

M269 Exams

Prsntn 2016J Exam

Contents

1	Prsntn 2016J Exam Qs	2
1.1	M269 2016J Exam Qs	2
1.2	M269 2016J Exam Q Part1	2
1.3	M269 2016J Exam Q 1	2
1.4	M269 2016J Exam Q 2	3
1.5	M269 2016J Exam Q 3	3
1.6	M269 2016J Exam Q 4	4
1.7	M269 2016J Exam Q 5	4
1.8	M269 2016J Exam Q 6	5
1.9	M269 2016J Exam Q 7	6
1.10	M269 2016J Exam Q 8	6
1.11	M269 2016J Exam Q 9	6
1.12	M269 2016J Exam Q 10	7
1.13	M269 2016J Exam Q 11	7
1.14	M269 2016J Exam Q 12	7
1.15	M269 2016J Exam Q 13	8
1.16	M269 2016J Exam Q 14	9
1.17	M269 2016J Exam Q 15	10
1.18	M269 2016J Exam Q Part2	10
1.19	M269 2016J Exam Q 16	10
1.20	M269 2016J Exam Q 17	12
2	Prsntn 2016J Exam Solns	12
2.1	M269 2016J Exam Solns	12
2.2	M269 2016J Exam Soln Part1	12
2.3	M269 2016J Exam Soln 1	12
2.4	M269 2016J Exam Soln 2	13
2.5	M269 2016J Exam Soln 3	13
2.6	M269 2016J Exam Soln 4	14
2.7	M269 2016J Exam Soln 5	15
2.8	M269 2016J Exam Soln 6	15
2.9	M269 2016J Exam Soln 7	15
2.10	M269 2016J Exam Soln 8	15
2.11	M269 2016J Exam Soln 9	16
2.12	M269 2016J Exam Soln 10	16
2.13	M269 2016J Exam Soln 11	16
2.14	M269 2016J Exam Soln 12	17
2.15	M269 2016J Exam Soln 13	17
2.16	M269 2016J Exam Soln 14	18
2.17	M269 2016J Exam Soln 15	18
2.18	M269 2016J Exam Soln Part2	18
2.19	M269 2016J Exam Soln 16	18
2.20	M269 2016J Exam Soln 17	19

1 Prsntn 2016J Exam Qs

1.1 M269 2016J Exam Qs

- M269 Algorithms, Data Structures and Computability
- Presentation 2016J Exam
- Date Wednesday, 7 June 2017 Time 14:30–17:30
- There are **TWO** parts to this examination. You should attempt all questions in **both** parts
- **Part 1** carries **65 marks** — **80 minutes**
- **Part 2** carries **35 marks** — **90 minutes**
- **Note** see the original exam paper for exact wording and formatting — these slides and notes may change some wording and formatting
- **Note** 2015J and before had Part 1 with 60 marks (100 minutes), Part 2 with 40 marks (70 minutes)

[Go to Exam Solns](#)

1.2 M269 2016J Exam Q Part1

- Answer every question in this part.
- The marks for each question are given below the question number.
- Answers to questions in this Part should be written **on this paper** in the spaces provided, or in the case of multiple-choice questions you should tick the appropriate box(es).
- If you tick **more** boxes than indicated for a multiple choice question, you will receive **no** marks for your answer to that question.
- Use the provided answer books for any rough working.

[Go to Soln Part1](#)

1.3 M269 2016J Exam Q 1

- Which **two** of the following statements are true? (Tick **two** boxes.) **(2 marks)**
- A. A problem is computable if it possible to build an algorithm which solves any instance of the problem in a finite number of steps.
 - B. An effective procedure is an algorithm which, for every instance of a given problem, solves that instance in the most efficient way — minimising the use of resources such as memory.
 - C. A decision problem is decidable if it is computable.
 - D. A decision problem is any problem stated in a formal language.

