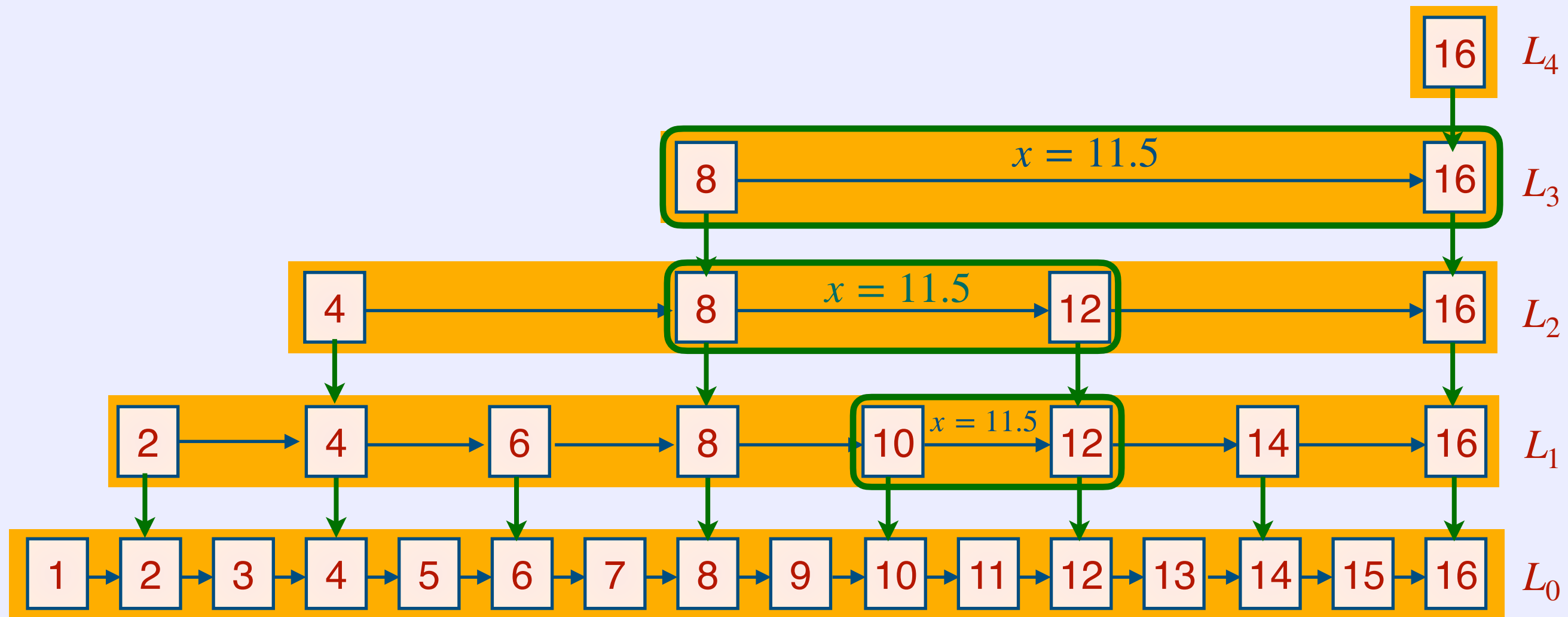
L_{i+1} contains alternate elements of L_i



Search in Sorted List

Generation Rule

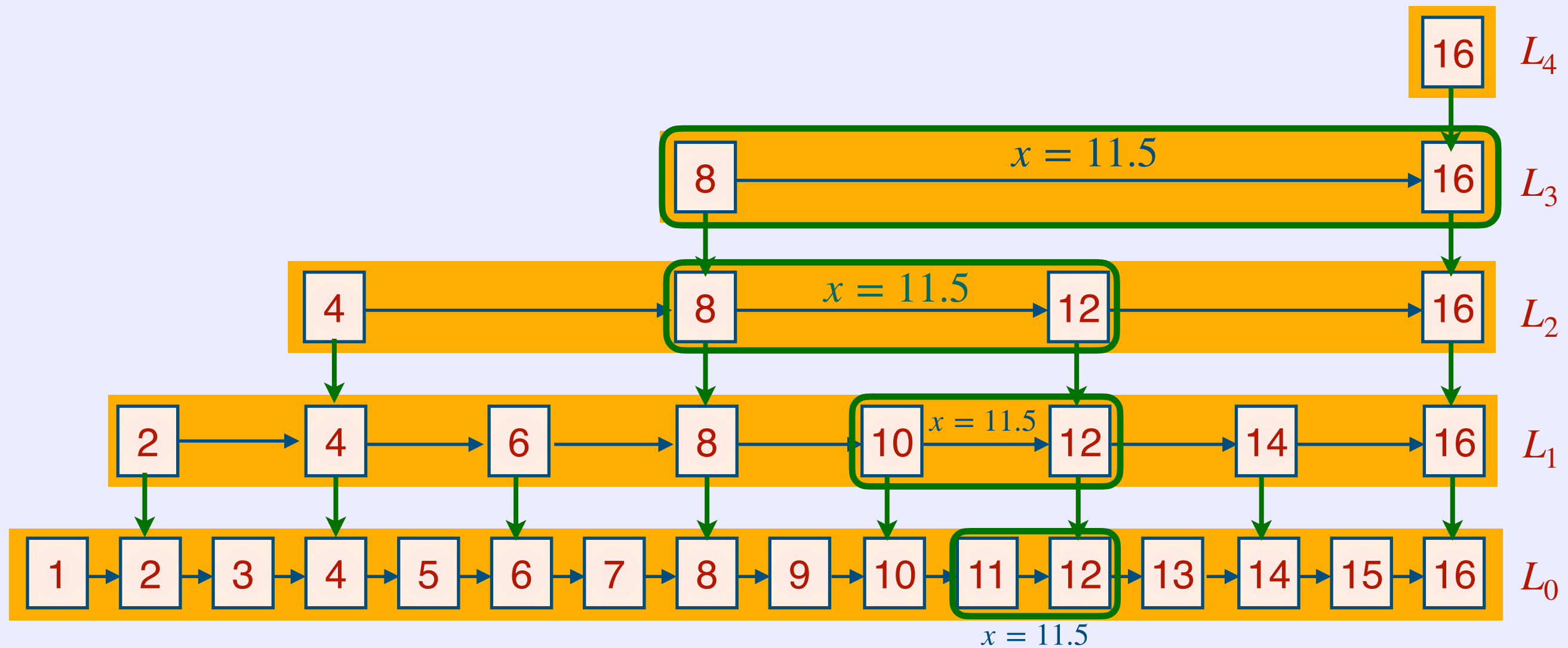
L_{i+1} contains alternate elements of L_i



