1. Note: for key value 9, it can not be inserted into the hash table as the hash table is full.

Key Value	Probe Sequence
43	0
23	6
1	3
0	1
15	7
31	2
4	9
7	5
11	5
3	7
5	0
9	10

Index	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

2. For this question, I took the following steps to implement my code:

1. Generated 1000 random integers between January 1, 2020 and December 31, 2004: For each integer, I used the rand() function to generate an integer between 0-11 then added 1 and multiplied by 10,000 to get a random integer between 10,000 and 120,000 to represent the month. I did something similar to generate day (100-2800) and year (00-04). I then added the month, day, and year. I repeated this until the array was full.
2. I implemented a function for hashing the keys ($h(k)=k \bmod m_i$).
3. I simulated 4 hash tables of different sizes using an array. I simulated the hash table that uses chaining to resolve collisions by adding 1 to the array contents at position h(k), k being the key value that was randomly generated. I iterated through all the generated data and did this.
4. To calculate the minimum and maximum, I iterated through the collision values (x-1, x being the value stored in my "hash table") and found them. While doing so, I summed all the collision values, dividing by the size of the table after the loop to get the mean of the collisions. To find the variance, I looped through the collision values again and summed the squared difference (difference being difference between the collision value and mean) then divided by the size of the hash table after the loop finished to get variance.

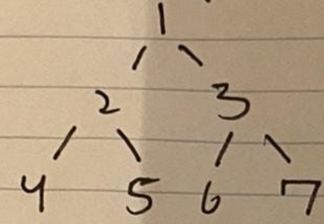
Looking at the values generated by my simulation, it seems as though an increased size of the hash table decreases the average number of collisions. It is hard to tell if it affects other variables because of how few samples were taken. Variance and the maximum number of collisions seem to be correlated. The following are the results from the hash table experiment:

i	m_i	Minimum Collisions	Maximum Collisions	Mean	Variance
1	97	1	19	9.31	14.57
2	98	0	24	9.20	36.69
3	100	0	208	9.00	1901.20
4	101	0	48	8.90	211.00

3. Proof by induction:

Proof by Induction

1. Base case: $i=2 \Rightarrow parent=1 \wedge left=4 \wedge right=5$



visually, we see at $i=2$, this is true.

2. assume for $i=k$, $\begin{cases} parent(k) = \lfloor k/2 \rfloor \\ left(k) = 2k \\ right(k) = 2k+1 \end{cases}$

3. Prove for $i=k+1$, $\begin{cases} parent(k+1) = \lfloor (k+1)/2 \rfloor \\ left(k+1) = 2(k+1) \\ right(k+1) = 2(k+1)+1 \end{cases}$

as the next 2 elements after parent(k) are $2k+2$ and $2k+3$, the following proves left and right:

$left(k+1) = 2(k+1) = 2k+2$ ✓

$right(k+1) = 2(k+1)+1 = 2k+3$ ✓

and for parent(k+1):

$$parent(k+1) = \left\lfloor \frac{k+1}{2} \right\rfloor = \left\lfloor \frac{k}{2} + \frac{1}{2} \right\rfloor = \begin{cases} parent(k), & k \text{ is even} \\ k-1, & k \text{ is odd} \end{cases}$$

which holds true □

$\therefore$ given index i of a node, the heap indices are as follows:

$Parent(i) = \lfloor i/2 \rfloor$

$left(i) = 2i$

$right(i) = 2i+1$