

## Complexity

2022 – 15 Marks

1.1 Use  $\tau$  - notation to determine the running time of the following program lines:

1a, 1b, 1c, 2 (8)

```
1. for ( int i=1; i<=n; i++) {  
2.     y = arr[i] * x;  
3. }
```

1.2 Use the asymptotic model to determine the running time of lines 5, 6b, 8a, 8b, 9 and 11 in the context of this program segment. In addition, you need to determine the Total. (7)

```
1 public class Question 1.2  
2 {  
3     public static int numbers (int n)  
4     {  
5         int prod = 1;  
6         for (int i=0; i<=n+2; i++)  
7         {  
8             for ( int j=0; j<i; ++j)  
9                 prod = prod + i;  
10        }  
11        return prod;  
12    }  
13 }
```

2021 – 15 marks

1.1 Use  $\tau$  - notation to determine the running time of the following program lines.

(8)

```
1. for ( int i=0; i<=n; i++) {  
2.     y = arr[i]+getVal(i);  
3. }
```

1a

1b

1c

2

1.2 Use the simplified model to determine the running time of **ONLY lines 6b, 8a and 8b** in context of this program segment. You need not simplify the expressions.

(7)

```
1 public class Question 1.2
2 {
3     public static int numbers (int n)
4     {
```

```
5         int prod = 1;
6         for (int i=1; i<=n+2; i++)
7         {
8             for ( int j=0; j<=i; ++j)
9                 prod = prod + i;
10        }
11        return prod;
12    }
13 }
```

2019 – 20 marks

1.1 Use  $\tau$  - notation to determine the running time of the following program lines. (8)

```
1. for ( int i=0; i<=n; i++) {
2.     y = arr[i];
3.     z= fix(x);
4. }
```

1 a

1.2 Use the simplified model to determine the running time of **ONLY lines 6b, 8b and 9** in context of this program segment. You need not simplify the expressions. (8)

```
1 public class Question 1.2
2 {
3     public static int numbers (int n)
4     {
5         int prod = 1;
6         for (int i=0; i<=n; i++)
7         {
8             for ( int j=1; j<=i+1; ++j)
9                 prod = prod + i + 2;
10        }
11        return prod;
12    }
```

1.3. Give the definition for  $O(n)$ .

1.4 Two functions representing the running time of a program, derived from the simplified mode, are given as follows:

$F_1(n) = n^2 + 9n + 1$  and

$F_2(n) = 7n + 5$ .

Use the big Oh properties to compute the asymptotic running time of the following:

1.4.1  $F_1 + F_2$

1.4.2  $F_1 * F_2(2)$

## MyArrayList

2022 – 15 marks

2.1 The given implementation class MyArraylist does NOT support generic objects. Write down the correct version of the program lines of this class in your script to address this problem. (3)
2.2 Write the Java code of the CompareTo() method for the clubMember class to enable it to be sorted in a GENERIC MyArrayList class. Assume that members are compared according to their handicap. (5)
2.3 Write the Java code for the remove() method in the MyArrayList class. (7)

2021 – 15 marks

2.1 Discuss the difference between an array and linked list implementation of a list in terms of:

- i. Data element storage and access
- ii. Deletion of data items

*Employ diagrams to support your answer. (10)*

2.3 Write the Java code for the toString method in the MyArrayList classs. (5)

2019 – 20 marks

2.1 Discuss the difference between an array and linked list implementation of a list in terms of:

- i. Data element storage and access
- ii. Deletion of data items

*Employ diagrams to support your answer. (10)*

2.2.1 The given implementation class MyArraylist does NOT support generic objects. Write down the correct version of program lines of this class in your script to address this problem. (3)

2.2.2 Write the Java code of the appropriate method for the GymMember class to enable it to be sorted a GENERIC MyArrayList class. Assume that members are compared according to their addresses. (3)

2.2.3 Write the Java code for the toString method in the MyArrayList class. (4)

# MyLinkedList

2022 – 30 marks

3.1 Develop a **delete()** method for the class `myLinkedList`. The method removes the first element that matches the parameter object by value. The method returns a Boolean value indicating whether the task was successful. (8)

3.2 Design and code a method for the `MyLinkedList` class called *getMissing( ... )*. This method receives a linked list as parameter and should return a linked list with all elements that are in the calling list but not in the parameter list.

Example: All values indicate Integer objects.

