2016 EXAMINATIONS

Lancaster University

Part I (First Year)

SCHOOL OF COMPUTING AND COMMUNICATIONS

SCC.120 Fundamentals of Computer Science (1 hour & 30 Minutes)

- **Answer any <u>THREE</u> out of the four questions.**
- **Use a <u>separate</u> answer book for <u>each</u> question.**

Question 1

1.a Let A = { a, d, g, y}, B = { d, g, h, i, n, t}, C = { m, n, o} and D = { x, y}. Calculate:

i.	$A \cup B$	**[1 mark]**
ii.	$A \cap B$	**[1 mark]**
iii.	$A - B$	**[1 mark]**
iv.	$C \times D$	**[2 marks]**

1.b Let f(x) = 2x -3 and g(x) = 3x + 2. Calculate:

i.	f + g	**[1 mark]**
ii.	f * g	**[2 marks]**
iii.	$f \circ g$	**[2 marks]**
iv.	f^{-1}	**[2 marks]**
v.	Define f(x) recursively	**[3 marks]**

1.c For the following question, here are some equivalences:

- Associativity : $(P \vee Q) \vee R \Leftrightarrow P \vee (Q \vee R)$
- Commutivity : $(P \vee Q) \Leftrightarrow (Q \vee P)$
- Complement : $(P \vee True) \Leftrightarrow True$
- DeMorgan's Law : $\sim(P \wedge Q) \Leftrightarrow (\sim P \vee \sim Q)$
- And a further "simple" identity : $\sim Q \vee Q \Leftrightarrow True$

Using the equivalences to perform transformations, prove that $(a \vee \sim(a \wedge b))$ is a tautology. Show every step of the transformation and name the equivalence you are applying.

[6 marks]

Question 1 continues on the next page…

1.d Let P(x) be the statement "x can't dance", where the universe of discourse for x is the set of people. Express the following English statements as logical predicates:

i. There exists (at least) one person who can't dance. **[1 mark]**

ii. There exists (at least) one person who can dance **[1 mark]**

iii. Everyone can dance **[1 mark]**

Let P and Q be the propositions:

P: "Your car is out of petrol."

Q: "You can't drive your car."

Write the following propositions using P and Q and logical connectives.

iv. Your car is not out of petrol if you can drive it. **[1 mark]**

[Total 25 Marks]

Question 2

2.a Briefly compare the advantages and disadvantages of arrays and linked lists in terms of the storage capacity and accessing performance. For each type of data structure, give an example when it is best to use that type of data structure.

[4 marks]

2.b Briefly describe the main differences between objects and structures. Give one example to show why data encapsulation is important in practice when programming a queue.

[4 marks]

2.c We have a structure type called `node` that is defined as follows in C:

```
typedef struct _node {
      int val;
      struct _node *next;
}node;
```

Consider the following code:

```
 1  node* new(int v){
 2           node *n = malloc(sizeof(node));
 3           n->val = v;
 4           n->next = NULL;
 5           return n;
 6  }
 7
 8  node* createList(node* head){
 9          int i=0;
10          node  *p;
11          for (i=0; i<4; i++) {
12                p = new(i+1);
13                p->next = head;
14                head = p;
15          }
16          return head;
17  }
18
19  node* foo(){
20          node* head = NULL;
21          head = createList(head);
22          return head;
21  }
```

Question 2 continues on the next page…

Question 2 continued.

i. Explain what the `malloc()` function does at line 2.

[2 marks]

ii. Function *createList*() constructs a linked list consisting of four nodes. In this example pointer *head* would store the address of the first node of the linked list. Using "box-and-arrow" notation (as used throughout the course, with a general example shown below in Figure 1) to illustrate what the linked list looks like after executing function *foo().*

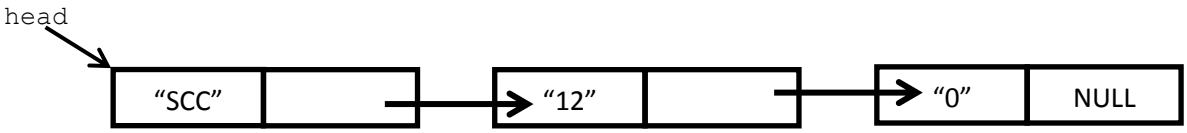


Figure 1. A general example for "box-and-arrow" diagrams. For question 2.c ii, your answer would have four boxes with integer values.

[5 marks]

iii. Recall the lecture on regions of memory, when a function is called, a stack frame will be pushed into a memory region called stack. What might be contained in the stack framework created when calling function *new()* in this example?

