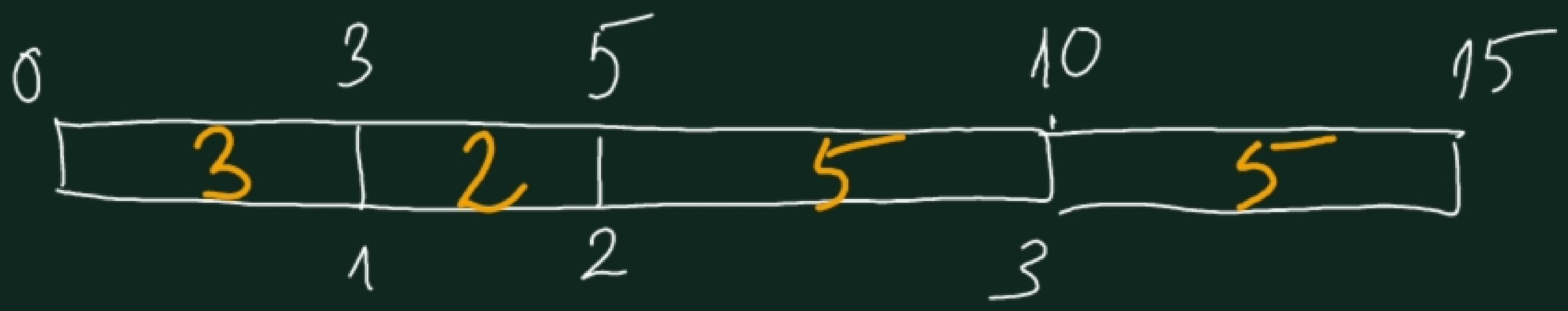
Rod cutting problem

We are given a rod with length ℓ and values

$0 < x_1 < x_2 < x_3 < \ldots < x_{n-1} < \ell$

We have to cut the rod into n pieces at the marks x_i. The cost of a cut equals the length of the rod being cut.

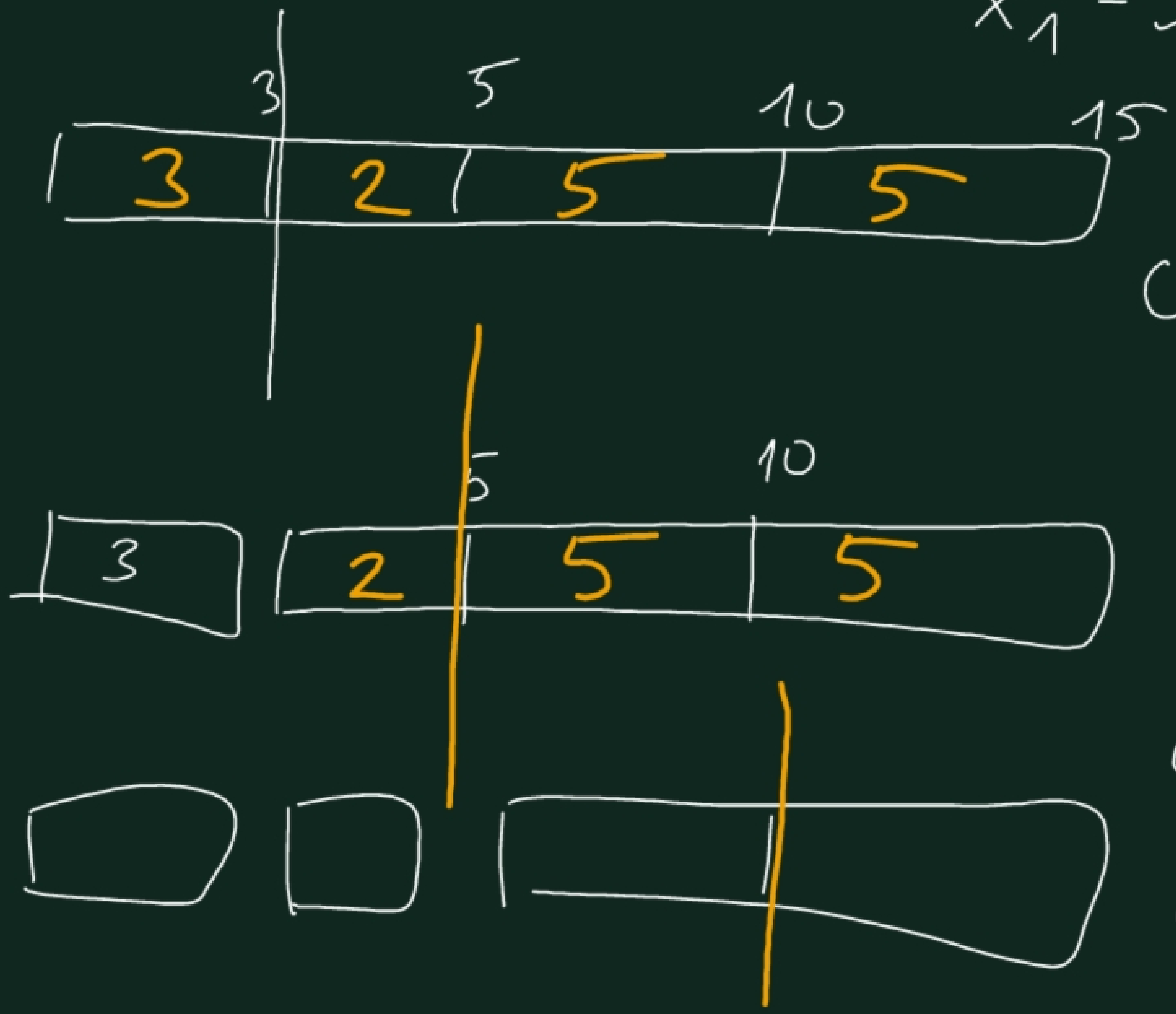
We would like to minimize total cost.



