In the assignment, it is required to write insertion and search methods related to Cuckoo summarization. Arrays can be used when adding and searching, but the purpose of hash functions is to have access with a fixed number of steps (O(1)). Code 2 class is given to keep your access to the arrays under control. To get a value from the array, you need to use the get method of the HashTable class. The read operations performed are considered as probes, the number of probe operations at the end of the insertion and search operations must be the same as the required number. Too few or too many probes will cause the tests to fail. An example addition process and probe numbers are given in section 2.

1 Cuckoo Recap

In Cuckoo hashing, there are multiple hash tables with different hash functions. The values are located in one of the tables and can be accessed with the hash value of the relevant table. Unlike the summarization in the theoretical course, this assignment has a maximum number of attempts (MAX_REHASH_ATTEMPTS). In the assignment, it is expected to perform a two-table version of Cuckoo summarization. During the insertion process, values are swapped between the two tables until a free place is found. If the number of replacements exceeds the maximum number of attempts, the insertion will fail. In the search process, the number of probes will not exceed two since the searched value will be in one of the two tables.

2 Sample Addition Process

Cuckoo summarization will be performed with a table size (N) of 7. The summary functions of the first and second table are given below (N: table size, R: largest prime number less than N):

H1 = x mod N (1)

H2 = R − (x mod R) (2)

Values 8, 10, 2, 99, 3, 85, 47, 6, 71 will be added to the table respectively. The hash values of the numbers to be added are given in Table 1. The maximum number of probes that can be performed when adding a value is five(5) for this example. Adding the last value, 71, fails because the number of probes made in both tables is 5 and a suitable location to place all the values cannot be found. Summary tables and probe numbers resulting from the addition operations are given in Tables 2 to 11.

Table 1: Hash values of the numbers to be added. H1 = x mod N, H2 = R − (x mod R)

Key	H1	H2
8	1	2
10	3	5
2	2	3
99	1	1
3	3	2
85	1	5
47	5	3
6	6	4
71	1	4

Table 2: Start

i	T1	T2
0		
1		
2		
3		
4		
5		
6		
Sonda	0	0

Table 3: 8 additions

i	T1	T2
0		
1	8	
2		
3		
4		
5		
6		
Sonda	1	0

Table 4: 10 additions

i	T1	T2
0		
1	8	
2		
3	10	
4		
5		
6		
Sonda	2	0

Table 5: 2 additions

i	T1	T2
0		
1	8	
2	2	
3	10	
4		
5		
6		
Sonda	3	0

Table 6: 99 additions

i	T1	T2
0		
1	99	
2	2	8
3	10	
4		
5		
6		
Sonda	4	1

Table 7: 3 additions

i	T1	T2
0		
1	99	
2	2	8
3	3	
4		
5		10
6		
Sonda	5	2

Table 8: 85 additions

i	T1	T2
0		
1	85	99
2	2	8
3	3	
4		
5		10
6		
Sonda	6	3

Table 9: 47 additions

i	T1	T2
0		
1	85	99
2	2	8
3	3	
4		
5	47	10
6		
Sonda	7	3

Table 10: 6 additions

i	T1	T2
0		
1	85	99
2	2	8
3	3	
4		
5	47	10
6	6	
Sonda	8	3

Table 11: 71 insertions (failed)

i	T1	T2
0		
1	8	99
2	2	3
3	10	
4		
5	47	85
6	6	
Sonda	11	5

3 Homework Classes

All assignment classes and structures are located in the AbstractCuckoo.java file.

Write your code in Cuckoo.java file.