[3 marks]

2.d

i. Convert an array with values [12, 32, 42, 70, 12, 8, 65, 7] into a min-heap. Draw a figure to show the final result.

[2 marks]

ii. Convert the above array into a balanced binary search tree. Draw a figure to show the final result.

[2 marks]

iii. Explain the main difference between a min-heap and a binary search tree.

[3 marks]

[Total 25 Marks]

Question 3

3a. The powerset of a set A is a set B that contains all the subsets of A.

Imagine the best possible algorithm that computes the powerset of a set. State the following.

i. The size of the input to the algorithm **[2 marks]**
ii. The number of elements in the powerset of a set **[2 marks]**
iii. The best case complexity of the algorithm in Big O **[2 marks]**
iv. The worst case complexity of the algorithm in Big O **[2 marks]**

3b. For each of the following pairs of functions, state which one grows faster asymptotically for input of size *n* (in other words, which is worse in terms of efficiency).

i. f(n) = 1000n + 53 and g(n)= $\log_{10}(n)$ **[2 marks]**
ii. f(n) = n! and g(n) = $2^n + n$ **[2 marks]**
iii. f(n) = n^3 and g(n) = $2^n + n$ **[2 marks]**
iv. f(n) = log(n) and g(n) = square-root(n) **[2 marks]**

3c.

i. State the worst case complexity (in Big O) of algorithm *numCharacters(a,b)* which takes two strings a and b and returns the total number of their characters by using a constant time string length function.

[2 marks]

ii. For 3c(i) above, state the worst case complexity (in Big O) when you do not have a constant time string length function and must instead perform the counting character by character.

[2 marks]

iii. State the worst case complexities (in Big O) of binary search and linear search on an array of sorted integers.

[2 marks]

iv. Imagine an algorithm that searches for an integer in an array of integers by first using insertion-sort to sort the array and then using binary-search to search for the number. State the worst case complexity of this algorithm (in Big O).

[3 marks]

[Total 25 Marks]

Question 4

4.a

For a linear array with a circular buffer, what is the complexity in big O notation for adding an item and the complexity in big O notation for removing an item? Explain your answers.

[4 marks]

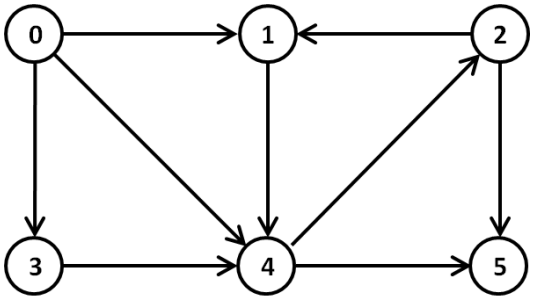
4.b

You are given two priority queues with n integers each. Describe in words how you would efficiently merge them into one priority queue. What is the time complexity of your method in big O notation?

[4 marks]

4.c

Consider the directed graph **G** shown below:



i. State whether **G** is non-connected, weakly connected, or strongly connected. Explain your answer.

[2 marks]

ii. Write down the order in which the nodes are visited during depth-first traversal, starting at node 0. Include the symbol ↑ wherever the traversal is forced to back up to a previous node (as covered in lecture). You can assume that adjacent nodes are selected in increasing order of their value.

[4 marks]

iii. How many arcs are there in the complementary graph of **G**?

[2 marks]

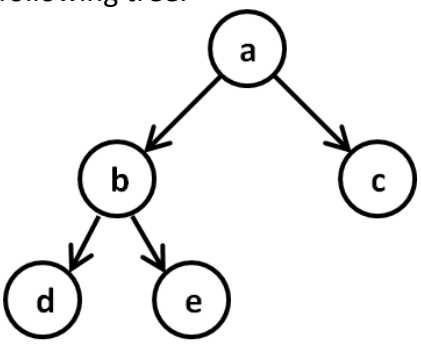
iv. How many subgraphs of **G** are there that have exactly two nodes? Explain your answer.

[2 marks]

Question 4 continues on the next page…

4.d

This question is about the following tree.



i. What is the sequence of nodes visited by the postorder traversal?

[2 marks]

ii. How many zeros are there in the adjacency matrix representation of this tree?

[2 marks]

iii. You are given a pointer to the root node of the above tree. Describe a method to create a new tree where for each node (in the whole tree) you swap the left and right child nodes.

[3 marks]

[Total 25 Marks]

END OF PAPER