Office Use Only

MONASH University

Semester One 2016
Examination Period

Faculty of Information Technology

EXAM CODES: **FIT1045**
TITLE OF PAPER: **INTRODUCTION TO ALGORITHMS AND PROGRAMMING – PAPER 1**
EXAM DURATION: 3 hours writing time
READING TIME: 10 minutes

THIS PAPER IS FOR STUDENTS STUDYING AT: (tick where applicable)

☐ Berwick	☑ Clayton	☑ Malaysia	☐ Off Campus Learning	☐ Open Learning
☐ Caulfield	☐ Gippsland	☐ Peninsula	☐ Monash Extension	☐ Sth Africa
☐ Parkville	☐ Other (specify)			

During an exam, you must not have in your possession any item/material that has not been authorised for your exam. This includes books, notes, paper, electronic device/s, mobile phone, smart watch/device, calculator, pencil case, or writing on any part of your body. Any authorised items are listed below. Items/materials on your desk, chair, in your clothing or otherwise on your person will be deemed to be in your possession.

No examination materials are to be removed from the room. This includes retaining, copying, memorising or noting down content of exam material for personal use or to share with any other person by any means following your exam.
Failure to comply with the above instructions, or attempting to cheat or cheating in an exam is a discipline offence under Part 7 of the Monash University (Council) Regulations.

AUTHORISED MATERIALS

OPEN BOOK	☐ **YES**	☑ **NO**
CALCULATORS	☐ **YES**	☑ **NO**
SPECIFICALLY PERMITTED ITEMS	☐ **YES**	☑ **NO**

if yes, items permitted are:

Candidates must complete this section if required to write answers within this paper

STUDENT ID: __ __ __ __ __ __ __ __ DESK NUMBER: __ __ __ __ __

Page		Mark
3		3
5		3
7		5
9		4
11		5
13		6
15		10

Page		Mark
17		8
19		5
21		6
23		8
25		10
27		7
Total:		80

Blank Page for Working

For Questions 1-6 circle only one letter for each question corresponding to the correct response.

Question 1 [1 mark]

Prim's algorithm finds a minimum spanning tree by:

(a) Beginning with all vertices and no edges and iteratively adding edges that don't create cycles.
(b) Beginning with a single vertex and iteratively adding edges with one vertex in the tree and the other vertex not in the tree.
(c) Both a and b.
(d) None of the above.

Question 2 [1 mark]

What does the following code do?

```
for item in zombie_movies:
    if target == item:
        print(item)
```

(a) Always prints the first item in the database.
(b) Prints every item in the database which is identical to the target.
(c) Prints all the movie titles that do not contain the target.
(d) None of these.

Question 3 [1 mark]

Suppose you have the following function:

```
def triple(aList):
    for k in range(len(aList)):
        aList[k] *= 3
```

What is printed by the following code?

```
aList = ['a','b','c']
triple(aList)
print(aList)
```

(a) ['aaa','bbb','ccc']
(b) ['a','b','c']
(c) [‘a’, ‘a’, ‘a’, ‘b’, ‘b’, ‘b’, ‘c’, ‘c’, ‘c’]
(d) None of these.

3

Blank Page for Working

Question 4 [1 mark]

What does the following code print?

```
x = 5
y = 7
result = 1
while x > 0:
  result = result * 2
  if x % 2 == 1:
     result = result + y
  x = x // 2
print(result)
```

(a) 32
(b) 35
(c) 43
(d) None of these.

Question 5 [1 mark]

What is the minimum index that the partition function in the quick sort algorithm can return?

(a) 0
(b) 1
(c) 2
(d) None of these.

Question 6 [1 mark]

Which of the following statements best describes the invariant for the main loop for sorting a list into increasing order using insertion sort?

(a) At the kth iteration the first k items in the list are the largest items in the list.
(b) At the kth iteration the first k items in the list are sorted.
(c) At the kth iteration the first k items in the list are sorted and are also the minimum items in the list.
(d) None of these.

3

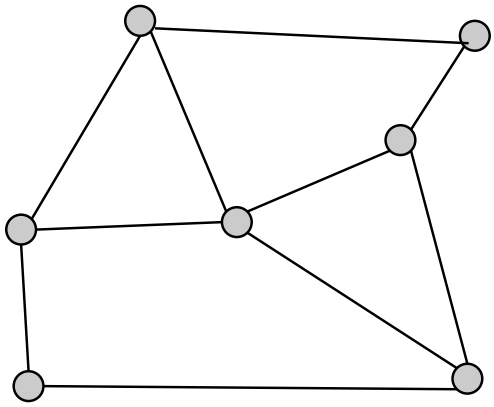
Blank Page for Working

Question 7 [2 + 2 + 1 = 5 marks]

(a) Give a definition of an algorithm.

(b) Give a definition of a Hamiltonian cycle in a connected graph G.

(c) Find a Vertex Cover of size **4** for the following graph. Circle the vertices that belong to the Vertex Cover.



5

Blank Page for Working

Question 8 [1 + 1 + 1 + 1 = 4 marks]

This question is about *time complexity*. For each of the given Python functions, state the time complexity in big O notation and provide a brief explanation.