[Go to Soln 1](#)

1.4 M269 2016J Exam Q 2

- Complete these paragraphs correctly using words or phrases from the list below.
(2 marks)
- Abstraction as _____ can be understood in terms of the relationship between a _____ and a _____. The latter represents the details of interest and captures the essentials, ignoring certain irrelevant details.
- Abstraction as _____ generally involves two layers — the _____ (which is a layer through which users interact with the model) and the _____ (a layer that automates the model)
- Possible words and phrases to insert:
encapsulation model modelling procedural
algorithm process automation interface
part of reality data simulation implementation

[Go to Soln 2](#)

1.5 M269 2016J Exam Q 3

- This question is about bubble sort and selection sort, where we are sorting numbers in **ascending** order.
(6 marks)
- (a) Selection sort improves on bubble sort by making only one exchange for every pass through the list.
- In selection sort, given the starting list below, indicate which two elements are to be swapped at each stage, and complete below as necessary.
- You have space to indicate up to 5 swaps and the resulting list.
- If selection sort requires fewer than 5 swaps for this list, leave any remaining step(s) blank.

1	6	2	3	5
---	---	---	---	---

1. Swap elements _____ and _____ to give

--	--	--	--	--

2. Swap elements _____ and _____ to give

--	--	--	--	--

3. Swap elements _____ and _____ to give

--	--	--	--	--

4. Swap elements _____ and _____ to give

--	--	--	--	--

5. Swap elements _____ and _____ to give

--	--	--	--	--

- (b) Although both bubble sort and selection sort make the same number of comparisons for a list of the same length, they do not make the same number of swaps.

How many swaps are made in a **worst** case, with a list of length 5, for each of bubble sort and selection sort?

Explain how you arrived at the number of swaps for each. There is no need to refer to Big-O in your answer.

[Go to Soln 3](#)

1.6 M269 2016J Exam Q 4

- A Python program contains a loop with the following guard (4 marks)

```
while a <= 3 or b > 8:
```

Make the following substitutions:

P represents $a > 3$

Q represents $b \leq 8$

Complete the following table

P	Q	$\neg P$	$\neg Q$	$\neg P \vee \neg Q$	$P \vee Q$	$\neg(P \wedge Q)$
T	T					
T	F					
F	T					
F	F					

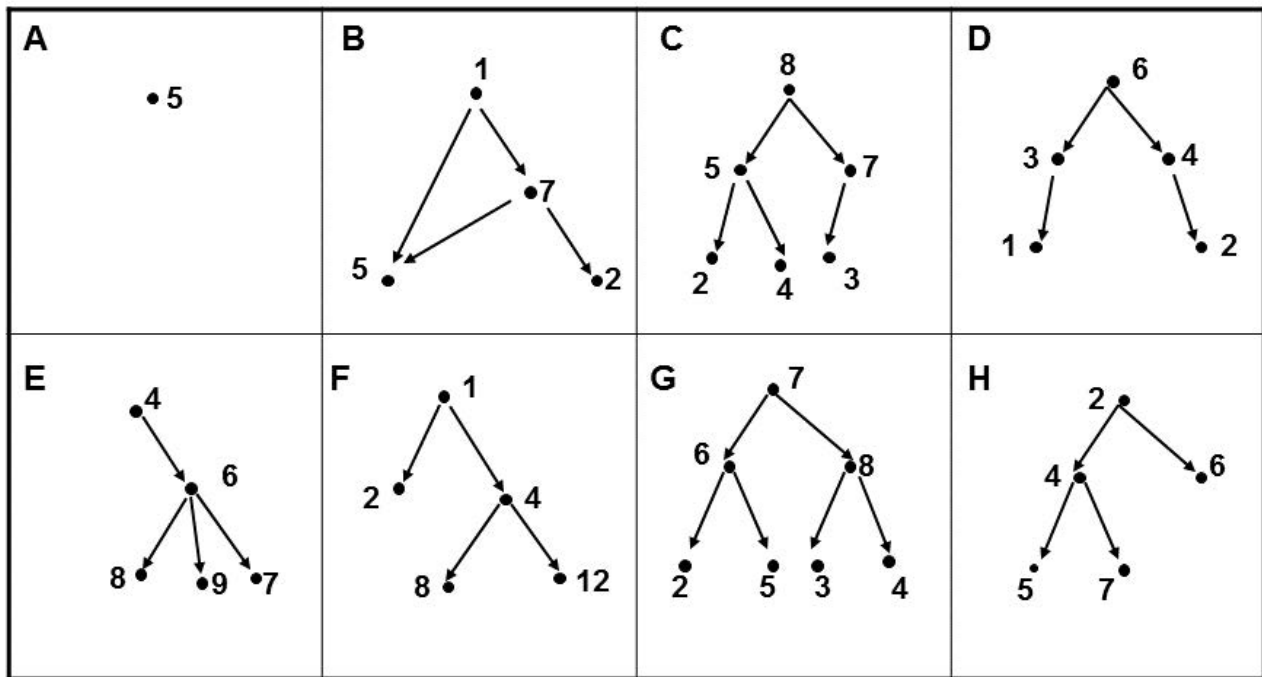
- Based on the table, which of the following expressions is equivalent to the above guard? (Tick **one** box.)