$\ell = 15$

$x_1 = 3 \quad x_2 = 5 \quad x_3 = 10$

$x_1 \to x_2 \to x_3$



cost = 15

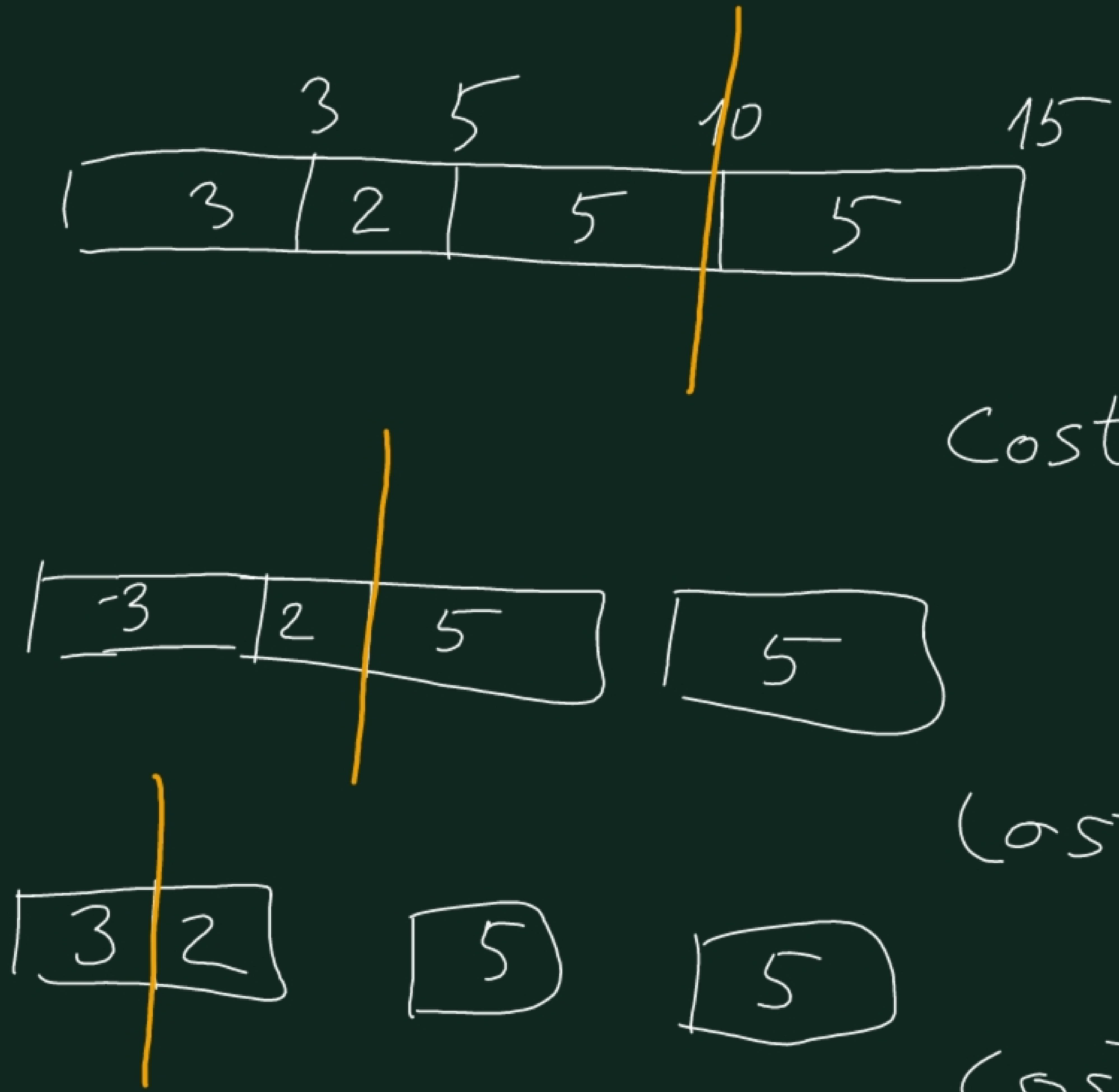
cost = 12

cost = 10

$\underline{\underline{37}}$

$$x_3 \to x_2 \to x_1$$

$$\{1, 2, 3, \ldots, n-1\}$$



Cost = 15

Cost = 10

Cost = 5

$\underline{\underline{30}}$

Find the best cutting order!

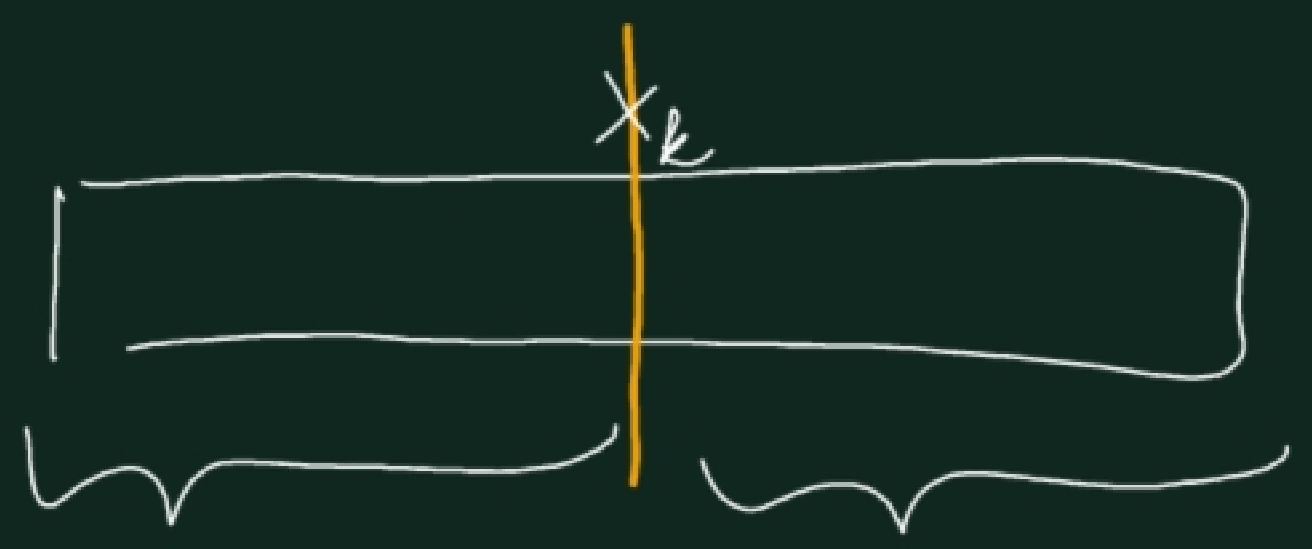
$\rightarrow$ Check all orders?

