Using a Huffman tree to generate discrete rv:s

- Store the distribution in a Huffman tree, so that the most probable candidates have the shortest path.
- The Huffman tree is created with the following algorithm:
  1. Create an ordered list of nodes. The elements in the list represent values of the rv and are ordered by probability. These nodes will be the leaves in the tree.
  2. Take the first two items from the list (two smallest probabilities) and create a parent node for them so that the smaller child is on the left. The probability of the new node will be the sum of the children. The new node is added to the list in the right position.
  3. Repeat the previous step until the list has only one element, which is the root of the tree.
- Now the sample generation goes as follows:
  1. Generate a random number $U \sim U(0,1)$ and compare it to the probability of the left child.
  2. If the value is smaller, move to the left child. If it is greater, subtract the probability of the left child from the random number and move to the right child. Compare the random number to the probability of the new left child.
  3. Repeat the previous step until the current node is a leaf.

```
value     f(x)     F(x)
  1       0.10     0.10
  2       0.30     0.40
  3       0.20     0.60
  4       0.15     0.75
  5       0.25     1.00

Iteration 1
-----------
List: (0.1, 1), (0.15, 4), (0.20, 3), (0.25, 5), (0.30, 2)
Tree:
         0.25:*
         /     \
     0.10:1   0.15:4

Iteration 2
-----------
List: (0.20, 3), (0.25,*), (0.25, 5), (0.30, 2)
Tree:
         0.45:*
         /    \
     0.20:3  0.25:*
             /     \
          0.10:1  0.15:4

Iteration 3
-----------
List: (0.25, 5), (0.30, 2), (0.45,*)
Tree:
          0.55:*
         /     \
      0.25:5  0.30:2

Iteration 4
-----------
List: (0.45,*), (0.55,*)
Tree:
              _____1.0:*______
             /                \
         0.45:*             0.55:*
         /    \             /      \
     0.20:3  0.25:*      0.25:5   0.30:2
             /     \
          0.10:1  0.15:4
```