- A. **not** $a < 3$
- B. **not** $b \leq 8$
- C. **not** ($a \leq 3$ **and** $b > 8$)
- D. $a > 3$ **and** $b \leq 8$
- E. **not** ($a > 3$ **and** $b \leq 8$)

[Go to Soln 4](#)

1.7 M269 2016J Exam Q 5

- Consider the diagrams in A-H, where nodes are represented by black dots and edges by arrows. The numbers are the keys for the corresponding nodes.



- On the following lines, write the letter(s) of the diagram(s) that satisfies (satisfy) the condition, or write "None" if no diagram satisfies the condition. **(4 marks)**

- Which of **A**, **B**, **C** and **D**, if any, are **not** a tree?
- Which of **E**, **F**, **G** and **H**, if any, are binary trees?
- Which of **C**, **D**, **G** and **H**, if any, are complete binary trees?
- Which of **C**, **D**, **G** and **H**, if any, are (min or max) heap?

[Go to Soln 5](#)

1.8 M269 2016J Exam Q 6

- Consider the following function, which takes a list as an argument.

```

1 def someFunction(aList):
2     n = len(aList)
3     counterOne = 0
4     counterTwo = 0
5     for i in range(n):
6         counterOne = counterOne + 1
7         for j in range(n):
8             counterTwo = counterTwo + 1
9             for k in range(n):
10                counterOne = counterOne + 1
11                counterTwo = counterTwo + 1
12     return counterOne + counterTwo

```

- From the options below, select the two that represent the correct combination of $T(n)$ and Big-O complexity for this function.

You may assume that a step (i.e. the basic unit of computation) is the assignment statement.

- A. $T(n) = 4n + 3$ i. $O(1)$
 B. $T(n) = 2n^3 + n^2 + n + 3$ ii. $O(n)$
 C. $T(n) = 2n^2 + n + 3$ iii. $O(n^2)$
 D. $T(n) = n^3 + n^2 + n + 3$ iv. $O(n^3)$
 E. $T(n) = 3 \log n + n^3 + n^2 + n + 3$ v. $O(\log n)$

- Explain how you arrived at $T(n)$ and the associated Big-O

[Go to Soln 6](#)

1.9 M269 2016J Exam Q 7

- (a) Which **two** of the following statements are true? (Tick **two** boxes.) (4 marks)

- A. Hash tables are an implementation of Map ADTs because they are searchable structures that contain key-value pairs, which allow searching for the key in order to find a value.
- B. Chaining, where a slot in the hash table may be associated with a collection of items, is a standard way of implementing hash functions.
- C. Clustering occurs when the number of unoccupied slots in a hash table exceeds the number of occupied slots.
- D. The efficiency of inserting new items into a hash table decreases as the load factor becomes greater.

- (b) Calculate the load factor for the hash table below. Show your working.

A	Q		S	F		U			N
0	1	2	3	4	5	6	7	8	9

[Go to Soln 7](#)

1.10 M269 2016J Exam Q 8

- (a) Lay out the keys [51, 22, 73, 65, 81, 92] as a Binary Search Tree, adding the nodes in the order in which they appear in the list, i.e. starting with 51 as the root node.
- (b) Label each node with its balance factor. Is the tree balanced? Explain. (5 marks)

[Go to Soln 8](#)

1.11 M269 2016J Exam Q 9

- (a) Consider the food web in a certain ecosystem. It can be modelled by a graph in which each node represents an animal or plant species, and where an edge indicates that one species eats another species.

