$$= \sum_{i=1}^{n} \text{depth}_T(k_i) \cdot p_i + \sum_{i=1}^{n} p_i$$

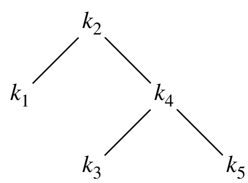
$$= 1 + \sum_{i=1}^{n} \text{depth}_T(k_i) \cdot p_i \qquad \text{(since probabilities sum to 1)} \qquad (*)$$

[Keep equation (∗) on board.]

[Similar to optimal BST problem in the textbook, but simplified here: we assume that all searches are successful. Textbook has probabilities of searches between keys in tree.]

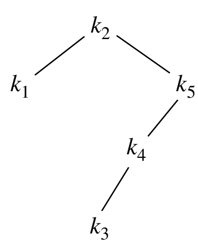
Example

i	1	2	3	4	5
p_i	.25	.2	.05	.2	.3



i	$\text{depth}_T(k_i)$	$\text{depth}_T(k_i) \cdot p_i$
1	1	.25
2	0	0
3	2	.1
4	1	.2
5	2	.6
		1.15

Therefore, $\text{E}[\text{search cost}] = 2.15$.



i	$\text{depth}_T(k_i)$	$\text{depth}_T(k_i) \cdot p_i$
1	1	.25
2	0	0
3	3	.15
4	2	.4
5	1	.3
		1.10

Therefore, $\text{E}[\text{search cost}] = 2.10$, which turns out to be optimal.