# Data Structures and Algorithms

# INFO 6205

# Homework 3

# Due: May 29, 2020

Put all your java, compiled class files and documentation files into a zip file named Homework3.zip and submit it via dropbox to blackboard before the END of due date. Put your name on all .java files. There will be a short quiz on this homework.

1. Consider PriorityQueue data structure discussed in class. Create an input file to include all

mixed integer numbers. You are to write Java code for MaxPQ and MinPQ. Your code should support all operations of priority queue. Compile and run your program.

2. Consider String “There are sunny days, rainy days, and windy days”

a) Generate a binary Huffman Tree

b) Show binary data both before and after compression. Analyze difference.

c) Consider this Java code example:

https://www.geeksforgeeks.org/huffman-coding-greedy-algo-3/

Write Java code for Huffman algorithm

Show as to why/how the algorithm uses PriorityQueue

d) Compile Java code and run it with the input string provided above.

3. Consider the following, Input Data: {21, 18, 38, 3, 9, 82, 10, 31, 25, 29 }

a) Graphically build a Circular queue for input data. Discuss and

show Head and Tail pointers at each step:

i) enqueue all input data

ii) dequeue three elements

iii) enqueue two elements

iv) dequeue all elements

b) Write Java code for the Circular queue, provide enqueue, dequeue,

isEmpty, isFull, and displayQueue methods, to show the status of the queue

with steps described in (a). Compile code and Run with input data.

4. Consider signed byte X, and unsigned byte Y. What are the possible values for both

X and Y can have?

5. Human use Infix expression and computers use Postfix expression. You are to write a simple Calculator. There are three steps: a) Read Infix expression, b) Convert Infix expression to Postfix by hand, and c) Evaluate Postfix expression. Consider this Algorithm:

Maintain a stack and scan the postfix expression from left to right – When we get a number, output it – When we get an operator, pop the top element in the stack until there is no operator having higher priority then this operator, and then push (current operator) into the stack – When the expression is ended, pop all the operators remain in the stack.

A) Write Java code to transform this Infix expression to Postfix:

(4 + 8) \* (6 - 5)/((3 - 2) \* (2 + 2))

B) Write Java code to Compute postfix expression.

6. Consider the following code for User class.

A) Discuss code in details

B) Write Java code to test User class with multiple test cases to test quals,

hashCode and CompareTo methods.

public class User implements Comparable<User> {

private String name;

private int id;

private Date birth;

public User (String name, int id, Date birth)

{ this.name = name; this.id = id; this.birth = birth; }

@Override

public boolean equals(Object other) {

if (this == other) return true;

if (other == null || (this.getClass() != other.getClass()))

{ return false; }

User guest = (User) other;

return(this.id == guest.id) &&

(this.name = null && name.equals(guest.name)) &&

(this.dob != null && dob.equals(guest.birth));

}

@Override

public int hashCode() {

int result = 0;

result = 31\*result + id;

result = 31\*result + (name !=null ? name.hashCode() : 0);

result = 31\*result + (birth !=null ? dob.hashCode() : 0);

return result;

}

@Override

public int compareTo(User o) {

return this.id - o.id; } }}

7. Java Generics allow you to build collections with unique data type. To perform uniqueness, comparisons of object types need to be made:

A) Using compareTo() method from Comparable interface, the equals, and hashcode.

Explain the differences?

B) Java String class object hashCode is described as following: pasted-image.tiff

What is the hashCode 32-bit integer number for string =“Welcome Students to Class“ :

a) mathematically by hand, b) Write Java code