For a **typical** food web, e.g. all animals and plants living in and around a lake, the graph is _____ (choose from UNDIRECTED/DIRECTED) because

insert answer here

(b) Is an adjacency matrix a good data structure for a sparse graph? Explain. (4 marks)

[Go to Soln 9](#)

1.12 M269 2016J Exam Q 10

- The graph showing the dependencies of tasks in a project has been lost. The project manager remembers that there were 5 tasks (let's call them A, B, C, D and E) and that ABCDE and ABEDC were not possible schedules (i.e. topological sorts of the graph), but ABDEC and ADBEC were.
- Draw a directed acyclic graph that is compatible with the given information.
- Each node has to be connected to or from at least one other node. (4 marks)

[Go to Soln 10](#)

1.13 M269 2016J Exam Q 11

(a) In propositional logic, a tautology is a well-formed formula (WFF) that is TRUE in every possible interpretation.

- It follows that if a WFF is a tautology, it is satisfiable.
- Explain what “satisfiable” means, and why a tautology must be satisfiable.

(b) The following WFF is satisfiable. Complete the truth table.

$$(P \vee Q) \rightarrow Q$$

P	Q	$(P \vee Q)$	$(P \vee Q) \rightarrow Q$
T	T		
T	F		
F	T		
F	F		

- State whether the WFF is a tautology or not, and explain why. (4 marks)

[Go to Soln 11](#)

1.14 M269 2016J Exam Q 12

- A particular interpretation of predicate logic allows facts to be expressed about people and their pets. Some of the assignments in the interpretation are given below (where the symbol $\mathcal{I}$ is used to show assignment).
- The domain of individuals is $\mathcal{D} = \{\text{Clara, Nicky, Mark, Rex, Fifo, Henny, Admiral}\}$.
- The constants *clara*, *nicky*, *mark*, *rex*, *fifo*, *henny* and *admiral* are assigned to the individuals Clara, Nicky, Mark, Rex, Fifo, Henny and Admiral respectively.
- Four unary predicate symbols are assigned to individuals as follows:

- $I(\textit{person}) = \{\textit{Clara}, \textit{Nicky}, \textit{Mark}\}$
 - $I(\textit{pet}) = \{\textit{Rex}, \textit{Fifo}, \textit{Henny}, \textit{Admiral}\}$
 - $I(\textit{dog}) = \{\textit{Rex}, \textit{Fifo}\}$
 - $I(\textit{chicken}) = \{\textit{Henny}\}$
- Two further predicate symbols are assigned binary relations as follows:
 - $I(\textit{has-pet}) = \{(\textit{Nicky}, \textit{Rex}), (\textit{Nicky}, \textit{Fifo}), (\textit{Mark}, \textit{Henny})\}$
 - $I(\textit{feeds}) = \{(\textit{Clara}, \textit{Rex}), (\textit{Nicky}, \textit{Fifo})\}$
 - On the next page, you will be asked whether a given sentence is true or false. In your explanation, you need to consider any relevant values for the variables, and show, using the domain and interpretation above, whether they make the quantified expression TRUE or FALSE.
 - **In your answer, when you explain why a sentence is true or false, make sure that you use formal notation.** So instead of stating that “*Henny is a chicken in the interpretation*”, write $\textit{Henny} \in I(\textit{chicken})$. Similarly, instead of “*Henny is not a dog*” you would need to write $\textit{Henny} \notin I(\textit{dog})$ **(6 marks)**
- (a) Consider the following sentence in English: “*All dogs are Nicky’s pets*”. Which **one** well-formed formula is a translation of this sentence into predicate logic?
- A. $\forall X.(\textit{dog}(X) \wedge \textit{has-pet}(\textit{nicky}, X))$
- B. $\forall X.(\textit{dog}(X) \rightarrow \textit{has-pet}(\textit{nicky}, X))$
- C. $\exists X.(\textit{dog}(X) \wedge \textit{has-pet}(\textit{nicky}, X))$
- (b) Give an appropriate translation of the well-formed formula $\forall X. \exists Y. (\textit{dog}(X) \rightarrow \textit{feeds}(Y, X))$ into English