$n-1$ cuts to do $\quad (n-1)! \approx O(n^n)$

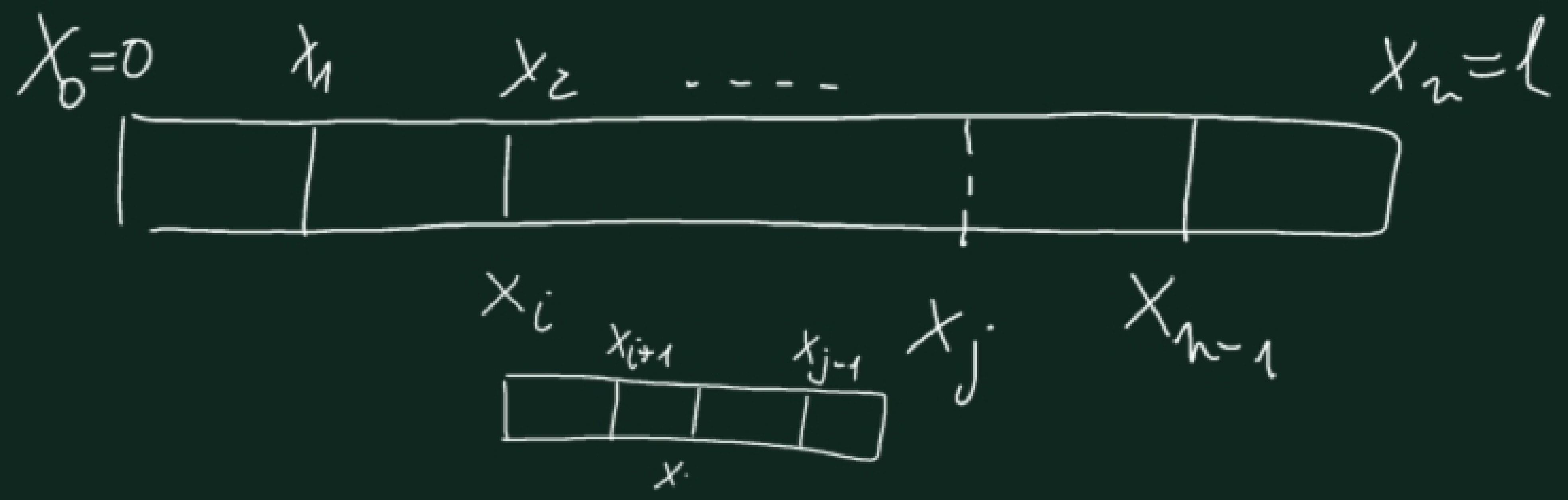
DP algorithm!

Similarity with matrix chain mult. problem

$$(A_1 A_2 \cdots A_k)(A_{k+1} \cdots A_{n-1} A_n)$$



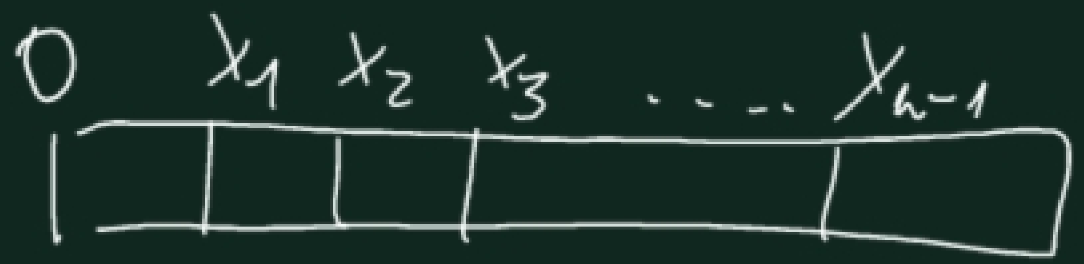
Subproblem:

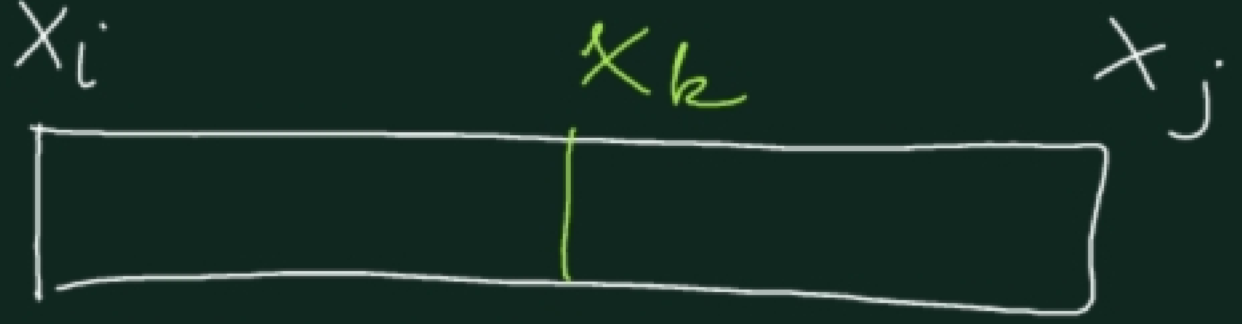


$S[i,j]$ = the subproblem of cutting the piece of rod between x_i and x_j with minimal cost

$C[i,j]$ = this optimal cost

$c[i,j] = 0$ if $j \leq i+1$



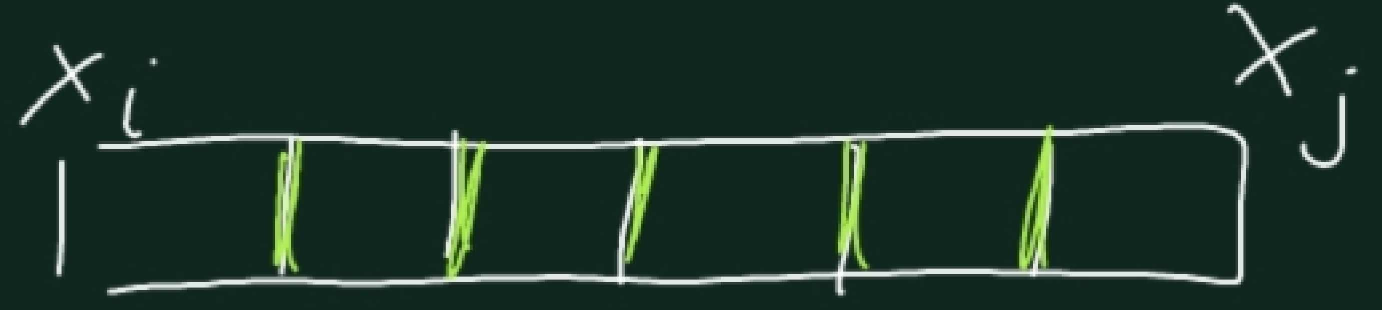


$j \geq i+2$

* in the optimal solution

STATEMENT

If* the first cut is made at x_k $i<k<j$ then the rest of the cuts make an optimal cost cutting in $S[i,k]$ and $S[k,j]$.

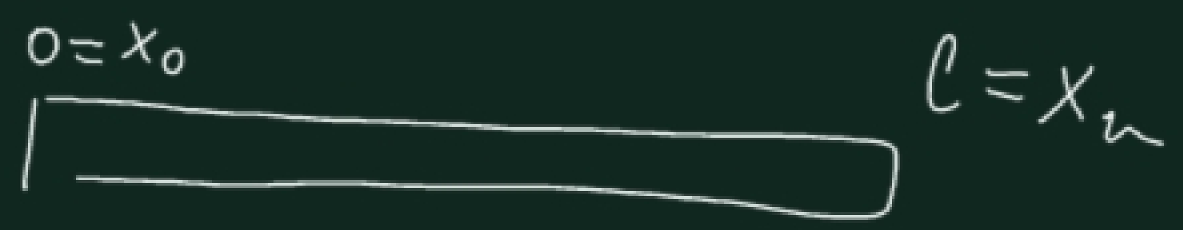


Recursive formula

$$c[i,j] = \underbrace{x_j - x_i}_{\substack{\text{cost of} \\ \text{the first cut}}} + c[i,k] + c[k,j] \qquad i < k < j$$

We don't know where this first cut is made ($k = ?$), so we look at this for all k values & select the one that gives minimal cost.

