**LAB 6**

SE172806 – Phạm Hoàng Minh Khôi

1. **Writing exercises:**

**1)**

Using separate chaining with unordered lists and the hash function h(x) = k mod M, where M = 5:

Inserting items with the keys E A S Y Q U E S T I O N in that order:

E (key = 5): [E]

A (key = 1): [A]

S (key = 19): [S]

Y (key = 25): [Y]

Q (key = 17): [Q]

U (key = 21): [U]

E (key = 5): [E, E]

S (key = 19): [S, S]

T (key = 20): [T]

I (key = 9): [I]

O (key = 15): [O]

N (key = 14): [N]

Contents of the hash table:

Index 0:

Index 1: A

Index 2:

Index 3:

Index 4: E, E

Index 5:

Index 6:

Index 7:

Index 8:

Index 9: I

Index 10:

Index 11:

Index 12:

Index 13:

Index 14: N

Index 15:

Index 16:

Index 17: Q

Index 18:

Index 19: S, S

Index 20: T

Index 21: U

Index 22:

Index 23:

Index 24:

Index 25: Y

**2)**

Using linear probing and the hash function h(x) = k mod M, where M = 16:

Inserting items with the keys E A S Y Q U E S T I O N in that order:

E (key = 5): [ , , , , , E, , , , , , , , , , ]

A (key = 1): [ , A, , , , E, , , , , , , , , , ]

S (key = 19): [ , A, , , , E, , , , , , , , , , S]

Y (key = 25): [ , A, , , , E, , , , , , , , , Y, S]

Q (key = 17): [ , A, , , , E, , , , , , , , Q, Y, S]

U (key = 21): [ , A, , , , E, , , , , , U, , Q, Y, S]

E (key = 5): [ , A, , , , E, , , , , , U, , Q, Y, S, E]

S (key = 19): [ , A, , , , E, , , , , , U, , Q, Y, S, E, S]

T (key = 20): [ , A, , , , E, , , , , , U, T, Q, Y, S, E, S]

I (key = 9): [ , A, , , , E, , , , I, , U, T, Q, Y, S, E, S]

O (key = 15): [ , A, , , , E, , , , I, , U, T, Q, Y, S, E, S, O]

N (key = 14): [ , A, , , , E, , , , I, , U, T, Q, Y, S, E, S, O, N]

Contents of the hash table:

Index 0:

Index 1: A

Index 2:

Index 3:

Index 4:

Index 5: E

Index 6:

Index 7:

Index 8:

Index 9: I

Index 10:

Index 11:

Index 12:

Index 13:

Index 14: N

Index 15: O

Index 16:

Index 17: Q

Index 18:

Index 19: S

Index 20: T

Index 21: U

Index 22:

Index 23:

Index 24:

Index 25: Y

**3)**

Using quadratic hashing and the hash function h(x) = k mod M for the initial probe, and resolving collisions by finding an available position at (h(x) + i^2)%M), where i = 1, 2, ...

Inserting items with the keys E A S Y Q U E S T I O N in that order:

E (key = 5): [ , , , , , E, , , , , , , , , , ]

A (key = 1): [ , A, , , , E, , , , , , , , , , ]

S (key = 19): [ , A, , , , E, , , , , , , , , , S]

Y (key = 25): [ , A, , , , E, , , , , , , , , Y, S]

Q (key = 17): [ , A, , , , E, , , Q, , , , , , Y, S]

U (key = 21): [ , A, , , , E, U, , Q, , , , , , Y, S]

E (key = 5): [ , A, , , , E, U, , Q, , , E, , , Y, S]

S (key = 19): [ , A, , , , E, U, , Q, , , E, , S, Y, S]

T (key = 20): [ , A, , , , E, U, , Q, T, , E, , S, Y, S]

I (key = 9): [ , A, I, , , E, U, , Q, T, , E, , S, Y, S]

O (key = 15): [ , A, I, , , E, U, , Q, T, , E, , S, Y, S, O]

N (key = 14): [ , A, I, , , E, U, , Q, T, , E, , N, Y, S, O]

Contents of the hash table:

Index 0:

Index 1: A

Index 2:

Index 3:

Index 4:

Index 5: E

Index 6:

Index 7:

Index 8:

Index 9: I

Index 10:

Index 11:

Index 12:

Index 13:

Index 14: N

Index 15: O

Index 16:

Index 17: Q

Index 18:

Index 19: S

Index 20: T

Index 21: U

Index 22:

Index 23:

Index 24:

Index 25: Y

**4)**

To calculate the value of the Shift Folding Hash Function with K = 432-351-459-763-88 and TSize = 1000:

Splitting K into segments: 432, 351, 459, 763, 88

Summing the segments: 432 + 351 + 459 + 763 + 88 = 2093

Calculating the remainder after dividing by TSize: 2093 % 1000 = 93

The value of the Shift Folding Hash Function with K = 432-351-459-763-88 and TSize = 1000 is 93.

**5)**

To calculate the value of the Boundary Folding Hash Function with K = 432-351-459-763-88 and TSize = 1000:

Splitting K into segments: 432, 351, 459, 763, 88

Reversing the segments: 234, 153, 954, 367, 88

Taking the last three digits from each segment: 234, 153, 954, 367, 088

Summing the segments: 234 + 153 + 954 + 367 + 088 = 1796

Calculating the remainder after dividing by TSize: 1796 % 1000 = 796

The value of the Boundary Folding Hash Function with K = 432-351-459-763-88 and TSize = 1000 is 796.