Exercise 11.4-1

- Consider inserting the keys 10, 22, 31, 4, 15, 28, 17, 88, 59 into a hash table of length m = 11 using open addressing with the auxiliary hash function $h'(k) = k$. Illustrate the result of inserting these keys using linear probing, using quadratic probing with $c_1 = 1$ and $c_2 = 3$, and using double hashing with $h_1(k)$ = k and $h_2(k) = 1 + (k \bmod (m - 1))$.
- 考虑用开放寻址法将元素10, 22, 31, 4, 15, 28, 17, 88, 59 加入到长度为 m = 11 的散列表中，使用辅助散列函数$h'(k) = k$。分别展示使用线性探查，二次探查（$c_1 = 1$, $c_2 = 3$）和双重散列（$h_1(k)$ = k，$h_2(k) = 1 + (k \bmod (m - 1))$）加入散列表的过程。

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11$.
- Insert without collisions:
  - $h(10, 0) = 10$

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

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11$.
- Insert without collisions:
  - $h(10, 0) = 10$
  - $h(22, 0) = 0$

0	1	2	3	4	5	6	7	8	9	10
22										10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11$.
- Insert without collisions:
  - $h(10, 0) = 10$
  - $h(22, 0) = 0$
  - $h(31, 0) = 9$

0	1	2	3	4	5	6	7	8	9	10
22									31	10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11$.
- Insert without collisions:
  - $h(10, 0) = 10$
  - $h(22, 0) = 0$
  - $h(31, 0) = 9$
  - $h(4, 0) = 4$

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11.$
- Further inserts:
  - $h(15, 0) = 4$ – collision! $h(15, 1) = 5$

0	1	2	3	4	5	6	7	8	9	10
22				4	15				31	10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11$.
- Further inserts:
  - $h(15, 0) = 4$ – collision! $h(15, 1) = 5$
  - $h(28, 0) = 6$

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11.$
- Further inserts:
  - $h(15, 0) = 4$ – collision! $h(15, 1) = 5$
  - $h(28, 0) = 6$
  - $h(17, 0) = 6$ – collision! $h(17, 1) = 7$

primary clustering 主要群集

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28	17		31	10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11.$
- Further inserts:
  - $h(15, 0) = 4$ – collision! $h(15, 1) = 5$
  - $h(28, 0) = 6$
  - $h(17, 0) = 6$ – collision! $h(17, 1) = 7$
  - $h(88, 0) = 0$ – collision! $h(88, 1) = 1$

0	1	2	3	4	5	6	7	8	9	10
22	88			4	15	28	17		31	10

Exercise 11.4-1 Solution: Linear Probing

- $h(k, i) = (h'(k) + i) \bmod m = (k + i) \bmod 11$.
- Last insert:
  - $h(59, 0) = 4$ – collision!
  - $h(59, 1) = 5$ – collision!
  - h(59, 2) = 6 – collision!
  - h(59, 3) = 7 – collision!
  - h(59, 4) = 8

0	1	2	3	4	5	6	7	8	9	10
22	88			4	15	28	17	59	31	10

Exercise 11.4-1 Solution: Quadratic Probing

- $h(k, i) = (h'(k) + c_1 i + c_2 i^2) \bmod m = (k + i + 3i^2) \bmod 11.$
- Insert without collisions:
  - $h(10, 0) = 10$
  - $h(22, 0) = 0$
  - $h(31, 0) = 9$
  - $h(4, 0) = 4$

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

Exercise 11.4-1 Solution: Quadratic Probing

- $h(k, i) = (h'(k) + c_1 i + c_2 i^2) \bmod m = (k + i + 3i^2) \bmod 11$.
- Further inserts:
  - $h(15, 0) = 4$ – collision! $h(15, 1) = (15 + 1 + 3 \cdot 1^2) \bmod 11 = 8$
  - $h(28, 0) = 6$
  - $h(17, 0) = 6$ – collision! $h(17, 1) = (17 + 1 + 3 \cdot 1^2) \bmod 11 = 10$ – collision!
    $h(17, 2) = (17 + 2 + 3 \cdot 2^2) \bmod 11 = 9$ – collision!
    $h(17, 3) = (17 + 3 + 3 \cdot 3^2) \bmod 11 = 3$

