PROBLEM 2

Part a)

0	
1	
2	
3	
4	
5	
6	

Start point

① insert 15

15 % 7 = 1 (success)

insert 15 in index[1]

$\alpha = \frac{1}{7}$

0	
1	15
2	
3	
4	
5	
6	

② insert 22

22 % 7 = 1 (fail)

$(1+1^2)\%7 = 2$ (success)

insert 22 in index[2]

$\alpha = \frac{2}{7}$

0	
1	15
2	22
3	
4	
5	
6	

③ insert 36

36 % 7 = 1 (fail)

$(1+1^2)\%7 = 2$ (fail)

$(1+2^2)\%7 = 5$ (success)

$\alpha = \frac{3}{7}$

0	
1	15
2	22
3	
4	
5	36
6	

④ Remove 22

22 % 7 = 1 (fail)

array[1] = 15 ≠ 22 (fail)

array[1+1] = array[2] = 22 = 22

Remove 22 and mark "R" (success)

$\alpha = \frac{3}{7}$

0	
1	15
2	R
3	
4	
5	36
6	

⑤ find 36

36 % 7 = 1

array[1] = 15 ≠ 36 (fail)

array[1+1²] = array[2] = R ≠ 36 (fail)

array[1+2²] = array[5] = 36 = 36 (success)

The 36 is found at index 5

$\alpha = \frac{5}{7}$

0	
1	15
2	R
3	
4	
5	36
6	

⑥ Insert 10.

10 % 7 = 3

0	
1	15
2	R
3	10
4	
5	36
6	

Because the load factor = $\frac{4}{7} > 0.5$

we need to resize and rehash the table.

⇓ insert 15, 36, 10 to the new hash table with size 11

insert 15 : 15 % 11 = 4

insert 10 : 10 % 11 = 10

insert 36 : 36 % 11 = 3.

0	
1	
2	
3	36
4	15
5	
6	
7	
8	
9	
10	10

2 = $\frac{3}{7}$

Part b)

① insert 15

15 % 7 = 1 (success)

insert 15 to the index[1]

$\alpha = \frac{1}{7}$

0	
1	15
2	
3	
4	
5	
6	

② insert 22

22 % 7 = 1 (failed)

(22 % 7) + (3 - 22 % 3) = 3 (success)

0	
1	15
2	
3	22
4	
5	
6	

insert 22 to the index [3]

$\alpha = \frac{2}{7}$

③ insert 36

36 % 7 = 1 (failed)

(36 % 7) + (3 - 36 % 3) = 1 + 3 = 4 (success)

0	
1	15
2	
3	22
4	36
5	
6	

insert 36 at index [4]

$\alpha = \frac{3}{7}$

④ Remove 22

22 % 7 = 1 array[1] = 15 ≠ 22 (failed)

(22 % 7) + (3 - 22 % 3) = 3 array[3] = 22 (found)

remove 22 from index [3] and flag it as "R"

0	
1	15
2	
3	R
4	36
5	
6	

$\alpha = \frac{3}{7}$

0	
1	15
2	
3	R
4	36
5	
6	

⑤ Find 36

$36 \% 7 = 1$ array[1] = 15 ≠ 36

$(36 \% 7) + (3 - 36 \% 3) = 4$ array[4] = 36 = 36 (found)

$\alpha = \frac{3}{7}$

⑥ Insert 10

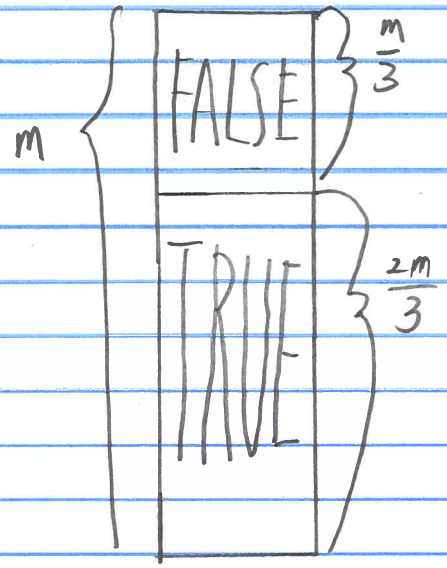
$10 \% 7 = 3$

clear the "R" flag and insert key 10 to array[3]

0	
1	15
2	
3	10
4	36
5	
6	

$\alpha = \frac{3}{7}$

2 c)



Because, 1) the three hash functions are universal and independent so the probability that $h_i(x) = p$ is exactly $\frac{1}{m}$ for all indices p

2) $h_i(x)$ is independent of $h_j(y)$ for all $i, j \in \{1, 2, 3\}$ and x, y when i does not equal j and x does not equal y

Therefore, for each hash function, it is pseudo-random, we can think that the possibility P_T that will return a True is $\frac{2}{3}$, the possibility P_F that will return a False will be $\frac{1}{3}$.

In order to let false positive to happen, all three hash functions have to return True. so the possibility for one item to have false positive is $\left(\frac{2}{3}\right)^3$.

There are 27 items, so the average false positive is

$$\left(\frac{2}{3}\right)^3 \times 27 = \underline{8}$$

PROBLEM 3

1. What test files did you use (describe them)?

Large Test	The Entire text of Hamlet
Moderate-sized Test (Uniformly Random Data)	3000 strings that are randomly generated with lower/upper case characters, numbers, special characters
Moderate-sized Test (English Text)	Paragraphs with 3000 words from the Romeo and Juliet

2. What was the capacity of your cache for each test?

Large Test	1000
Moderate-sized Test (Uniformly Random Data)	1000
Moderate-sized Test (English Text)	1000

3. What was the total number of rotations for each file?

Large Test	302917
Moderate-sized Test (Uniformly Random Data)	58152
Moderate-sized Test (English Text)	40831

4. What was the size of each file?

Large Test	32013
Moderate-sized Test (Uniformly Random Data)	3000
Moderate-sized Test (English Text)	3000

5. What was the average number of rotations per item (that is, your answer to 3, divided by your answer to 4)?

Large Test	9.46
Moderate-sized Test (Uniformly Random Data)	19.384
Moderate-sized Test (English Text)	13.61

6. How many items did you have to remove from the cache? This occurs when you bring a new item into the cache, which is referred to as a cache miss. Caches are designed in such a way as to minimize cache misses.

Large Test	22311
Moderate-sized Test (Uniformly Random Data)	2000
Moderate-sized Test (English Text)	455

7. Was there a noticeable difference between the two moderate-sized tests? Explain why there was or was not a difference.

For the two moderate-sized tests. The noticeable difference is that The Moderate-sized Test (English Text) has less number of rotations per item and less cache misses than the moderate-sized Test (Uniformly Random Data). The reason for it is that there are many duplicated words in the English text. Because the cache is implemented by the splay tree, so the most frequently used word will be closer to the top of the cache. Besides, because the most frequently used words are always closer to the top, this will also decrease the number of cache miss because the frequent visited words will never be removed and always in the cache.

8. Include any other interesting analysis your test cases revealed.

For the Moderate-sized Test (English Text), if you increase the capacity of cache, the number of cache misses will decrease a lot. For the moderate-sized Test (Uniformly Random Data), the number of remove = the number of size - the number of capacity. The reason for this is that the text is uniformly random data, so every time it will bring a new string into the cache