Search in Sorted List

Generation Rule

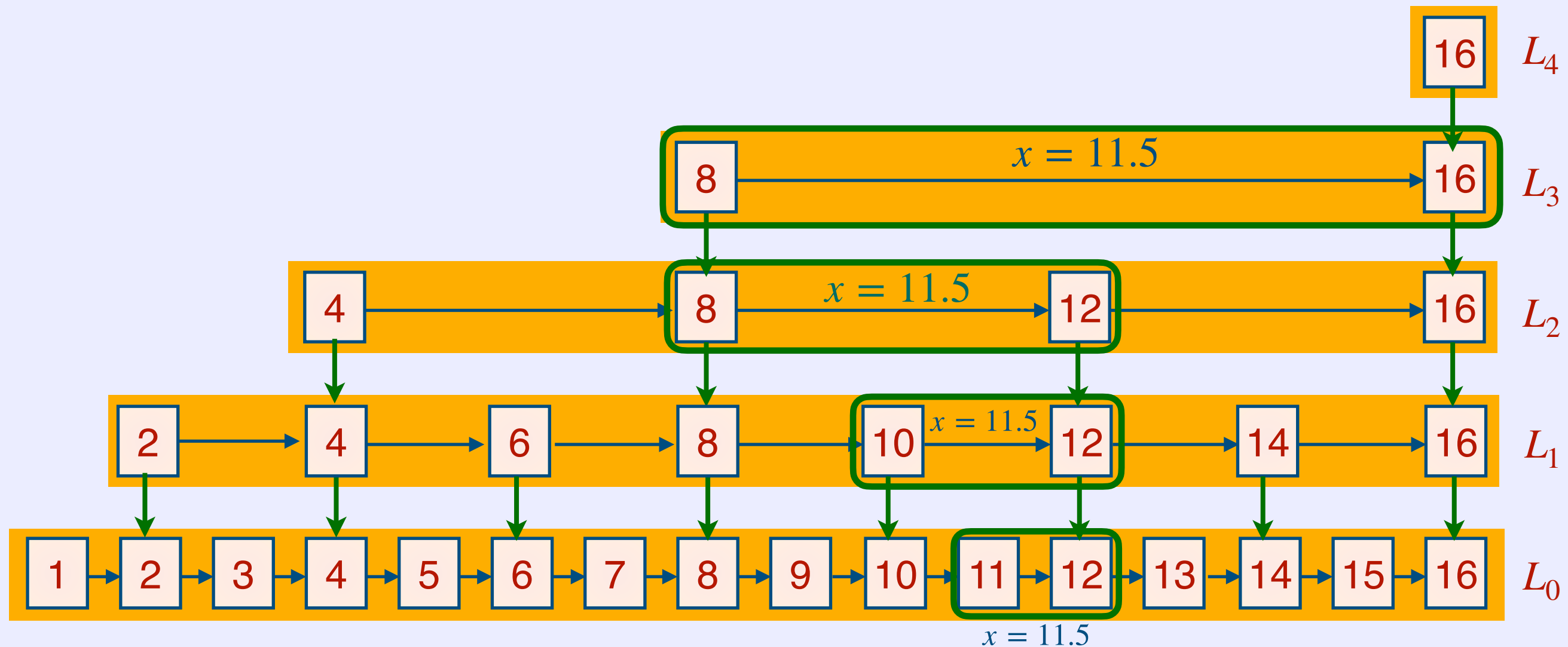
L_{i+1} contains alternate elements of L_i



Search in Sorted List

Generation Rule

L_{i+1} contains alternate elements of L_i



Search Time in
 L_i using L_{i+1}
is $O(1)$

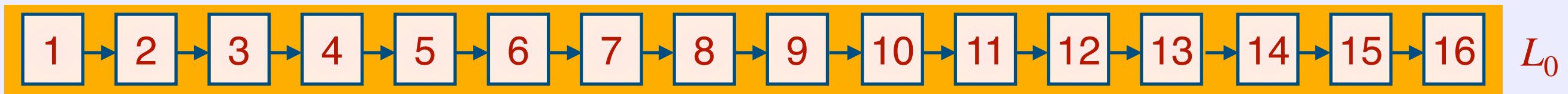
Total Search
time is
 $O(\log_2 n)$

Total Space
used is
 $O(n)$

What if we have randomness?

Generation Rule

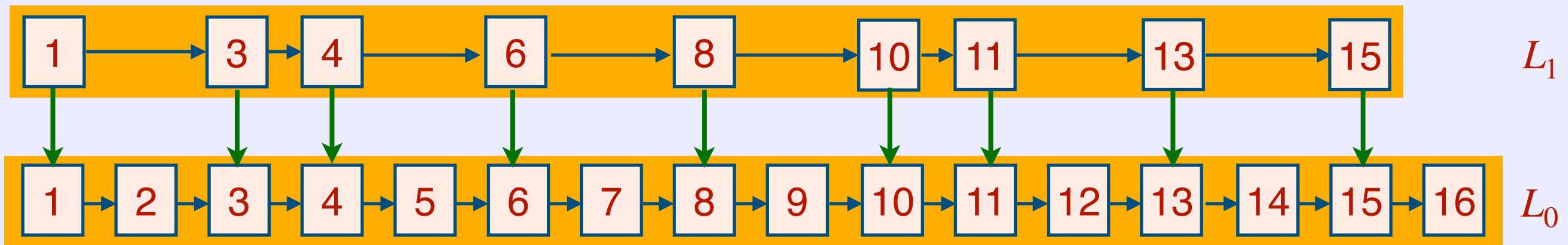
Each element of L_i is chosen
in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

