Skiplist

In this project we deal with the skiplist, a data structure which is used to quickly find a key (e.g. a telephone number) in a list and display its value (e.g. name of the telephone owner). Good description of skiplist can be found on the web, for example in Wikipedia or simply search them with Google. We start with a quick description of skiplist and how they are created, mainly to introduce the notation that we will need.

Suppose we have n key values we would like to insert in a skiplist. We order the values and create a list with n nodes, each one with a key. This is the level 0 in the skiplist. To create the next level, we flip a fair coin for each of the nodes in the previous level. If we get head, then we copy the node to the next level, otherwise we skip the node. So, with probability 1/2 we copy the node to the level 1. Similarly, each node at level i will be copied to the level $i + 1$ with probability 1/2. This procedure is repeated until the next level is empty. Moreover, each level starts with $-\infty$ and ends with $+\infty$. Figure 1 gives an example of one of all the possible skiplist with keys 2,5,10,13,18,21,24,25.

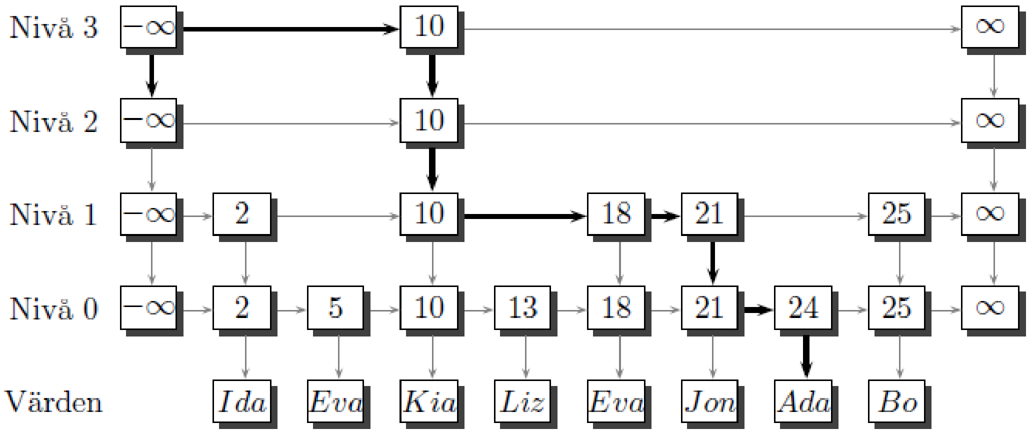

Figure 1: Example of a Skiplist with keys 2,5,10,13,18,21,24,25.

1. Number of nodes per level:

   (a) What is the probability that the first node reaches level i or higher?

   (b) If one knows that k nodes has reached level i, in how many ways can one choose these nodes?

   (c) What is the probability that there are exactly k nodes at level i, where $k = 0, 1, \ldots, n$ and $i = 0, 1, \ldots,$ if one does not count the ending nodes with $-\infty$ and $+\infty$?

2. Worst case;

   The worst thing that can happen in a skiplist, with respect to time spent in a search, is that all the nodes reach the same level.

(a) What is the probability that the first number reaches level i but not higher?

(b) What is the probability that all the n nodes reach level i but not level $i+1$, for $i = 0, 1, \ldots, ?$

(c) What is the probability that all nodes reach the same level? Simplify the expression as much as possible. Hint: If every node reaches the same level, it means that no node reach level 1, or that every node reach level 1 but not level 2, or that every node reach level 2 but not 3, etc. Use (a) and the fact that "or=$\cup$ (union)"

3. Memory space

(a) Let X denote the number of nodes at level i. In point 1 you have computed the probability that $X = k$. What is the distribution of the random variable X and why? What are the parameter of the distribution?

(b) What is the expected number of nodes at level i for $i = 0, 1, 2, \ldots,$ without counting the ending nodes? Hint: use the expected value of the distribution of X.

(c) What is the expected total number of nodes in the skiplist, without counting the ending nodes? Simplify the expression as much as possible.

4. Skiplist height

(a) Let Y denote the height, i.e. the maximum level, of the first digit stack (in the example in figure 1 the first digit stack has height 1). In task 2(a) you have computed the probability that $Y = i$. What is the distribution of the random variable $X = Y + 1$ and why? What are the parameters of the distribution?

(b) Explain why the height H_n in a skiplist consisting of n numbers can be described as $H_n = \max\{X_1, \ldots, X_n\} - 1$ where $X_1, \ldots, X_n$ are independent random variables with the same distribution as in the task 4(a).

(c) Find the cumulative distribution function F_{H_n} of H_n. (Hint: $\max\{X_1, \ldots, X_n\} \leq k$ is equivalent to $X_1 \leq k, \ldots, X_n \leq k$.)

(d) Find the density function f_{H_n} of H_n. (Hint: for integer valued random variables it holds $f(k) = F(k) - F(k-1)$ if k is an integer.)

(e) Use the formula $\mathbb{E}[H_n] = \sum_{k=1}^{\infty} k f_{H_n}(k)$ and a computer to approximate the expected value $\mathbb{E}[H_{2^m}]$ for $m = 1, 2, ..., 7$. Do you see any pattern? Use that to motivate that $\mathbb{E}[H_n] \approx \log_2 n$. (Let $n = 2^m$ so that $m = \log_2 n$.) (An explicit expression for $\mathbb{E}[H_n]$ does not exist for all n, only an asymptotic expression is available.)

Optional questions:

5. When we search a key (and its value) we always start from the node $-\infty$ at the top of the leftmost stack. Then, we follow the arrows to the right as much as possible

without passing the number we seek. When we can't go the the right anymore because the next number is bigger than the one we seek, we go down a level. The bold arrows in the figure show the search path for the number 24.

(a) Let Z_i be the number of nodes we visit on each level. What is the expected value $\mathbb{E}[Z_i]$? Hint: imagine we follow the arrows backwards at level i. Then we will go straight to the left on the same level until we find a number in both level i and level $i+1$. For each number in level i (including the one where we start) the probability that the number is on level $i+1$ as well is $1/2$ (if we obtained head in the coin toss). Z_i is then the number of independent tosses needed before one head turns up!

(b) Try to explain using tasks 4(e) and 5(a) above why the expected number of nodes we visit is $\approx 2\log_2 n$. Obs: the information from 4(e) and 5(a) is not enough to make this argument rigorous, as H_n and Z_i are not independent. It follows that the time spent for a search is asymtptically $O(\log_2 n$. The time spent for inserting and deleting numbers from the list is of the same order, but we will not show it here.