Calling List = {3, 1, 5, 8, 6, 5}

Parameter List = {8, 6, 5, 4}

Returned List = {3, 1}

3.2.1 Draw a linked list to help you to design the algorithm.

3.2.2 Write down the overall plan for the general case of the problem in English.

3.2.3 Write down all the special cases for the problem in English. Give a short description of the required action for each of these cases. (6)

3.2.4 Write the method *getMissing ()* in Java. (11)

3.2.5 Write a test program to test the method thoroughly. (5)

2021 – 30 marks

3.1 Write a method in Java for the class `MyLinkedList` called *delete* that removes an element that matches the parameter object by value. (8)

3.2 Design and code a method for the `MyLinkedList` class called: *sharedElements( ... )*. This method receive a linked list as parameter and should return a linked list with all elements that are in both the calling and the parameter list. The returned list however may not have any duplicates.

Example: All values indicate Integer objects.

Calling List = {3, 1, 5, 8, 6, 5}

Parameter List = {8, 6, 5}

Returned List = {5, 6, 8}

3.2.1 Draw a linked list to help you to design the algorithm.

Write down the general and all the special cases for the problem in English. Give a short description of the required action for each of these cases. (5)

3.2.2 Write the method *sharedElements (...)* in Java. (12)

3.2.3 Write a test program to test the method thoroughly.  
(5)

2019

3.1 Write a method in Java for the class `MyLinkedList` called *append* that adds a new element at the back of the list (5)

## Stacks and Queues

2022 – 10 marks

4.1 When considering the operations of a stack implemented with a linked list. On what side must what operation be performed? Motivate your answer. (4)

4.2 What is the output of the following code? (2)

```
StackAsMyLinkedList stack = new StackAsMyLink
stack.push(new Integer(6));
stack.push(new Integer(4));
stack.push(new Integer(3));
stack.pop();
stack.push(new Integer(5));
stack.push(new Integer(1));
stack.pop();
stack.push(new Integer(9));
System.out.println(stack);
```

4.3 Write the code for the dequeue() method implemented with the MyArrayList class. (4)

2021 – 10 marks

4.1 Both operations should be done on the same side of the array when working with a Stack. For both operations state which is easier/ better to program and why? (4)

4.2 What is the output of the following code?

```
StackAsMyLinkedList stack = new
StackAsMyLinkedList();
stack.push(new Integer(6));
stack.push(new Integer(4));
stack.push(new Integer(3));
stack.push(new Integer(5));
stack.push(new Integer(1));
System.out.println(stack);
```

(2)

4.3 Write the Java code for the *dequeue* method implemented with the MyArrayList class (4)

4.1 Explain the difference between a stack and a queue. Use a diagram to support your answer. (4)

4.2 Complete the code for the *push()* method by writing the missing lines in the space provided.

(3)

```
1. public class StackAsMyLinkedList<E>
2. {
3.     MyLinkedList<E> theStack;
4.     public StackAsMyLinkedList()
5.     {
6.         theStack = new MyLinkedList<E>();
7.     }

8.     public void push (_____)
9.     {
10.         _____;
11.     }
```

8

10

## Recursion

2022 – 9 marks

5.1 What is recursion? (2)

5.2 Compare iterative algorithms with recursive algorithms. (3)

5.3 Complete the method TimesFive so that it will calculate  $n \times 5$  for  $n \geq 0$  recursively (4)

```
public static int TimesFive(int n)
{
```