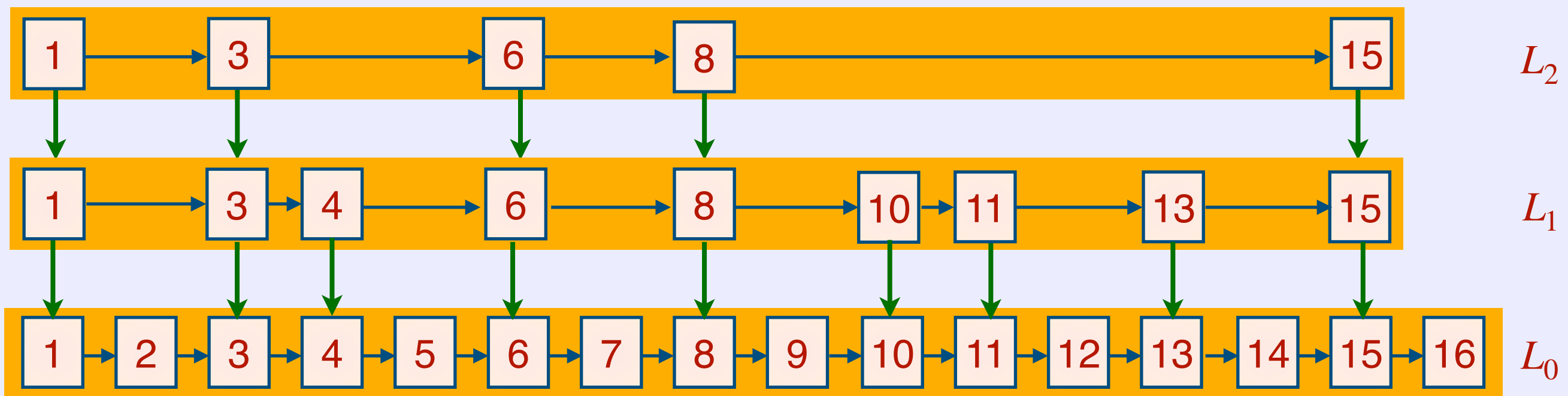
Each element of L_i is chosen
in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

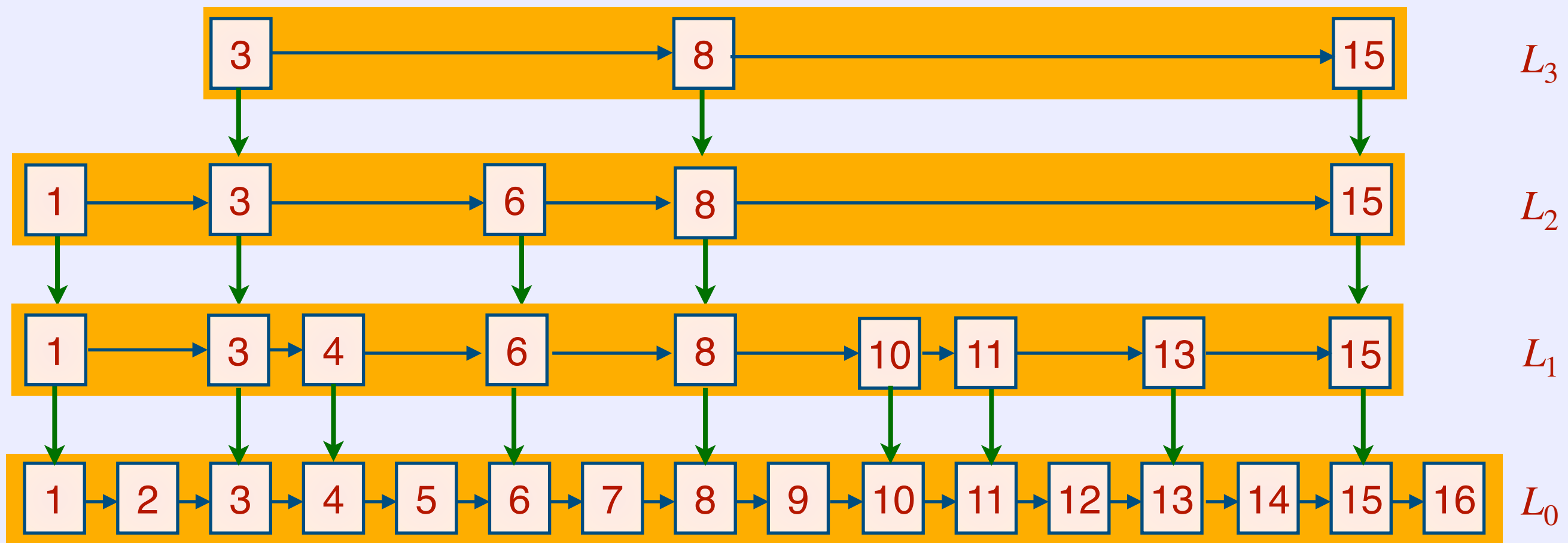
Each element of L_i is chosen in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

