CSCI 321 Computer Science III Summer 2019

Lecture 5 Activity 1

1. Draw the 11-entry hash table that results from using the hash function, h(i) = (3i+5) mod 11, to hash the keys 12, 44, 13, 88, 23, 94, 11, 39, 20, 16, and 5, assuming collisions are handled by chaining.

Index Keys

0 🡪 13

1 🡪 94 🡪 39

2

3

4

5 🡪 44 🡪 88 🡪 11

6

7

8 🡪 12 🡪 23

9 🡪 16 🡪 5

10 🡪 20

1. Redo P1 assuming collisions are handled by linear probing.

h(12)=(3x12+5) mod 11 = 41 mod 11 = 8

h(44)=(3x44+5) mod 11 = 137 mod 11 = 5

h(13)=(3x13+5) mod 11 = 44 mod 11 = 0

h(88)=(3x88+5) mod 11 = 269 mod 11 = 5 -> collision

(h(88)+f(1)) mod 11 = (5+1) mod 11 = 6 mod 11 = 6

h(23)=(3x23+5) mod 11 = 74 mod 11 = 8 -> collision

(h(23)+f(1)) mod 11 = (8+1) mod 11 = 9 mod 11 = 9

h(94)=(3x94+5) mod 11 = 287 mod 11 = 1

h(11)=(3x11+5) mod 11 = 38 mod 11 = 5 -> collision

(h(11)+f(1)) mod 11 = (5+1) mod 11 = 6 mod 11 = 6 -> collision

(h(11)+f(1)) mod 11 = (5+2) mod 11 = 7 mod 11 = 7

h(39)=(3x39+5) mod 11 = 122 mod 11 =1 -> collision

(h(39)+f(1)) mod 11 = (1+1) mod 11 = 2 mod 11 = 2

h(20)=(3x20+5) mod 11 = 65 mod 11 = 10

h(16)=(3x16+5) mod 11 = 53 mod 11 = 9 -> collision

(h(16)+f(1)) mod 11 = (9+1) mod 11 = 10 mod 11 = 10 -> collision

(h(16)+f(2)) mod 11 = (9+2) mod 11 = 11 mod 11 = 0 -> collision

(h(16)+f(3)) mod 11 = (9+3) mod 11 = 12 mod 11 = 1 -> collision

(h(16)+f(4)) mod 11 = (9+4) mod 11 = 13 mod 11 = 2 -> collision

(h(16)+f(5)) mod 11 = (9+5) mod 11 = 14 mod 11 = 3 -> collision

h(5)=(3x5+5) mod 11 = 20 mod 11 = 9 -> collision

(h(5)+f(1)) mod 11 = (9+1) mod 11 = 10 mod 11 = 10 -> collision

(h(5)+f(2)) mod 11 = (9+2) mod 11 = 11 mod 11 = 0 -> collision

(h(5)+f(3)) mod 11 = (9+3) mod 11 = 12 mod 11 = 1 -> collision

(h(5)+f(4)) mod 11 = (9+4) mod 11 = 13 mod 11 = 2 -> collision

(h(5)+f(5)) mod 11 = (9+5) mod 11 = 14 mod 11 = 3 -> collision

(h(5)+f(6)) mod 11 = (9+6) mod 11 = 15 mod 11 = 4 -> collision

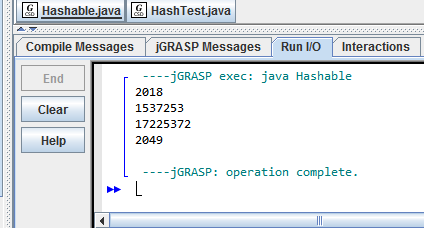
1. Redo P1 assuming collisions are handled by double hashing using the secondary hash function d(k) = 7− (k mod 7).
2. Hash code is used to “encode” general keys into integers. One approach of creating a hash code is to use Java's hashCode() method. The hashCode() method is implemented in the Object class and therefore each class in Java inherits it. The hash code provides a numeric representation of an object (this is somewhat similar to the toString method that gives a text representation of an object).
3. Write a program to show the hash code of the following.

Integer 2018,

String “2018”,

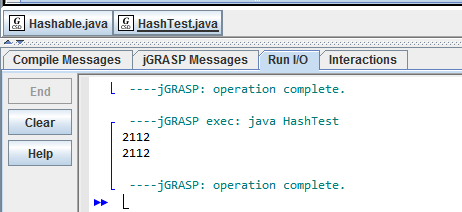
StringBuffer “2018”,

ArrayList with first element is Integer 2018.



import java.util.ArrayList;  
  
public class Hashable {  
 public static void main(String[] args)   
 {  
 Integer i = new Integer(2018);  
 String str="2018";  
 StringBuffer sb = new StringBuffer("2018");  
 ArrayList al = new ArrayList();  
 al.add(i);  
 System.out.println(i.hashCode());  
 System.out.println(str.hashCode());  
 System.out.println(sb.hashCode());  
 System.out.println(al.hashCode());  
   
 }  
}

1. Verify that different objects might have the same hashcode. For example, show the hashcode for the strings “Aa” and “BB”.



public class HashTest {  
 public static void main(String[] args)   
 {  
   
 String str="Aa";  
 String str2="BB";  
 System.out.println(str.hashCode());  
 System.out.println(str2.hashCode());  
   
 }  
}