Problem 1

Firstly, calculate h(key) by the hash function given. Secondly, put the key value in the hash table correspondingly. If the position is occupied, put key value in the next position until the position is available. Probe sequence is the position that might be available. Finally if the position in hash table is not enough, the remaining values will not be filled in table.

Key Value	**h(key)**	**Probe Sequence**
43	0	0
23	6	6
1	3	3
0	1	1
15	7	7
31	2	2
4	9	9
7	5	5
11	5	5, 6, 7, 8
3	7	7, 8, 9, 10
5	0	0, 1, 2, 3, 4
9	10	10, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, overflow

	Final Hash Table Contents
0	43
1	0
2	31
3	1
4	5
5	7
6	23
7	15
8	11
9	4
10	3

Problem 2

Result in COE Linux:

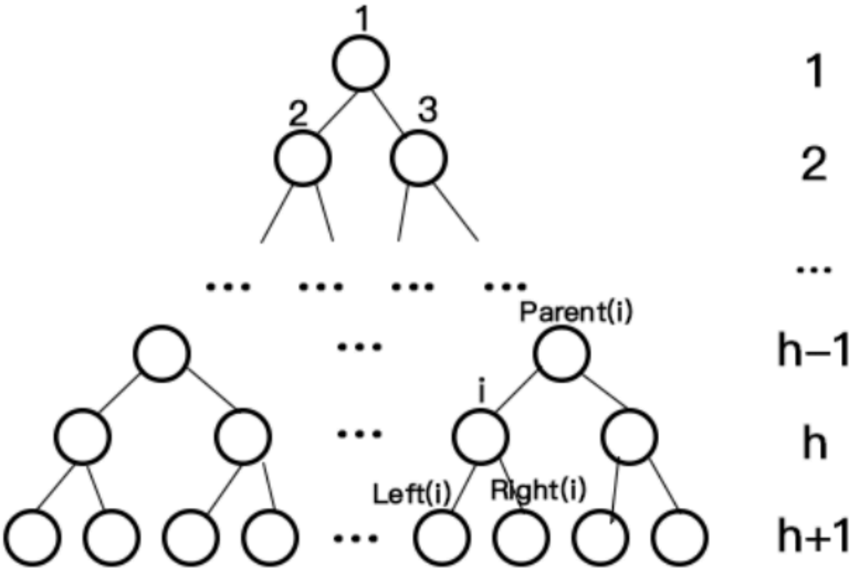
```
For m = 64
MAX:32 MIN:6 MEAN:15.625 VARIANCE:48.2969

For m = 66
MAX:25 MIN:4 MEAN:15.1515 VARIANCE:26.3104

For m = 67
MAX:28 MIN:4 MEAN:14.9254 VARIANCE:28.8153

For m = 59
MAX:26 MIN:8 MEAN:16.9492 VARIANCE:17.6415
```

Compare different p:

```
For m = 61
MAX:32 MIN:7 MEAN:16.3934 VARIANCE:22.1403
For m = 59
MAX:29 MIN:7 MEAN:16.9492 VARIANCE:20.794
For m = 53
MAX:28 MIN:11 MEAN:18.8679 VARIANCE:21.6995
For m = 47
MAX:29 MIN:12 MEAN:21.2766 VARIANCE:20.7533
For m = 43
MAX:35 MIN:14 MEAN:23.2558 VARIANCE:17.6322
For m = 41
MAX:37 MIN:14 MEAN:24.3902 VARIANCE:24.1404
For m = 37
MAX:43 MIN:18 MEAN:27.027 VARIANCE:25.5398
```

I pick p as 59, because comparing with 61, 53, 47,43,41 and 37, its mean and variance are relatively small. Comparing with 64, 66 and 67, 59 is far away from 2^8. Also when m = 59, the hash table has good performance on variance, and its mean is not bad.
From the results we can find that larger the hash table size is, smaller its mean and variance are, and correspondingly lower the collision rate is. But if slot is close to 2^n, collision rate will rise.

Problem 3



Suppose i is the a^{th} element of the h^{th} line, and

$$i = (2^0 + 2^1 + \cdots + 2^{h-1}) + a = 2^h - 1 + a$$

In $(h+1)^{th}$ line, there are $2 * (a - 1)$ elements in front of $Left(i)$, so

$$Left(i) = (2^0 + 2^1 + \cdots + 2^h) + 2 * (a - 1) + 1$$

$$= 2^{h+1} - 1 + 2a - 2 + 1 = 2^{h+1} - 2 + 2a = 2 * (2^h - 1 + a) = 2i$$

And

$$Right(i) = Left(i) + 1 = 2i + 1$$

In $(h-1)^{th}$ line, there are $(a - 1)/2$ elements in front of $Parent(i)$ when i is left child of $Parent(i)$, and $(a - 2)/2$ elements in front of $Parent(i)$ when i is right child of $Parent(i)$, so

$$Parent(i)$$

$$= \begin{cases} (2^0 + 2^1 + \cdots + 2^{h-2}) + \dfrac{(a - 1)}{2} + 1 = 2^{h-1} - 1 + \dfrac{a}{2} + \dfrac{1}{2} = \dfrac{(2^h + a - 1)}{2} = i/2 \\ (2^0 + 2^1 + \cdots + 2^{h-2}) + \dfrac{(a - 2)}{2} + 1 = 2^{h-1} - 1 + \dfrac{a}{2} = \dfrac{(2^h + a - 1)}{2} - \dfrac{1}{2} = i/2 - 1/2 \end{cases}$$

When i is right child, i is odd number. And because index is integer, $i/2 - 1/2 = i/2$. So

$$Parent(i) = i/2$$