Tshwane University of Technology

We empower people

Student Number	
Surname & Initials	

YEAR: 2020
SEMESTER: 1
ASSESSMENT: B

SUBJECT NAME:	DATA STRUCTURES AND ALGORITHMS V
SUBJECT CODE:	DTD117V
QUALIFICATION(S):	ADRS20 ADVANCED DIPLOMA IN COMPUTER SCIENCE

PAPER DESCRIPTION: COMPUTER BASED **DURATION:** 3 HOURS **PAPER:** ONLY

SPECIAL REQUIREMENTS

- ☐ **NONE**
- ☒ **NON-PROGRAMMABLE POCKET CALCULATOR**
- ☐ **SCIENTIFIC CALCULATOR**
- ☐ **COMPUTER ANSWER SHEET**
- ☐ **GRAPH PAPER**
- ☐ **DRAWING INSTRUMENTS**

OTHER: COMPUTER
ANSWER BOOK

INSTRUCTIONS TO CANDIDATES: ANSWER ALL QUESTIONS

THIS TEST IS TO BE ANSWERED ON A PC AND ON AN ANSWER BOOK. WHEN YOU ARE DONE, GIVE YOUR ANSWER SCRIPT BACK TO THE INVIGILATOR AND SUBMIT YOUR FILE.

TOTAL NUMBER OF PAGES INCLUDING COVER PAGE: 7

TOTAL NUMBER OF ANNEXURES: 0

EXAMINER: Dr Ntsako Baloyi **FULL MARKS:** 100

MODERATOR: Ms Mpho Nkosi **TOTAL MARKS**: 100

STUDENT TOTAL: ___

STUDENT %: ___

Preparation

Theory (Section A)
a. Use the answer book to answer questions.
b. Show all steps where applicable.
c. Write legibly

Practical (Section B)
a. Use Java/C++ to create the programs required in section 2
b. Write your code legibly and provide comments
c. To evidence your output, capture the output (screen prints) of all the various functionality requirements into a word document (correctly numbered) and save it as a PDF file at the end.
d. You are then required to put the PDF output file and the associated code into a folder (named suing student number) and send it to the Lecture PC.

Section A – Theory (Please use your answer booklet to write this section) **[50]**

Question 1 – True/False **[10]**

Indicate whether each of the following questions is **True** or **False**.

1.1 A double rotation on an AVL tree is called a zig zig rotation. True/**False** (2)

1.2 A heap is a complete binary tree, except for last level. **True**/False (2)

1.3 An abstract data type specifies an implementation not the interface of a data structure. True/**False** (2)

1.4 C++ has a Standard Template Library that implements basic data structures. **True**/False (2)

1.5 Elements in a queue are removed from the front of the queue. **True**/False (2)

Question 2 – Multiple choice **[10]**

Select the correct answer for each question from the given options.

2.1 A stack is a _____ structure. (2)

a. LIFO
b. FIFO
c. LILO
d. FILO

2.2 A _____ is a type of a heap? (2)

a. Binary tree heap
b. Min heap
c. Zig zag
d. Heap Up

2.3 The number of edges on the longest path from root node to a leaf node of a tree data structure. (2)

a. Length
b. Height
c. Depth
d. Level

2.4 Deletion of elements in a heap tree only takes place from _____. (2)

a. The root node
b. The last element entered
c. The rear of the tree
d. The leaf

2.5 The number of nodes in a tree data structure are computed using the _____ function. (2)

a. root()
b. depth()
c. height()
d. size()

Question 3 **[15]**

3.1 Explain how a deletion is performed in an AVL tree. (5)

Deleting any node requires one to recalculate the balance factors to determine if the tree is still balanced after the deletion. If not balanced, one needs to balance it out. When deleting a node with children, replace that node with its inorder predecessor (largest element from the left subtree) or inorder successor (smallest element from the right subtree).

3.2 Distinguish between queue and deque data structures. (4)

The ArrayQueue from the previous section is a data structure for representing a sequence that allows us to efficiently add to one end of the sequence and remove from the other end. The ArrayDeque data structure allows for efficient addition and removal at both ends.

3.3 What condition needs to hold before the dequeue() function can successfully be executed? (2)

The queue must not be empty.