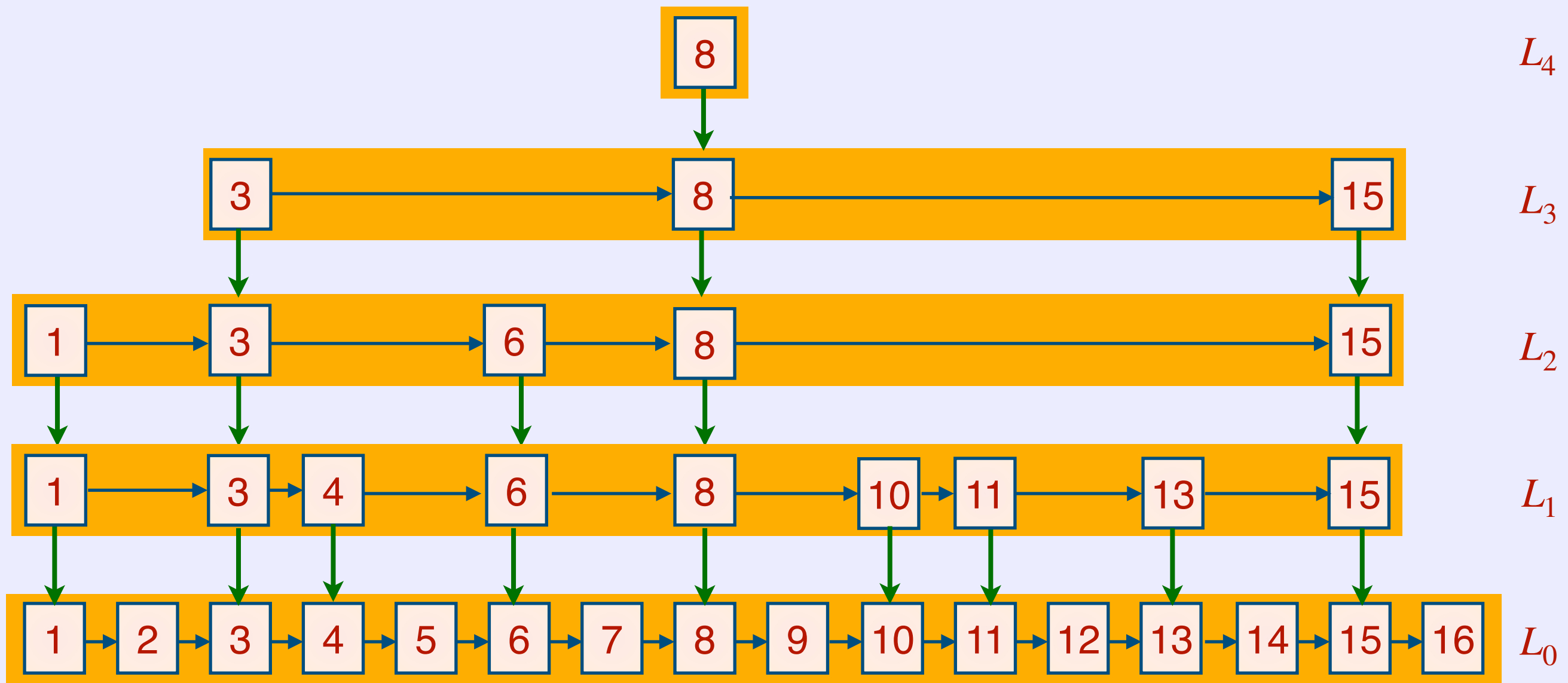
Each element of L_i is chosen in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

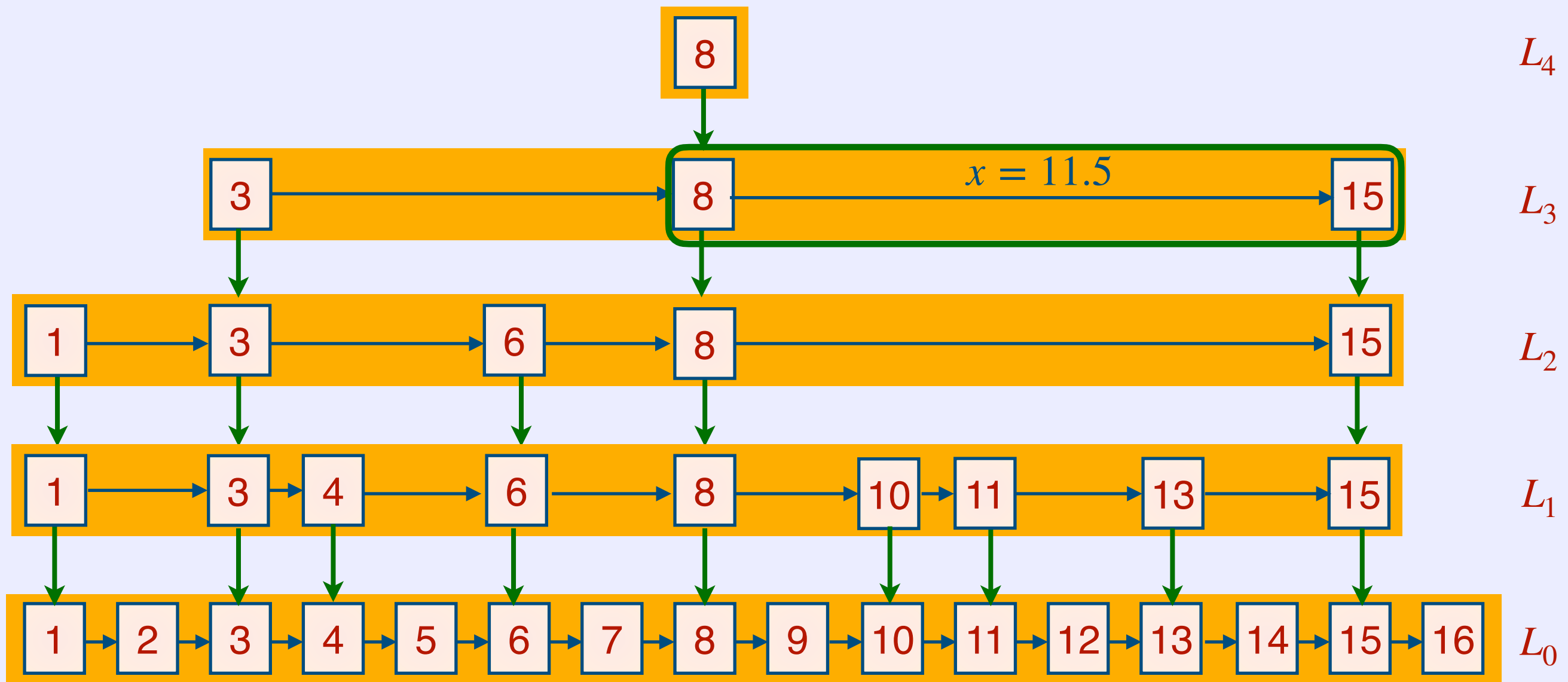
Each element of L_i is chosen in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