(a)

```
def product_func(n):
    product = 1
    for k in range(n):
        for j in range(5):
            product *= num
    return product
```

(b)

```
def total_func(n):
    total = 0
    for k in range(n):
        for j in range(n):
            total += num
    return total
```

(c)

```
def another_total_func(n):
    total = 0
    for k in range(n):
        total += k
    for k in range(n+1):
        total += num
    return total
```

(d)

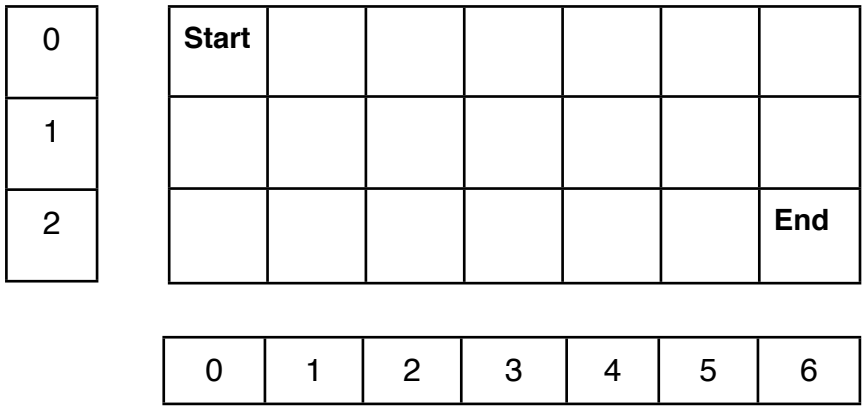
```
def division_func(n):
    product = 1
    while n > 1:
        product *= 2
        n //= 2
    return product
```

4

Blank Page for Working

Question 9 [5 marks]

A robot is located at the top-left corner of a **3x7** grid. The robot can only move **down one square** or **right one square** at any point in time and cannot move in any other direction. The robot is trying to reach the bottom-right corner of the grid.



Write a Python function, **countPaths**, which returns a table (a list of lists), **paths**. Each item in the table **paths** is the number of possible paths the robot can take from the **Start** position to the position corresponding to the item in the table. For example **paths[0][0] = 1** and **paths[2][6] = 28**.

5

Blank Page for Working

Question 10 [1 + 5 = 6 marks]

The Catalan numbers, **C(n)**, are defined by the following relations.

C(0) = 1 and C(n) = C(0)C(n-1) + C(1)C(n-2) + .. + C(n-2)C(1) + C(n-1)C(0)

So, **C(1) = C(0)C(0)** and **C(2) = C(0)C(1) + C(1)C(0)**.

(a) Give the value of **C(3)**.

(b) Write a recursive Python function which has as input a non-negative integer, **n**, and returns **C(n)**.

6

Blank Page for Working

Question 11 [5 + 5 = 10 marks]

The maximum sublist product problem involves looking for a slice of a list which has the largest product of the items within it.

For example if **aList = [1, -2, 3, -4, -3]** then **aList[2:4]** is the slice with the largest product of items.

This problem would be trivial if the list contained only positive numbers but in this task the list can contain positive and negative integers, and zeros. You can also assume that the list contains at least 1 number.

(a) Write a Python function, **product,** to return the product of the items in the list **aList**.

For example, given **aList=[1,2,3,4,5]** then **product(aList[1:3])** would return 24.

(b) Assuming you have the function **product,** write a Python function, **maxProduct,** to return the largest product of a slice in the given list of integers, **aList**.

10

Blank Page for Working

Question 12 [2 + 2 + 2 + 2 = 8 marks]

Consider the following type of items.

Type	1	2	3	4
Value	$12	$30	$11	$42
Weight	4 kg	3 kg	2 kg	5 kg

Assume you are only allowed to have at most one item of each type. Find which items you should take in the knapsack that has a capacity of 10kg, so that you maximize the total value of the items you have taken.

(a) Describe a greedy approach for finding which items you should take.

(b) Using the greedy approach you stated in **Part (a)**, state which items you would take.

(c) Describe the solution you gave in **Part (b)** as a bit list

(d) Write a Python function, **printItems**, which takes a bit list (corresponding to the above problem) as input and prints the items represented by this bit list.

8

Blank Page for Working

Question 13 [5 marks]

Write a Python function, **mergeLists,** which takes two sorted lists (sorted in ascending order) as input and merges the two sorted lists into a single sorted list, and returns this list.

5

Blank Page for Working

Question 14 [6 marks]

Insert the following numbers, in the order they appear, into a Heap. You are allowed to choose whether the Heap is a min-Heap or a max-Heap.

3, 1, 12, -2, 14, 3

Show the Heap after each number has been inserted. The answer should consist of **6 Heaps.**

6

Blank Page for Working

Question 15 [2 + 6 = 8 marks]

Consider the problem of placing 4 Queens on a 4x4 board so that no Queen attacks another Queen.

a) Describe how a partial solution can be represented as list of numbers.

b) Show how you could use backtracking to solve this problem.

In particular, show a search tree and indicate on your diagram for each position the corresponding partial solution (represented as a list of numbers). Once you have found the solution you do not need to show any more of the search tree.

8

Blank Page for Working

Question 16 [2 + 2 + 3 + 3 = 10 marks]

(a) Give a definition of what it means for a problem to be NP-complete.

(b) Give **2** examples of NP problems given in lectures and state their certificates.

(c) Describe the Halting problem.

(d) Describe what the P = NP problem is.

10

Blank Page for Working

Question 17 [1 + 1 = 2 Marks]

(a) Give an example of a sorting method that does not use divide and conquer.

(b) Give an example of a sorting method that does use divide and conquer.

Question 18 [5 Marks]

Write a Python function, **isClique,** that checks whether a list of vertices, **vertexList,** is a clique in a given graph.

The graph is represented as an adjacency matrix, **graphTable,** where **graphTable[j][k] = 1** if vertex **j** is adjacent to vertex **k**. The function, **isClique**, takes **vertexList** and **graphTable** as input and returns **True** if the vertices in **vertexList** form a clique in the graph represented by **graphTable**, and returns **False** otherwise.

7

END OF EXAM