The University of Nottingham Ningbo China

Centre for English Language Education

Semester One 2024-2025

MID-SEMESTER EXAMINATION

INTRODUCTION TO ALGORITHMS

Time allowed 60 Minutes

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 has Twenty Questions :

Fifteen multiple-choice questions.
Five questions Short answer type.

All answers must be written in this booklet.

No calculators are permitted in the 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.

INFORMATION FOR INVIGILATORS:

1. *Please give a 15-minute warning before the end of the exam*
2. *Please collect this booklet at the end of the exam.*

Student ID: ________________

Seminar Group (e.g. A35 or B13 or C17): ____________ **Marks (out of 50):** _______

This page is intentionally blank

You may use this space for rough work

Section A

Tick (✓) exactly ONE most appropriate answer for each of questions 1-20.

1. What can be said about the value of the following compounded statement:

```
(!A && !B) || (A && B)
```

☐ it is True when both `A` and `B` are False or both are True

☐ it is True only when one of `A` and `B` is True

☐ this statement is always True

☐ this statement is always False

2. Suppose you have a computer that can only perform subtraction; i.e. `a-b`. How would you then implement `x%y,(x>y)`?

☐ keep subtracting `y` from `x` until you get to `0`

☐ keep subtracting `x` from `y` until you get to `0` or `1`

☐ keep subtracting `y` from `x` until you get to a value less than `x`

☐ keep subtracting `y` from `x` until you get to a value less than `y`

3. Consider the following recursive algorithm:

Algorithm: `f(x)`

```
if (x>100)
   return x-5
else
   return f(x+5)
endif
```

Why is this algorithm recursive?

☐ because it has a base case

☐ because it returns two different values

☐ because the function `f(x)` calls itself

☐ all of the above

4. What is the value of `f(0)` in Question 3?

☐ 100 ☐ 105 ☐ 95 ☐ 500

5. Consider the following algorithm:

```
Algorithm:  show(x)
Requires: a nonnegative integer x
if x>10
   return nil
else
   return cons(x,show(x+1))
endif
```

what is the output of calling `show(5)`?

☐ [9,8,7,6,5] ☐ [5,6,7,8,9] ☐ [10,9,8,7,6] ☐ [5,6,7,8,9,10]

6. Which of the following statements about the algorithm `show(x)` in Question 5 is TRUE?

☐ for any input value x≥10, the algorithm will return `nil`

☐ the recursive formula goes to infinity if x≥10

☐ for input value x=0 the recursion stops when x reaches 11

☐ none of the above

7. Which of the following mini card numbers is valid following the Luhn algorithm?

☐ 2395 ☐ 3003 ☐ 1215 ☐ 7161

8. Consider the following algorithm, which takes two nonempty lists as its input.

```
Algorithm:  dow(L1,L2)
Requires: two nonempty lists, L1 and L2
if isEmpty(L1) && isEmpty(L2)
   return True
elseif isEmpty(L1) || isEmpty(L2)
   return False
else
   return dow(tail(L1),tail(L2))
endif
```

What is the output of calling `dow([1,2],[1,2,3]`?

☐ [3] ☐ [1,2] ☐ True ☐ False

9. Consider the following algorithm:

```
*************************
1.  let x=10
2.  let y=20
3.  if (x=10) || (y=20)
4.      return x+y
5.  endif
```

In which line(s) do you spot a logical error?

☐ lines 1, 2 ☐ line 3 ☐ line 4 ☐ no error

10. Which of the following problems can't be solved using recursion?

☐ Factorial of a number

☐ Nth fibonacci number

☐ Length of a string

☐ Problems without base case

11. A _______ gate gives the output as 1 only if all the inputs signals are 1.

☐ AND

☐ OR

☐ XOR

☐ NOR

12. Can linear search recursive algorithm and binary search recursive algorithm be performed on an unordered list?

☐ Binary search can't be used

☐ Linear search can't be used

☐ Both cannot be used

☐ Both can be used

13. Which of the following statements about binary search on lists is <u>NOT</u> correct?

☐ binary search is faster than linear search ☐ binary search can only be applied to sorted lists

☐ binary search has time complexity $O(lg(n))$ ☐ binary search has time complexity $O(n/2)$