2021 – 9 marks

5.1 What is recursion? (2)

5.2 Compare iterative algorithms with recursive algorithms. (3)

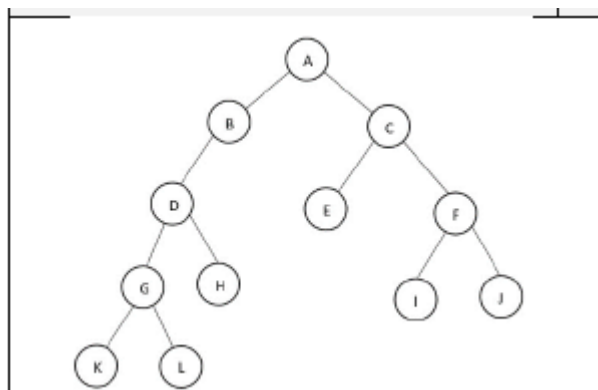
2019

5.1 Explain the two main parts of any recursive algorithm. (4)

5.2 Write down a recursive method in java to compute  $x^n$ . (5)

## Trees

2022 – 8 marks



7.1 Level of node E is:

7.2 The outdegree of C:

7.3 Height of the tree is:

7.4 Complete tree:

7.5 Leaf:

7.6 Balanced tree:

7.7 Order of post-order Traversal:

### Vraag 8 / Question 8

2021 – 8 marks

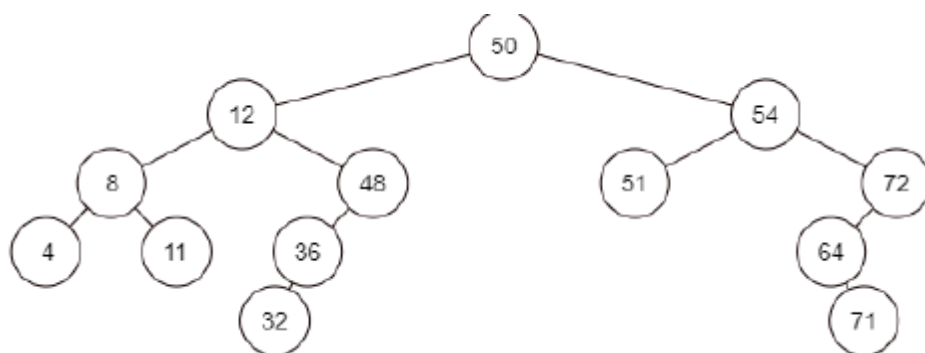
7.1 How does a tree differ from data structures like lists, stacks and queues? (2)

7.2 Write down the order in which the following algorithms will traverse this binary tree:

7.2.1 In-order (2)

7.2.2 pre-order (2)

7.2.3 post-order (2)



2019 – 8 marks

7.1 Define the following terms used in connection to binary trees.

7.1.1. Height of the tree (2)

7.1.2 Binary tree (2)



7.2 Write down the order in which the following algorithms will traverse this binary tree:

7.2.1 In-order

(2)

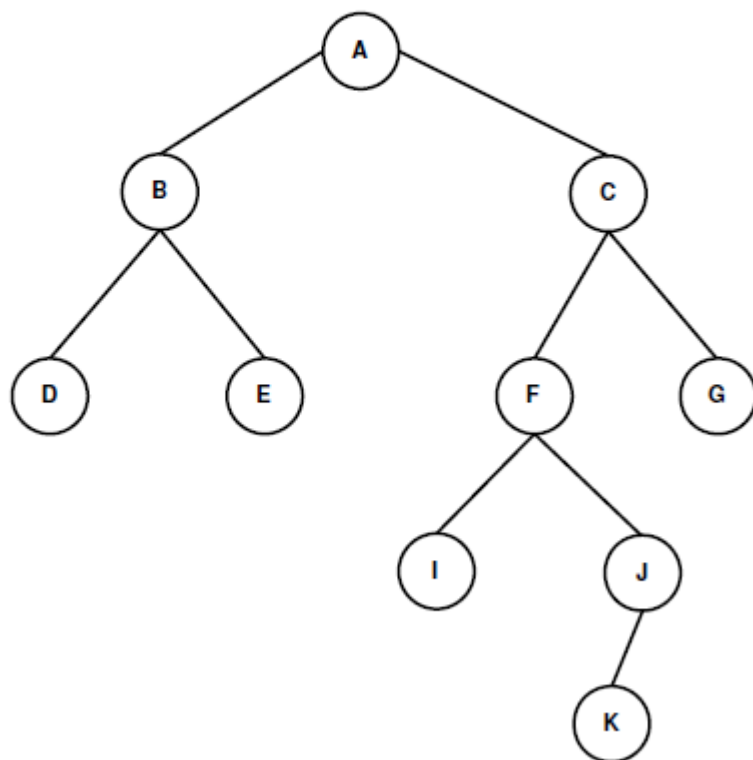
7.2.2 Pre-order

(2)

7.2 Skryf die volgende neer waar die algoritmes hierdie binêre boom sal deur

7.2.1 In-orde

7.2.2 Pre-orde



## Sorting

2022 – 8 marks

Indicate by means of a diagram how the insertion sort algorithm will sort the following list of numbers: (8)  
8; 1; 6; 3; 7; 4; 2; 5;

2021 – 8 marks

Indicate by means of a diagram how the merge sort algorithm will sort the following list of numbers: (8)  
8; 1; 6; 3; 7; 4; 2; 5;

2019

6. Describe the algorithm and indicate by means of a diagram how the bucket sort algorithm will sort the following list of numbers: (8)

5; 2; 2; 4; 1; 3; 2; 1;

## Hashing functions

2022 – 5 marks

8.1 Explain in your own words how a hash table stores data. (3)

8.2 What is a perfect hashing function? (2)

2021 – 5 marks

8.1 Explain in your own words how a hash table stores data. (3)

8.2 What is a perfect hashing function? (2)

2019 – 5 marks

8.1 Explain in your own words how a hash table stores data. (3)

8.2 What is a perfect hashing function? (2)