

# CPSC 2150 – Algorithms and Data Structure II

## Lab8: Hashing

Total - 40 Marks

---

### Learning Outcomes

- Practice on hashing and resolving collisions
- Implementing and programming with C++

### Resources

- Chapter 9 of the text book

### Description

**Exercise 1 – [20 marks]** You have a hash table of size  $m = 11$  and two hash functions  $h_1$  and  $h_2$ :

$h_1(x) = (\text{sum of the values of the first and last letters of } x) \bmod m$

$h_2(x) = ((\text{value of the last letter}) \bmod (m - 1)) + 1$

where the value of a letter is its position in the alphabet (e.g.,  $\text{value}(a)=1$ ,  $\text{value}(b)=2$ , etc.). Here are some precomputed hash values:

**word:** *ape bat bird carp dog hare ibex mud koala stork*

**h1:** 6 0 6 8 0 2 0 6 1 8

**h2:** 6 1 5 7 8 6 5 5 2 2

- A. **[7 marks]** Draw a picture of the resulting hash table after inserting, in order, the following words:

*ibex, hare, ape, bat, koala, mud, dog, carp, stork.*

- B. **[3 marks]** Highlight cells that are looked at when trying to find bird.

Do part A and B for each of the following techniques:

1. Separate chaining with  $h_1$  as your hash function.
2. Double hashing with  $h_1$  as your first hash function and  $h_2$  as your second hash function.

**Exercise 2 – Hash worse case:** A hash table of size  $M$  stores  $N$  integer keys. Collisions are handled by chaining and the hash function is  $h(K) = K \bmod M$ .

1. **[8 marks]** What is the worst-case search time? Give an example of a set of keys that achieves the worst-case search time.
2. **[2 marks]** Would you use this hash table for a time-critical application (e.g., air traffic control)?

**Exercise 3 – Linear probing with load factor:** Demonstrate the insertion of the keys 5,28,19,15,20,33 into a hash table with collisions resolved by linear probing. Assume that the hash table has  $m$  slots ( $m=7$ ) and its load factor is 0.70 and the hash function is  $h(k) = k \bmod m$ . For rehashing, choose the closest prime number less than twice of current  $m$  as the new value of  $m$ .

**[10 marks]** Draw the state of the hash table after every insertion.

### **SUBMIT to D2L**

Submit a zip file named **StudentNumber-Lab8.zip** including **answers.pdf** file by the due date. For example, if your student number is 10023449, the submitted file must be named as **10023449-Lab8.zip**.