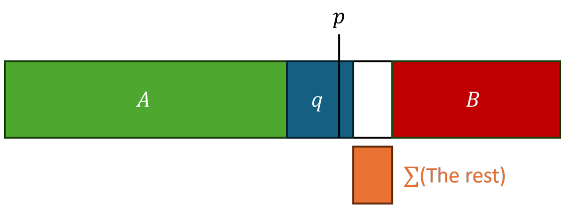
Problem:

> You have a biased 6-sided die with known probabilities for each face.
> You must stop throwing the die once a face that differs from your first roll comes up.
> You're told a probability p.
> How can you use rolls of this die to simulate a biased coin with probability p of heads?
>
> \- Mikhail on Twitter

Sequences of the form $ii\dots j$ with $i,j \in \{1,\dots,6\}$ are the possible outcomes. The idea is to partition the set of all possible outcomes M into two sets A and B, such that the probabilities in A sum to p. Then we throw the die. If the resulting sequence is in A, we output *HEAD*, else *TAILS*. To contruct A, we can sort M by probability, descending. Take the first (i.e. highest probability) outcome from M and put it into A if it fits without overflowing p, otherwise put it into B. Could it happen that the outcome does not fit into either A or B without crossing p? That could only happen if the probability q of that outcome is higher than the sum of all following outcomes:



$$q > \sum_{\substack{m\in M \\ P(m)<q}} P(m) \tag{1}$$

There should be a one-line proof that this can not happen, but I did not find it. Instead I will bound the right side of inequality 1 from below. All possible outcomes have the form $ii\dots j$ and occur with probability $p_i^n p_j$. By piling on p_i's we cross q. The worst case happens when p_i^n is just short of q and we need to go to $p_i^{n+1} \approx qp_i$, which can happen for all six faces at once. These six probabilities form the trunks from which all probabilities smaller than q sprout. For each additional throw of the dice we can either get the same face with probability p_i and continue or a different face with probability $1 - p_i$ and stop. Summing over all outcomes and all trunks gives

$$
\begin{aligned}
\sum_{\substack{m \in M \\ P(m) < q}} P(m) &> \sum_i \sum_{n \geq 1} q p_i^n (1 - p_i) \\
&> q \sum_i (1 - p_i) \sum_{n \geq 1} p_i^n \\
&> q \sum_i (1 - p_i) \left(\frac{1}{1 - p_i} - 1 \right) \\
&> q \sum_i p_i \\
&> q
\end{aligned} \tag{2}
$$

Putting the inequalities together we get $q > q$, a contradiction. Building the sets A and B never results in a dead end. Since the sets are infinite, we cannot build them completely in a computer. To make an algorithm, first throw the dice and record the sequence - Then build the sets A and B, until the actual sequence is put into A or B. Return *HEAD* or *TAILS* accordingly.