- This well-formed formula is ____ (*choose from TRUE/FALSE*), under the interpretation on the previous page, because:

[Go to Soln 12](#)

1.15 M269 2016J Exam Q 13

- A database contains the following tables, *lawnmower* and *brand*. **(6 marks)**

lawnmower

<i>make</i>	<i>model</i>	<i>type</i>
MowIt	Bella	push
MowIt	Speedy	electric
Mamouth	Kodiak	petrol
Mamouth	Pachyderm	petrol
Blades	Meadow	petrol
Blades	Nibble	robot
Blades	Yard	electric

brand

<i>manufacturer</i>	<i>location</i>
Mamouth	France
MowIt	USA
Blades	China
Scythes	China

(a) For the following SQL query, give the table returned by the query.

```
SELECT make, model
FROM lawnmower
WHERE type = 'electric';
```

- Write the question that the above query is answering.

(b) Write an SQL query that answers the question *Which lawnmowers are from manufacturers located in China?* The answer should be the following table:

<i>manufacturer</i>	<i>model</i>
Blades	Meadow
Blades	Nibble
Blades	Yard

[Go to Soln 13](#)

1.16 M269 2016J Exam Q 14

- Which **two** of the following statements are true? (Tick **two** boxes.) (4 marks)

- A. If a decision problem is in NP, then it is computable.
- B. The complexity of an algorithm that solves a problem places a lower bound on the complexity of the problem itself.
- C. If the best algorithm we currently have for solving a decision problem has complexity $O(2^n)$, then we know that problem can't be in P.
- D. If an NP-hard problem A can be Karp-reduced to a problem B, then problem B is NP-hard too.
- E. Every NP-hard problem is also NP-complete.

[Go to Soln 14](#)

1.17 M269 2016J Exam Q 15

- Consider the following decision problems: (4 marks)

1. The 3SAT Problem
2. Is a given list of numbers already sorted?
3. The Totality Problem
4. Is a given path from A to B in a given undirected graph the shortest path from A to B?

- For each of the following groups of problems, write on the line the numbers of any of the above problems that belong to that group, or write “*none*” if none of the above problems belongs to that group.

- (a) undecidable
- (b) tractable
- (c) NP-complete

[Go to Soln 15](#)

1.18 M269 2016J Exam Q Part2

- Answer every question in this Part.
- Answers to questions in this Part must be written in the separate **answer books**, which you should also use for your rough working.

[Go to Soln Part2](#)

1.19 M269 2016J Exam Q 16

- **Question 16** (20 marks)

- Consider an ADT for **undirected** graphs, named UGraph, which includes these two operations:
- **nodes**, which returns a sequence of all nodes in the graph, in no particular order;
- **neighbours**, which takes a node and returns a sequence of all its adjacent nodes, in no particular order.
- How each node is represented is irrelevant.

- (a) The following stand-alone Python function checks if a graph has a loop (an edge from a node to itself), assuming that UGraph is implemented as a Python class.

```
def hasLoop(graph):  
    for node in graph.nodes():  
        if node in graph.neighbours(node):  
            return True  
    return False
```

- Assume that the if-statement guard does a linear search of the sequence returned by `neighbours`.
 - If the graph has no node with a loop, is that a best-, average-, or worst-case scenario for `hasLoop`?
 - Assuming the graph has n nodes and e edges, what is the Big-O complexity of that scenario? Justify your answers.
 - Note that the complexity is in terms of how many nodes and edges `hasLoop` visits, because it has no assignments. **(5 marks)**
- (b) A node is *isolated* if it has no adjacent nodes. Isolated nodes cannot be reached from any other node and hence won't be processed by some graph algorithms.
- It is therefore useful to first check if a graph has isolated nodes.
- (i) Specify the problem of finding all isolated nodes in an undirected graph by completing the following template.
- Note that `isolatedNodes` is specified as an independent problem, not as a `UGraph` operation.
 - You may write the specification in English and/or formally with mathematical notation. **(4 marks)**