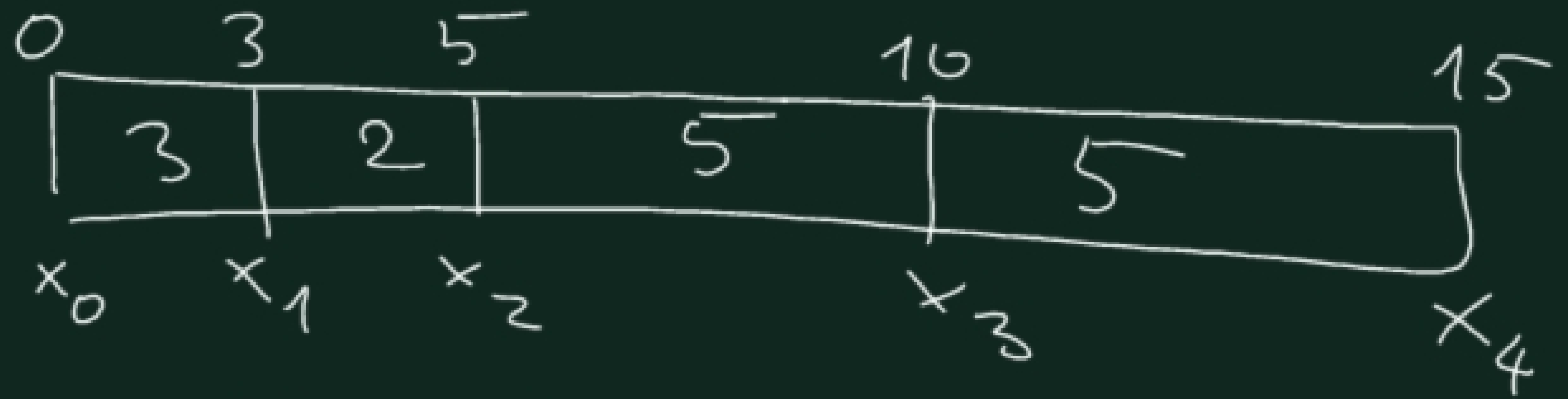
$$C[i,j] = \begin{cases} 0 & \text{if } j \le i+1 \\ \min\limits_{i<k<j} \left(x_j - x_i + C[i,k] + C[k,j] \right) & \text{if } j > i+1 \end{cases}$$



$(n+1) \times (n+1)$

	0	1	2	3	4
0	0	0	5	15	30
1	0	0	0	7	19
2	0	0	0	0	10
3	0	0	0	0	0
4	0	0	0	0	0

$$\begin{array}{ccccccccc} x_0 & & x_1 & & x_2 & & x_3 & & x_4 \\ 0 & < & 3 & < & 5 & < & 10 & < & l = 15 \end{array}$$



$$x_2 - x_0 + \underset{0}{C[0,1]} + \underset{0}{C[1,2]}$$

$$\underbrace{x_3 - x_0}_{10} + \min \begin{cases} \underset{0}{C[0,1]} + \overset{7}{C[1,3]} \\ \underset{5}{C[0,2]} + \underset{0}{C[2,3]} \end{cases} = 15$$