3.4 Which condition must be satisfied for a resize() function to increase the array size? (2)

a.length > 3n

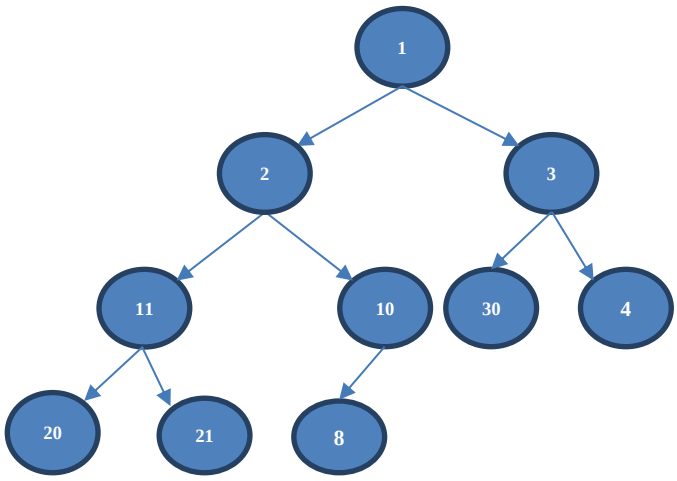
3.5 What is the formula to calculate the balance factor for AVL trees? (2)

Balance factor = Height (left subtree) – Height (right subtree)

Question 4 – Problem solving **[15]**

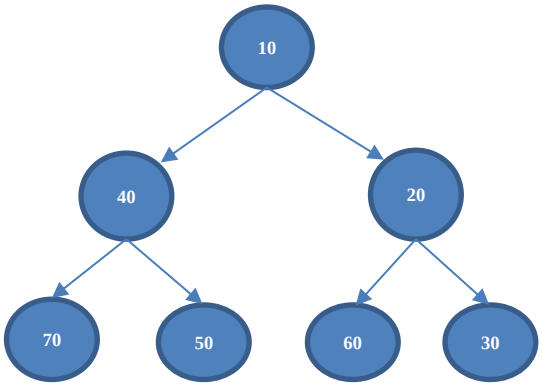
4.1 Construct a min heap using the following elements or insert the following elements into a min heap: **10, 20, 30, 1, 2, 3, 4, 11, 21, 41.** (10)

Answer

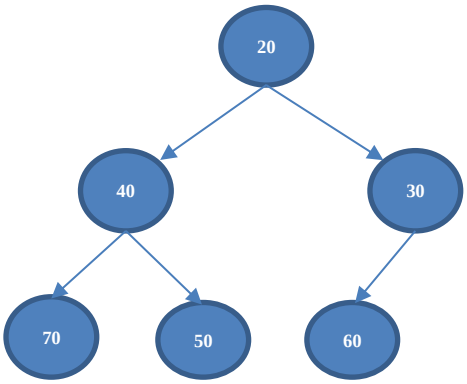


[allocate 2 marks per subtree]

4.2 Perform a delete operation on the following heap and depict the resultant new balanced heap. (5)



Answer



[allocate 1 mark per correctly positioned element, excluding the root]

Section B – Practical/Programming (Please use PC to answer this section) [50]

Question 5 [30]

Develop a simple java or C++ program that uses integer stack and list data structure. The stack should be called NumberStack and list NumberList. Perform the following operations:

5.1 Create and initialise NumbersStack with the following values: 55, 46, 90, 39, 20, 13, 56, 100, 20 and 77 using the appropriate function. Each initialisation operation should display a message such as "value XX added into stack Numbers". (9)

Answer: NumbersStack stack should have the values 55, 46, 90, 39, 20, 13, 56, 100, 20 and 77

[allocate 6 marks for correct initialisation using the stack data structure and 3 marks for displaying a message as each initialisation operation is taking place]

5.2 Display the values in the NumberStack data structure. (3)

Answer: NumbersStack stack should have the values 55, 46, 90, 39, 20, 13, 56, 100, 20 and 77

[allocate 3 marks for displaying the values in the correct order from the stack]

5.3 Display the size of the NumberStack data structure. (2)

Answer: 10

[allocate 2 marks for correct answer]

5.4 Create the NumberList list data structure and initialise it by moving the top half numbers of the NumberStack data structure into the NumberList data structure. (12)

Answer: NumbersList list should have the values 77, 20, 100, 56 and 13

[allocate 1 mark for each pop() operation, 1 mark for each insert() operation totaling 10 marks and 2 marks for having the list elements in the correct order]

5.5 Display the values in the NumberList data structure. (2)

Answer: NumbersList list should have the values 77, 20, 100, 56 and 13

[allocate 2 marks for displaying the values from the list in the correct order]

5.6 Display the size of the NumberList data structure. (2)

Answer: 5

[allocate 2 marks for correct answer]

Question 6	**[20]**

Develop a simple java or C++ program to demonstrate the use of the binary search tree data structure. Perform the following operations on a String linked list:

6.1 Create and initialise a balanced binary search tree called **NumbersTree** with the following values: 5, 9, 7, 3, 8, 12, 6, 4 and 20. (10)

Answer: check the code and verify with the output displayed in the following questions.

[allocate 10 marks for correctly initialization the BST data structure and balancing the tree]

6.2 Write a function to and display the values of **NumbersTree** in Preorder transversal. (10)

Answer: 5, 3, 4, 9, 7, 6, 8, 12 and 20

[allocate 10 marks for correct displaying the values in the correct order]

The End