The University of Nottingham Ningbo China

Centre for English Language Education

Semester One 2023-2024

INTRODUCTION TO ALGORITHMS

Time allowed 2 hours

Candidates may complete the information required on the front page of this booklet but must NOT write anything else until the start of the examination period is announced.

This paper comprises FIVE questions. Answer all questions.

All questions carry equal marks. Marks total 50.

No calculators are permitted in this exam.

Dictionaries are not allowed with one exception.Those whose first language is not English may use a standard translation dictionary to translate between that language and English provided that neither language is the subject of this examination. Subject specific translation dictionaries are not permitted.

No electronic devices capable of storing and retrieving text, including electronic dictionaries, may be used.

Do not turn examination paper over until instructed to do so.

ADDITIONAL MATERIAL: None.

INFORMATION FOR INVIGILATORS:

1. *Please advise students at the start of the exam that they can do rough work on the last page.*
2. *Please give a 15 minutes warning.*
3. *Please collect this booklet at the end of the exam.*

This page is intentionally left blank

Question 1:

1 (a) Given numbers $X = 8$, $Y = 3$, and $Z = 5$. [3]

Determine the values of the following expressions:

(i) X mod Y.
(ii) $X * Y / Z$.
(iii) $(X * Z)$ mod Y.

(b) Given $a =$ true and $b =$ false. State the result of: [1]

```
if OR (AND(a,b),b) then
    return a
else
    return b
end if
```

(c) A fashion store awards a discount to all of its customers over the Spring Holiday period, using the following policy.

If a customer's purchases reach 500 RMB or more they will get a discount of 10%, and for a purchase of 1000 RMB or more the customer will get a discount of 25% on the total bill. For smaller purchases a standard discount of 2% is given.

Write an algorithm which calculates the discount a customer will receive based on the total bill, and returns the amount to be paid after discount is subtracted. [3]

(d) Write an algorithm isEven that takes a number x as an argument and determines whether or not x is even. You may use addition, subtraction, equality tests, and recursion.
(*You may NOT use any division or mod operators.*) [3]

Question 2:

(a) Write a recursive algorithm **sumDownBy2** with the following specifications:

Requires: a positive integer n

Returns: a positive integer which is the sum $n + (n-2) + (n-4) + \ldots + 0$.

e.g. **sumDownBy2**(7) should return $7 + 5 + 3 + 1 = 16$.

sumDownBy2(8) should return $8 + 6 + 4 + 2 = 20$.

sumDownBy2(0) should return 0.

sumDownBy2(-1) should return 0. [3]

(b) The least common multiple (LCM) of two numbers is the smallest multiple of both the numbers. Write a recursive algorithm **LCM** with the following specifications:

Requires: positive integers x and y

Returns: the least common multiple (LCM) of x and y

e.g. **LCM**$(3, 5)$ should return 15.

LCM$(6, 8)$ should return 24. [3]

(c) Write a recursive algorithm, **binL**, which takes an n −digit number N as argument and returns another number with n −digits but with 1 in the place of odd digits and 0 in the place of even digits.

e.g. **binL**(352176) should return 110110.

binL(412986) should return 010100.

Use $N = 3416$ to test the algorithm showing all of the intermediate steps. [4]

Question 3:

(a) Given list L_1: [5, 9, 8].

(i) Write list L_1 explicitly using Nil and Cons.

(ii) Write pseudocode to obtain second element of list L_1.

(iii) What is the result of isEmpty (tail (tail (tail (L_1)))))? [2]

(b) Write an algorithm **countN** which takes a list of integers, *list*, and an integer N and returns the number of elements of list that are less than N. [3]

e.g. **countN** ([2, 3, 3, 7, 6, 8], 7) should return 4

(because there are 4 numbers less than 7)

countN ([2, 3, 3, 7, 6, 8], 5) should return 3

countN ([2, 3, 3, 7, 6, 8], 2) should return 0

(c) Write a recursive algorithm **split_IV**, that splits a list of numbers into a pair of two lists. The first list consists of all the even numbers and the second list consists of all odd numbers from the given list. [3]

e.g. **split_IV** ([9,4,3,5,2,7]) should return

$L_1 = [4, 2]$ and $L_2 = [9, 3, 5, 7]$.