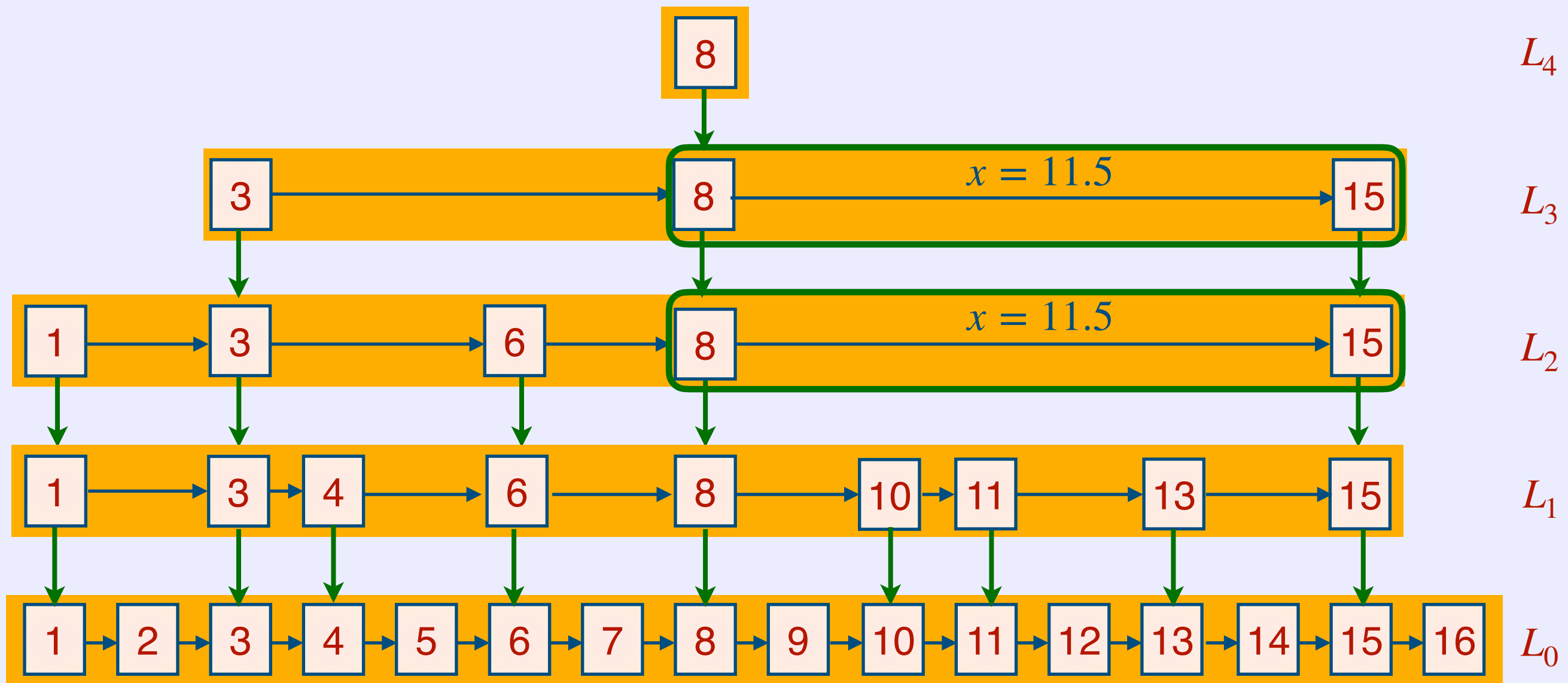
Each element of L_i is chosen
in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

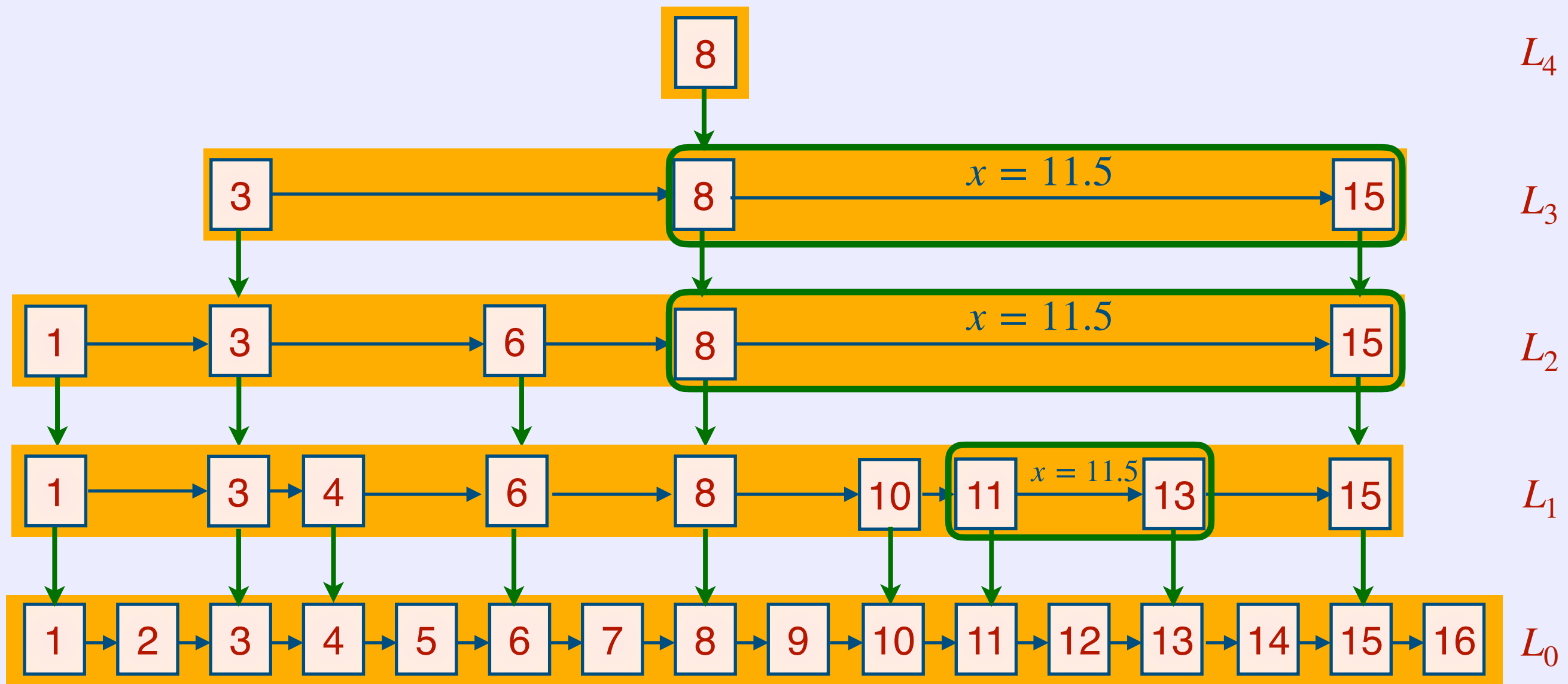
Each element of L_i is chosen in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