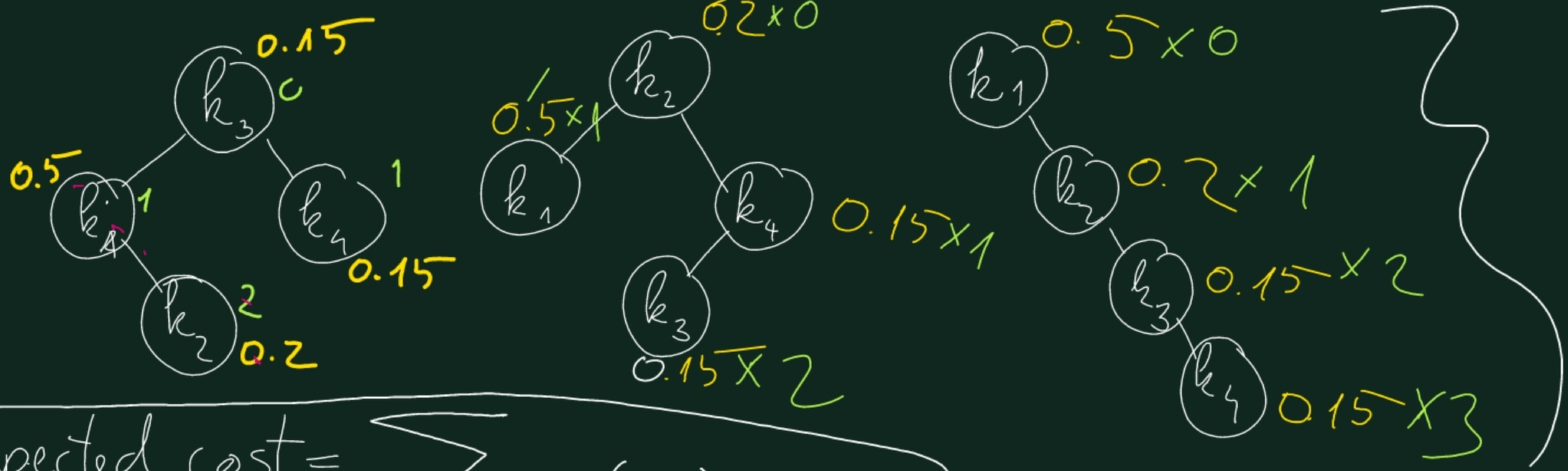
Optimal binary search tree

We are given a set of key values $k_1 < k_2 < k_3 < \ldots < k_n$ and probabitities $p_1, p_2, p_3, \ldots, p_n$ such that $\sum p_i = 1$ and k_i is being searched with probability p_i. Build a binary search tree where the expected search cost is minimal.

