O = occupied
C = collision

Linear Probing: h(k, i) = (h(k) + i) mod m
Numbers to insert: [41, 30, 74, 55, 68, 39, 64, 72]
Table capacity: 8
Hash function: h(k) = (3k + 4) mod 8

Insert 41: h(41) = 7, insert to index 7
Insert 30: h(30) = 6, insert to index 6
Insert 74: h(74) = 2, insert to index 2
Insert 55: h(55) = 1, insert to index 1
Insert 68: h(68) = 0, insert to index 0
Insert 39: h(39) = 1 (O), + 1 (O) = 2, + 1 (C) = 3, insert to index 3
Insert 64: h(64) = 4, insert to index 4
Insert 72: h(72) = 4 (O), + 1 (C) = 5, insert to index 5

index	element
0	68
1	55
2	74
3	39
4	64
5	72
6	30
7	41

Quadratic Probing: h(k, i) = (h(k) + i^2) mod m
Numbers to insert: [19, 29, 16, 26, 14, 24, 13, 23]
Table capacity: 8
Hash function: h(k) = (3k) mod 8

Insert 19: h(19) = 1, insert to index 1
Insert 29: h(29) = 7, insert to index 7
Insert 16: h(16) = 0, insert to index 0
Insert 26: h(26) = 6, insert to index 6

Insert 14: h(14) = 2, insert to index 2
Insert 24: h(24) = 0 (O), + 1^2 (O) = 1, + 2^2 (C) = 4, insert to index 4
Insert 13: h(13) = 7 (O), + 1^2 (C) = 0, + 2^2 (C) = 3, insert to index 3
Insert 23: h(23) = 5, insert to index 5

index	element
0	16
1	19
2	14
3	13
4	24
5	23
6	26
7	29

Double Hashing: h(k, i) = (h1(k) + i * h2(k)) mod m
Numbers to insert: [22, 14, 39, 23, 80, 53, 49, 50]
Table capacity: 8
Hash functions: h1(k) = (k) mod 8, h2(k) = ((5k + 3) mod 7) + 1

Insert 22: h(22) = 6, insert to index 6
Insert 14: h(14) = 6 (O), + 1 * h2(14) (C) = 2, insert to index 2
Insert 39: h(39) = 7, insert to index 7
Insert 23: h(23) = 7 (O), + 1 * h2(23) (O) = 6 , + 2 * h2(23) (C) = 5, insert to index 5
Insert 80: h(80) = 0, insert to index 0
Insert 53: h(53) = 5 (O), + 1 * h2(53) (O) = 0, + 2 * h2(53) (C) = 3, insert to index 3
Insert 49: h(49) = 1, insert to index 1
Insert 50: h(50) = 2 (O), + 1 * h2(50) (O) = 4, insert to index 4

index	element
0	80
1	49
2	14

3	53
4	50
5	23
6	22
7	39

Cuckoo Hashing: h1(k) for table 1, h2(k) for table 2
Numbers to insert: [9, 23, 24, 15, 87, 20, 12, 47]
Table capacity: 14 (7 for each sub-table)
Hash functions: h1(k) = (3k + 1) mod 7, h2(k) = (floor(5k / 2) + 3) mod 7

Insert 9: h1(9) = 0, insert to index 0
Insert 23: h1(23) = 0 (O), insert to index 1, h2(9) = 4 (C), insert at index 4
Insert 24: h1(24) = 3, insert to index 3
Insert 15: h1(15) = 4, insert to index 4
Insert 87: h1(87) = 3 (O), insert to index 0, h2(24) = 0 (C), insert at index 0
Insert 20: h1(20) = 5, insert to index 5
Insert 12: h1(12) = 2, insert to index 2
Insert 47: h1(47) = 2 (O), insert to index 2, h2(12) = 5, insert at index 5

index	element (first table)	element (second table)
0	23	24
1		
2	47	
3	87	
4	15	9
5	20	12
6		