0	1	2	3	4	5	6	7	8	9	10
22			17	4		28		15	31	10

Exercise 11.4-1 Solution: Quadratic Probing

- $h(k, i) = (h'(k) + c_1 i + c_2 i^2) \bmod m = (k + i + 3i^2) \bmod 11.$
  - $h(88, 0) = 0$ – collision!
    $h(88, 1) = (88 + 1 + 3 \cdot 1^2) \bmod 11 = 4$ – collision!
    $h(88, 2) = (88 + 2 + 3 \cdot 2^2) \bmod 11 = 3$ – collision!
    $h(88, 3) = (88 + 3 + 3 \cdot 3^2) \bmod 11 = 8$ – collision!
    $h(88, 4) = (88 + 4 + 3 \cdot 4^2) \bmod 11 = 8$ – collision!
    $h(88, 5) = (88 + 5 + 3 \cdot 5^2) \bmod 11 = 3$ – collision!
    $h(88, 6) = (88 + 6 + 3 \cdot 6^2) \bmod 11 = 4$ – collision!
    $h(88, 7) = (88 + 7 + 3 \cdot 7^2) \bmod 11 = 0$ – collision!
    $h(88, 8) = (88 + 8 + 3 \cdot 8^2) \bmod 11 = 2$

Problem: repeated probes
➥ not all slots will be tried
➥ cannot insert some keys while the table is not full

0	1	2	3	4	5	6	7	8	9	10
22		88	17	4		28		15	31	10

Exercise 11.4-1 Solution: Quadratic Probing

- $h(k, i) = (h'(k) + c_1 i + c_2 i^2) \bmod m = (k + i + 3i^2) \bmod 11.$
- Last insert:
  - $h(59, 0) = 4$ – collision!
    $h(59, 1) = (59 + 1 + 3 \cdot 1^2) \bmod 11 = 8$ – collision!
    $h(59, 2) = (59 + 2 + 3 \cdot 2^2) \bmod 11 = 7$

0	1	2	3	4	5	6	7	8	9	10
22		88	17	4		28	59	15	31	10

Exercise 11.4-1 Solution: Double hashing

- $h(k, i) = (h_1(k) + ih_2(k)) \bmod m = (\mathrm{k} + i[1 + (k \bmod 10)]) \bmod 11$
- Insert without collisions:
  - $h(10, 0) = 10$
  - $h(22, 0) = 0$
  - $h(31, 0) = 9$
  - $h(4, 0) = 4$

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

Exercise 11.4-1 Solution: Double Hashing

- $h(k, i) = (h_1(k) + ih_2(k)) \text{ mod } m = (\text{k} + i[1 + (k \text{ mod } 10)]) \text{ mod } 11$
- Further inserts:
  - $h(15, 0) = 4$ – collision!
    $h(15, 1) = (15 + 1[1 + (15 \text{ mod } 10)]) \text{ mod } 11 = 10$ – collision!
    $\text{h}(15, 2) = (15 + 2[1 + (15 \text{ mod } 10)]) \text{ mod } 11 = 5$
  - $h(28, 0) = 6$
  - $h(17, 0) = 6$ – collision!
    $h(17, 1) = (17 + 1[1 + (17 \text{ mod } 10)]) \text{ mod } 11 = 3$

0	1	2	3	4	5	6	7	8	9	10
22			17	4	15	28			31	10

Exercise 11.4-1 Solution: Double Hashing

- $h(k, i) = (h_1(k) + ih_2(k)) \bmod m = (k + i[1 + (k \bmod 10)]) \bmod 11$
- Last two inserts:
  - $h(88, 0) = 0$ – collision!
    $h(88, 1) = (88 + 1[1 + (88 \bmod 10)]) \bmod 11 = 9$ – collision!
    $h(88, 2) = (88 + 2[1 + (88 \bmod 10)]) \bmod 11 = 7$
  - $h(59, 0) = 4$ – collision!
    $h(59, 1) = (59 + 1[1 + (59 \bmod 10)]) \bmod 11 = 3$ – collision!
    $h(59, 2) = (59 + 2[1 + (59 \bmod 10)]) \bmod 11 = 2$

0	1	2	3	4	5	6	7	8	9	10
22		59	17	4	15	28	88		31	10

Exercise 11.4-1 Solution: General comments

- Watch the primary clustering (for linear probing), where keys that have almost the same hash lead to many collisions.
- For quadratic probing, a problem is that some slots are not touched by the probing function (it must repeat after at most *m* probes).
- There are very many collisions!
  The exercise has been created to demonstrate handling of collisions.
  Practical implementations avoid load factors $\alpha > 2/3$ (Python) or $> 3/4$ (Java).
  ➥ at most 7 or 8 entries in a hash table with 11 elements