$k_1 < k_2 < k_3 < k_4$

$0.5 \quad 0.2 \quad 0.15 \quad 0.15$

$c(k_i)$ = cost of searching k_i

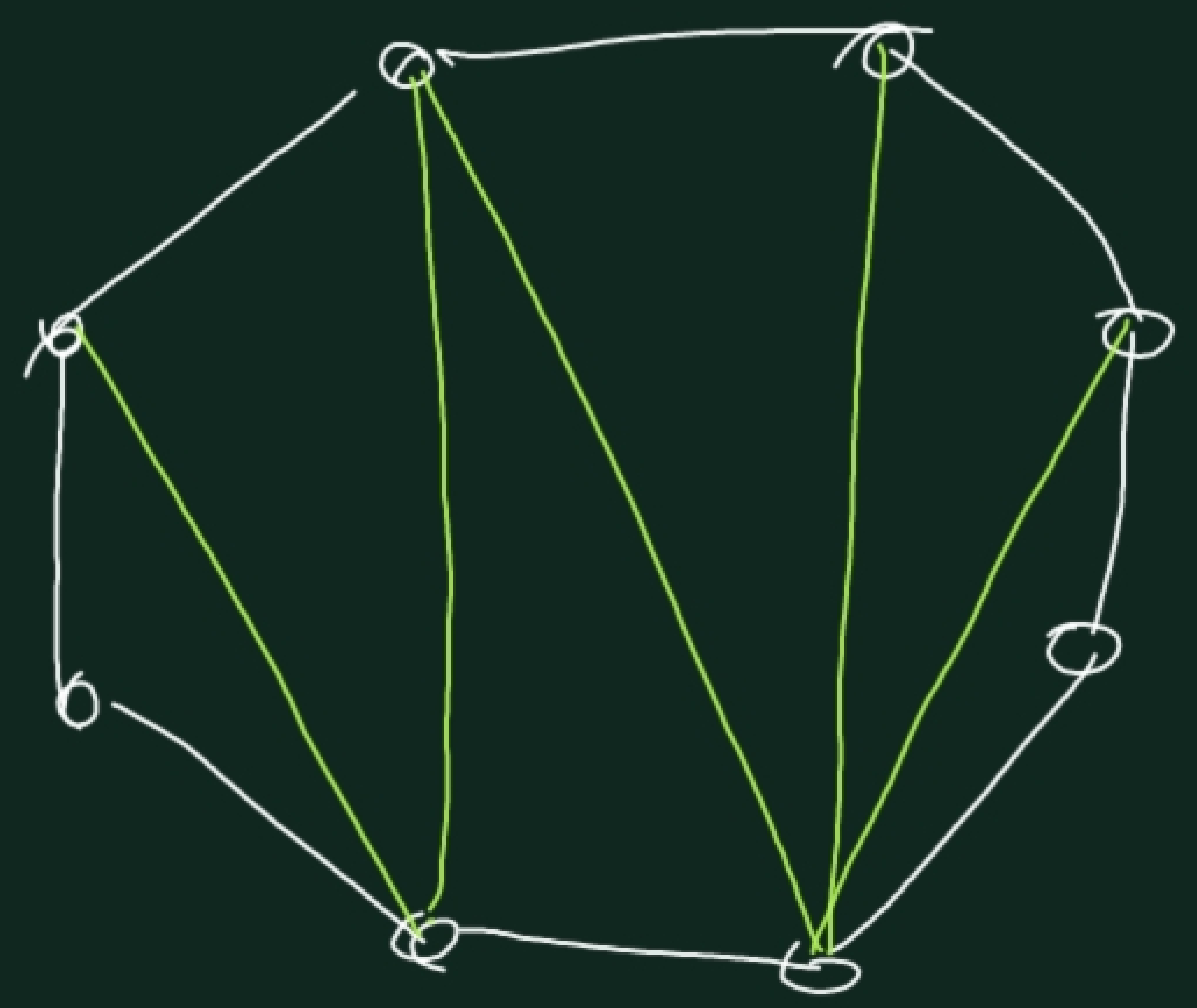


$$\boxed{\text{expected cost} = \sum c(k_i) \times p_i} \rightarrow \min$$

STATEMENT:

If we have an optimal binary tree, and k_j is in the root of this tree, then we have $k_1 < k_2 < \dots < k_{j-1}$ in the left subtree and $k_{j+1} < k_{j+2} < \dots < k_n$ in the right subtree, and these subtrees are optimal BSTs for these keys.

Optimal polygonal triangulation problem



cost of a triangle

minimize the sum of the costs

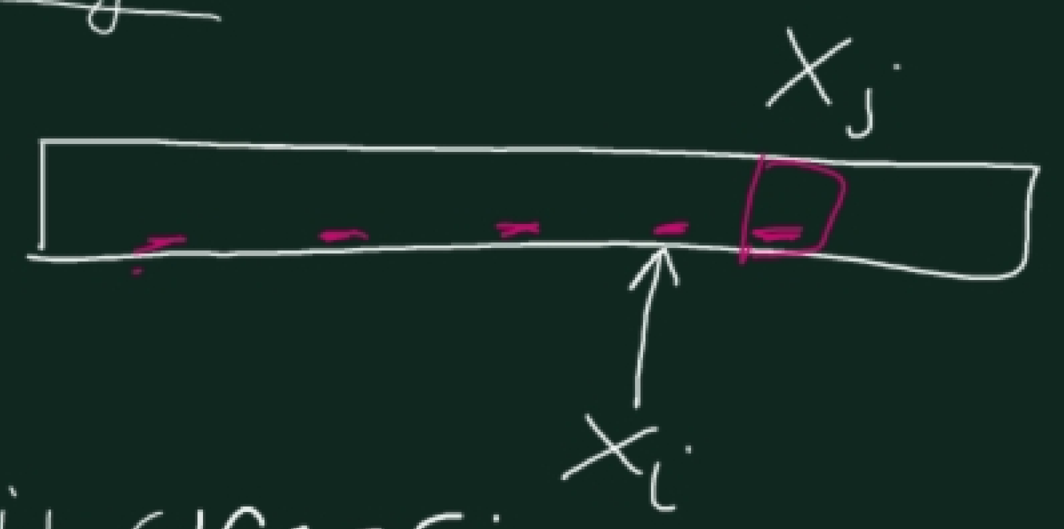
Longest increasing subsequence problem:

Find the longest increasing subsequence in a sequence of numbers $x_1, x_2, x_3, \dots, x_n$

$$\boxed{2}\ 1\ \boxed{4}\ 3\ \boxed{6}\ 5\ \ 5.5$$

$S[j]$ = is the subproblem of finding a longest increasing subsequence <u>ending with x_j.</u>

$x_1, x_2, \ldots, x_j$



<u>**Statement:**</u> if l is a longest increasing subsequence ending with x_j, and in this subsequence x_i is the element before x_j then

$l \setminus \{x_j\}$ is an optimal solution for $S[i]$

(a LIS ending with x_i)

$c[j]$ the length of the opt. solution for $S[j]$

Recursive formula:

$$c[j] = \max \{ c[i] + 1 \}$$

for all i with $1 \le i < j$ and $x_i < x_j$