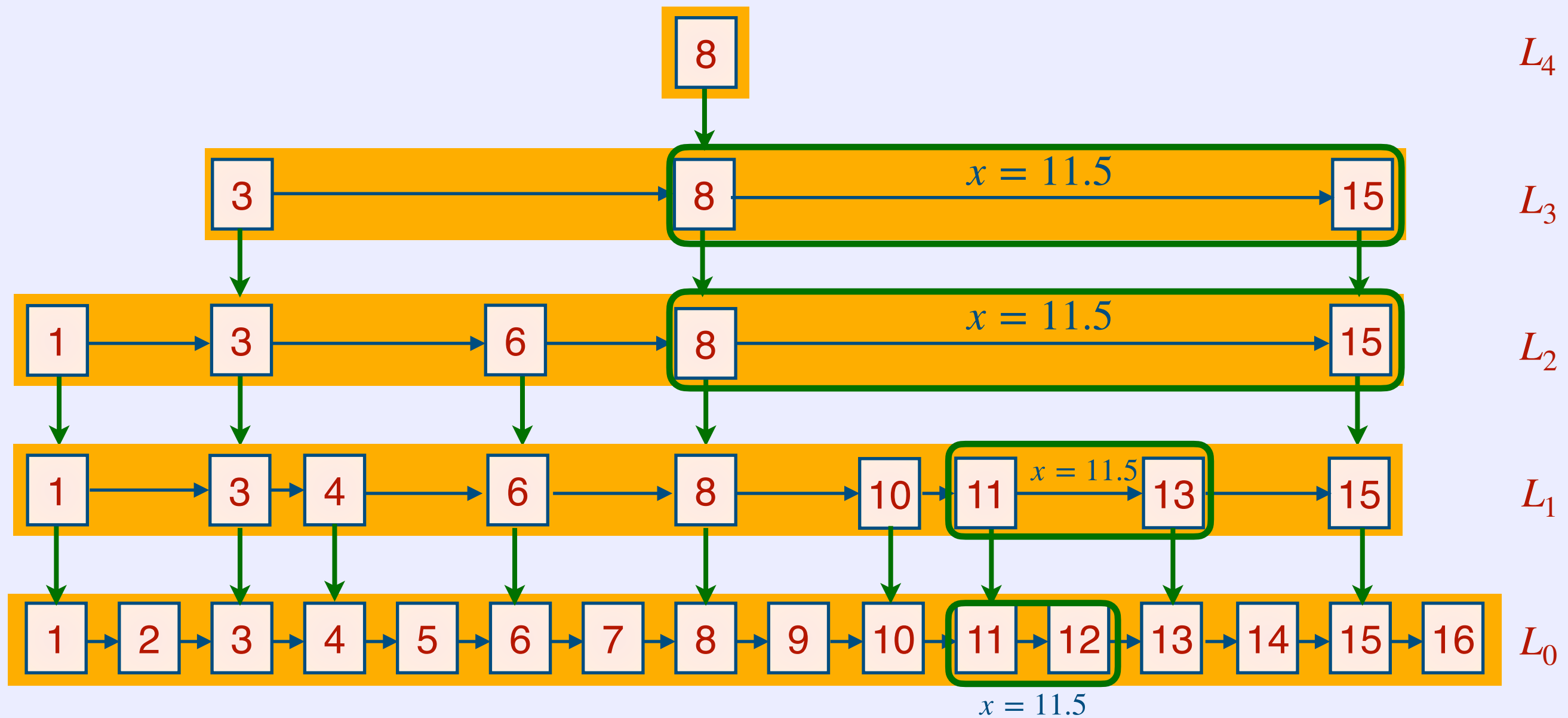
Each element of L_i is chosen in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