14. What is the formula for Euclidean algorithm?

- ☐ GCD (m,n) = GCD (n, m mod n)
- ☐ LCM(m,n)=LCM(n, m mod n)
- ☐ GCD(m,n,o,p) = GCD (m, m mod n, o, p mod o)
- ☐ LCM (m,n,o,p) = LCM (m, m mod n, o, p mod o)

15. What does the following algorithm do?

```
Algorithm: fun(n)
Requires: an integer n
Return: an integer
1. if (n == 0 || n == 1)
2. return n
3. elseif (n%3 != 0)
4. return 0;
5. else
6. return fun(n/3)
7. endif
```

- ☐ It returns 1 when n is a multiple of 3, otherwise returns 0
- ☐ It returns 1 when n is a power of 3, otherwise returns 0
- ☐ It returns 0 when n is a multiple of 3, otherwise returns 1
- ☐ It returns 0 when n is a power of 3, otherwise returns 1

Write your answers for each of questions 16-20 in the boxes provided.

16. Write a recursive algorithm called LCM(x,y) that finds the least common multiple of two positive integer numbers x and y. Follow your algorithm to compute LCM(12,15).

Algorithm: LCM(x,y)
Requires: two positive integers x, y.
Returns: LCM of x, y.

return LCMHelper(x, y, y)

Algorithm: LCMHelper(x, y, m)
Requires: three positive integers x, y, m
Returns: LCM of x, y

1. if m % x == 0
2. return m
3. else
4. return LCMHelper(x, y, m+y)
5. endif.

LCM(12, 15)
LCMHelper(12, 15, 15), 1: False
LCMHelper(12, 15, 30) 1: False
LCMHelper(12, 15, 45) 1: False
LCMHelper(12, 15, 60) 1: True
return 60

LCM(12, 15) = 60.

17. Write a recursive algorithm called `powerten(n)` that takes a nonnegative integer n and returns the value of 10^n. For example,

`powerten(0)=1, powerten(3)=1000.`

```
Algorithm: powerten(n)
Requires: integer n≥0
Returns: 10^n
    if n == 0
        return 1
    else
        return 10 * powerten(n-1)
    endif.
```

18. Write a recursive algorithm called `list2num(L)` that takes a non-empty list `L` of n nonnegative single-digit integers and returns the list into a n-digit number. For example,

`list2num([6])=6, list2num([5,3])=53, list2num([9,1,1,0])=9110.`

You can call the algorithm `powerten()` in Question 17, and call the algorithm `length()` to compute the length of the list.

```
Algorithm : list2_num(L)
Requires : a list L of n single digit integers .
Returns : an integer of n digits.
n =length(L)
if n==1 then
return value(L)
else
return value(L) * power(n-1) +list2_num(tail(L))
endif
```

19. Trace your algorithm in Question 18 for list2num([4,2,7]).

list2num([4,2,7])

L = [4,2,7]

1: n=3. 2: False 5: 4 * powerTen(2) + list2num([2,7])

($= 4\times10^2 + 27 = 427$)

L = [2,7]

1: n=2, 2: False 5: 2 * powerten(1) + list2num([7])

($= 2\times10^1 + 7 = 27$)

L = [7]

1: n=1, 2: True 3: return 7.

(backtracking)

list2num([4,2,7]) = 427

20. A list is made up of the following sequence of numbers $L = [13, 5, 2, 26, 54, 72, 9, 44]$. Show the step by-step outcomes involved when sorting this list using a insertion-sort algorithm.

Note: No need to write algorithm only intermediate outcomes to do insertion-sort.

	Unsorted	Sorted
Step1:	[13,5,2,26,54,72,9,44]	[ ]
Step2:	[5,2,26,54,72,9,44]	[13]
Step3 :	[2,26,54,72,9,44]	[5,13]
Step4 :	[26,54,72,9,44]	[2, 5,13]
Step5 :	[54,72,9,44]	[2,5,13,26]
Step6 :	[72,9,44]	[2,5,13,26,54]
Step7 :	[9,44]	[2,5,13,26,54,72]
Step8 :	[44]	[2,5,9,13,26,54,72]
Step9 :	[ ]	[2,5,9,13,26,44,54,72] Sorted