Name: `isolatedNodes`

Inputs:

Outputs:

Preconditions

Postconditions

- (ii) If instead of being an independent problem, `isolatedNodes` were an operation of the `UGraph` ADT, would it be a creator, inspector or modifier? Explain why. **(2 marks)**
- (iii) Give your initial insight for an algorithm that solves the problem, using the ADT's operations. **(4 marks)**
- (c) The ACME company used Prim's algorithm to connect its data centres with the least amount of fibre optic cable necessary.
- One of the centres is a gateway to the Internet.
 - ACME wants to know the maximum latency for an Internet message to reach any centre.
 - In other words, they want to know which centre is the furthest away from the gateway and what is the distance.
 - State and justify which data structure(s) and algorithm(s) you would adopt or adapt to solve this problem efficiently.
 - State explicitly any assumptions you make. **(5 marks)**

[Go to Soln 16](#)

1.20 M269 2016J Exam Q 17

- Imagine you have been invited to write a guest post for a technology blog, aimed at interested readers who know little about computing.
 - Write a draft of your blog post, which will explain relational databases and the formal logic that underpins them. **(15 marks)**
 - It should have
1. A suitable title and a short paragraph '*setting the scene*' by explaining the practical importance of relational databases.
 2. A paragraph describing in layperson's terms what a relational database is and how it's organised.
 3. A paragraph describing in layperson's terms what predicate logic is and its relationship with relational databases.
 4. A concluding paragraph stating your view on the importance, or not, of information technologies having a formal logic basis.
- Note that marks will be awarded for a clear coherent text that is appropriate for its audience, so avoid unexplained technical jargon and abrupt changes of topic, and make sure your sentences fit together to tell an overall '*story*' to the reader.
 - You may wish to use examples in your text to help explain the concepts.
 - As a guide, you should aim to write roughly three to five sentences per paragraph.

[Go to Soln 17](#)

2 Prsntn 2016J Exam Solns

2.1 M269 2016J Exam Solns

- The solutions given below are not official solutions
- For some questions, alternatives are given — a student would only have to provide one

[Go to Exam Qs](#)

2.2 M269 2016J Exam Soln Part I

- Part I solutions

[Go to Q Part I](#)

2.3 M269 2016J Exam Soln 1

- A. A problem is computable if it possible to build an algorithm which solves any instance of the problem in a finite number of steps. **Yes**

- B. An effective procedure is an algorithm which, for every instance of a given problem, solves that instance in the most efficient way — minimising the use of resources such as memory. **No** *An effective procedure is an algorithm that solves any instance of a decision problem in a finite number of steps (Reader, page 91)*
- C. A decision problem is decidable if it is computable. **Yes**
- D. A decision problem is any problem stated in a formal language. **No** *Problems where the answer is yes or no (Unit 1)*

[Go to Q 1](#)

2.4 M269 2016J Exam Soln 2

- Abstraction as **modelling** can be understood in terms of the relationship between a **part of reality** and a **model**.

The latter represents the details of interest and captures the essentials, ignoring certain irrelevant details.

- Abstraction as **encapsulation** generally involves two layers — the **interface** (which is a layer through which users interact with the model) and the **implementation** (a layer that automates the model).

[Go to Q 2](#)

2.5 M269 2016J Exam Soln 3

- Selection sort: sorting ascending and selecting largest first

```
def selSortAscByMax(xs):
    for fillSlot in range(len(xs) - 1, 0, -1):
        maxIndex = 0
        for index in range(1, fillSlot + 1):
            if xs[index] > xs[maxIndex]:
                maxIndex = index

        temp = xs[fillSlot]
        xs[fillSlot] = xs[maxIndex]
        xs[maxIndex] = temp
```

- Here is an informal version

```
for fillSlot = len(xs) - 1 down to 1 do
    find the maximum of
        xs[0] .. xs[fillSlot]
    and swap with xs[fillSlot]
```