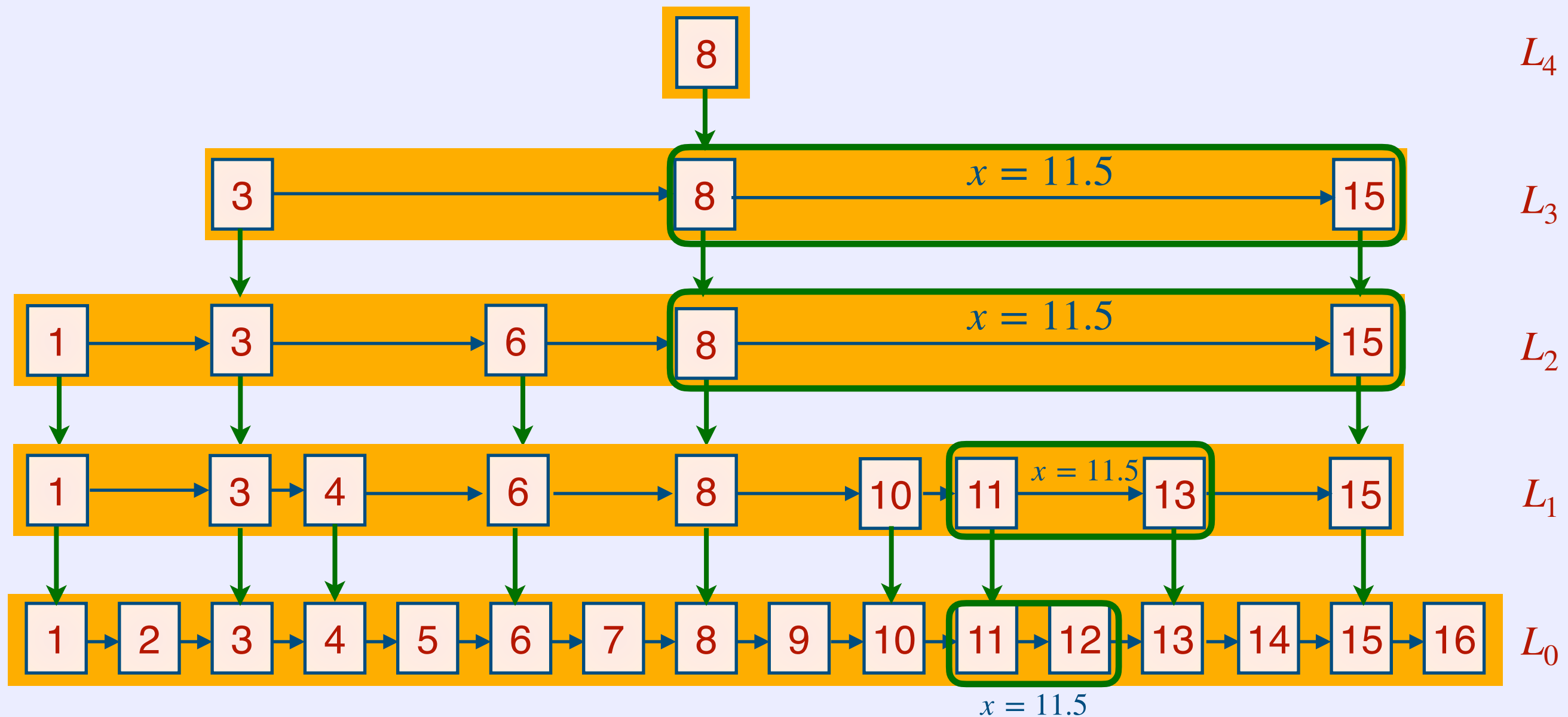
Each element of L_i is chosen in L_{i+1} with prob $1/2$.



What if we have randomness?

Generation Rule

Each element of L_i is chosen in L_{i+1} with prob $1/2$.



This data-structure is called Skip-list

Question: What is expected Search time?



Properties we will prove:

1. Number of Levels:

The number of levels is at most $5 \log_2 n$ with prob 0.9999

2. Time spent per level:

The expected time taken by Search/Insert/Delete at each level is $O(1)$

3. Total time:

Expected total time taken by Search/Insert/Delete operations is $O(\log_2 n)$

Number of levels in Skip List

Consider an element $x \in L_0$

Let $K = 5 \log_2 n$

Probability that x is present in list $L_1, L_2, \dots, L_K$ is:

$$\frac{1}{2^K} = \frac{1}{n^5}$$

Let $x_1, \dots, x_n$ be elements of L_0

For any $i \in [1, n]$,

$$\text{Prob}(x_i \in L_K) = \frac{1}{n^5}$$

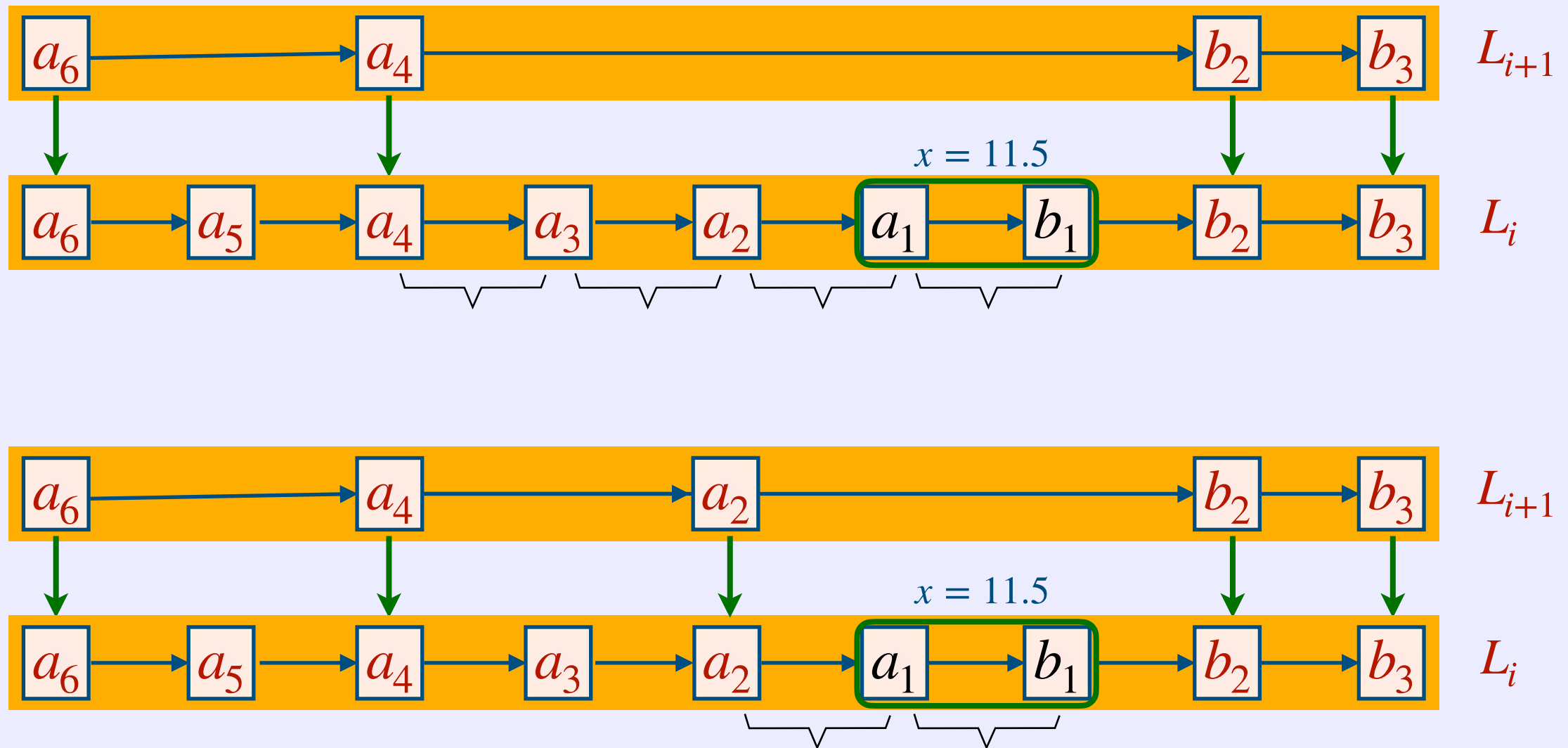
By union bound, probability that there exists an x_i in level L_k is **at most**:

$$n \cdot \frac{1}{n^5} = \frac{1}{n^4}$$

Lemma 1: If $n \geq 10$, then probability that the number of levels is more than $5 \log_2 n$ is at most $(1 - 1/n^4) \leq 0.9999$.

Search in Skip List at level 'i'

Consider following two cases wherein list L_i is identical but L_{i+1} is different.



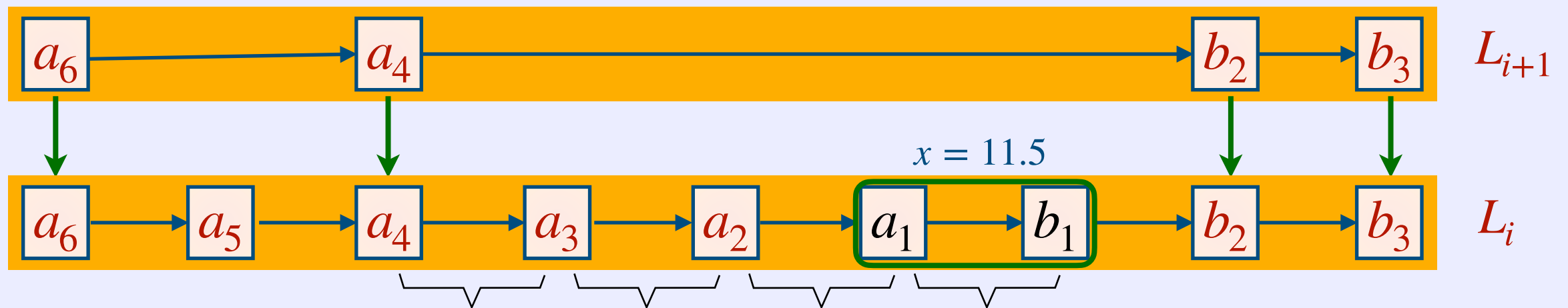
Note: For any $x_i \in [a_1, b_1]$, $\text{Search}(x, L_i)$ will take **lesser time** in the second case.

Search in Skip List at level 'i'

Lemma 2:

Suppose $L_i = (a_r, \dots, a_2, a_1, b_1, b_2, \dots, b_s)$, and let x be a query element in $[a_1, b_1]$.

If j is **smallest** index such that a_j lies in L_{i+1} , then time taken by $\text{Search}(x, L_i)$ is $O(j)$.

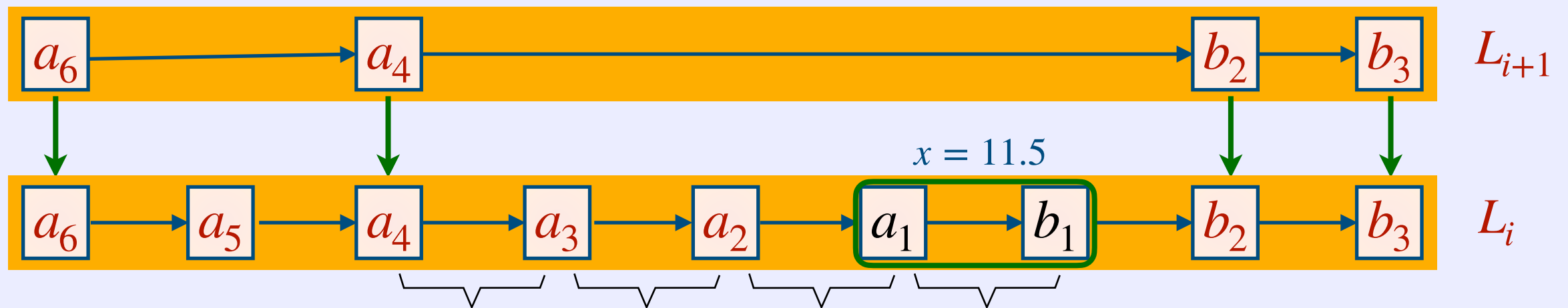


Search in Skip List at level 'i'

Lemma 2:

Suppose $L_i = (a_r, \dots, a_2, a_1, b_1, b_2, \dots, b_s)$, and let x be a query element in $[a_1, b_1]$.

If j is **smallest** index such that a_j lies in L_{i+1} , then time taken by $\text{Search}(x, L_i)$ is $O(j)$.



Homework:

1. What is expected value of ' j ' ?
2. What is expected time taken by Search operation in list L_i ?