(d) Write a recursive algorithm to determine whether or not the sum of any two numbers in the given list equals given value x. [2]

e.g. list = [3,5,4,7] and $x = 7$ should return true

(because 3+4 =7 is possible)

list = [3,5,4,7] and $x = 5$ or 13, should return false

(because no two numbers in the list add to either 5 or 13).

Use the list [3,7,6] and $x = 9$ to test the algorithm showing all the intermediate steps.

Question 4:

(a) Write an algorithm **ZeroIN** that takes a binary tree as input and returns true if the given tree contains the value 0; otherwise false. Your algorithm should work even if the input tree is empty. [3]

(b) Write an algorithm to check whether the given binary search tree is balanced or not.

Test your algorithm for the binary search tree t (given below). Also show all the intermediate steps. [3]

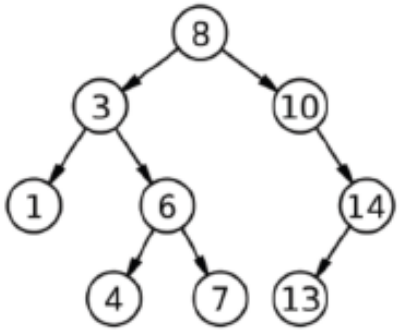
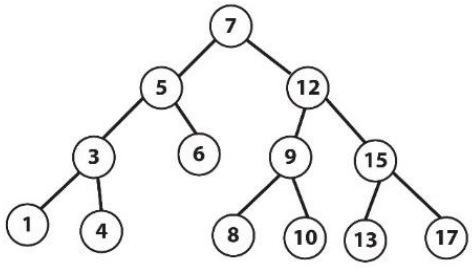


Figure -1: Binary search tree *t*

(c) The following operations are to be carried out in sequence on the binary search tree given below. [4]

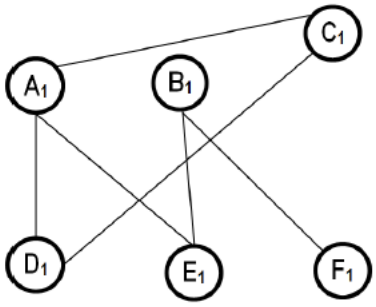
I. Add 2,
II. Add 11,
III. Delete 17,
IV. Delete 5.



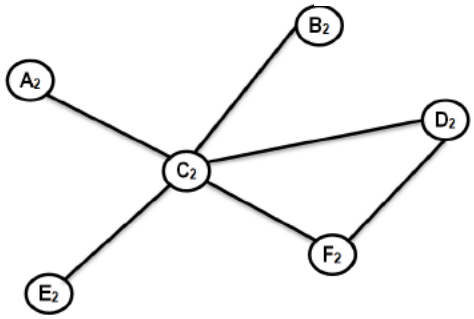
Draw a binary search tree structure after each operation in the sequence.

Note you should number each binary search tree I to IV and note that each new operation will be carried out on the previous tree.

5 (a) For each of the given graphs, determine whether the graph has a Euler path, a Euler tour, or neither. Justify your answer.



graph.1



graph.2

Note: -- Euler tour /Euler circuit is same.

5 (b) Given directed graph:

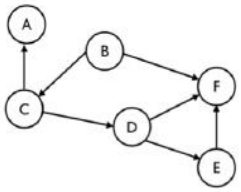


Figure-4

(i) Find any two possible topological sorting orders.

(ii) Which edge must be removed for the ordering ABCDEF to be a topological sorting order?

5 (c) Find a minimum spanning tree for the given weighted graph.