1	6	2	3	5
---	---	---	---	---

- Swap elements 6 and 5 to give

1	5	2	3	6
---	---	---	---	---

- Swap elements 5 and 3 to give

1	3	2	5	6
---	---	---	---	---

- Swap elements 3 and 2 to give

1	2	3	5	6
---	---	---	---	---

4. Swap elements 2 and 2 to give

1	2	3	5	6
---	---	---	---	---

- Note the last swap would not be there if there was a test for `fillSlot == maxIndex`
- Selection sort: sorting ascending and selecting smallest first

```
def selectionSort(xs):
    for fillSlot in range(0, len(xs)-1):
        minIx = fillSlot
        for ix in range(fillSlot + 1, len(xs)):
            if xs[ix] < xs[minIx]:
                minIx = ix

        # if fillSlot != minIx: # swap if different
        xs[fillSlot], xs[minIx] = xs[minIx], xs[fillSlot]
```

- Here is an informal version

```
for fillSlot = 0 to (len(xs) - 2) do
    find the minimum of
        xs[fillSlot]..xs[len(xs) - 1]
    and swap with xs[fillSlot]
```

1	6	2	3	5
---	---	---	---	---

1. Swap elements 1 and 1 to give

1	6	2	3	5
---	---	---	---	---

2. Swap elements 6 and 2 to give

1	2	6	3	5
---	---	---	---	---

3. Swap elements 6 and 3 to give

1	2	3	6	5
---	---	---	---	---

4. Swap elements 6 and 5 to give

1	2	3	5	6
---	---	---	---	---

- Note the swap at stage 1. would not be there if there was a test for `fillSlot == maxIx`
- (b) Bubble sort does 10 swaps in a worst case since it does $n - 1$ swaps iterating over n items so total = $4 + 3 + 2 + 1 = 10$ swaps
- Selection sort does 4 swaps in a worst case since it does (at most) one swap per pass and $n - 1$ passes

2.6 M269 2016J Exam Soln 4

P	Q	$\neg P$	$\neg Q$	$\neg P \vee \neg Q$	$P \vee Q$	$\neg(P \wedge Q)$
T	T	F	F	F	T	F
T	F	F	T	T	T	T
F	T	T	F	T	T	T
F	F	T	T	T	F	T

- The equivalent expression is E.

[Go to Q 4](#)

2.7 M269 2016J Exam Soln 5

- (a) **B** is not a tree since node 5 has two parents — **A** is a node with two empty sub-trees
- (b) **F, G, H** are binary trees — **E** is not a binary tree since node 6 has three sub-trees
- (c) **C, G, H** are complete binary trees — **D** is not a complete binary tree since the last level is not filled from left to right
- (d) **C** is a max heap, **H** is a min heap — **G** is not a heap since node 8 is greater than node 7

[Go to Q 5](#)

2.8 M269 2016J Exam Soln 6

- Options **B** and **IV**
- There are three levels of nested loops with each loop executing n times.
- The innermost loop has 2 assignments giving $2n^3$ assignments
- The middle loop has one assignment giving a further n^2 assignments
- The outer loop has one assignment giving n assignments
- A further 3 assignments precedes all the loops
- Total $2n^3 + n^2 + n + 3$

[Go to Q 6](#)

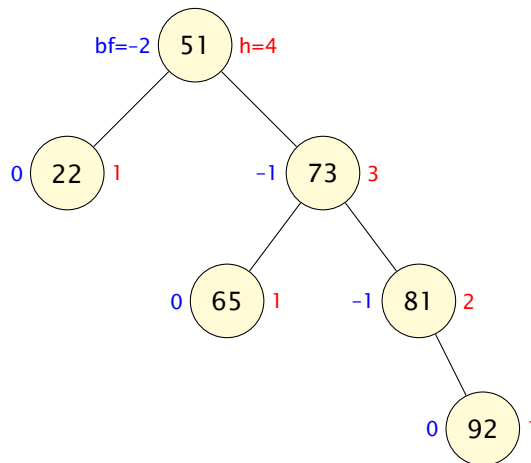
2.9 M269 2016J Exam Soln 7

- (a) **A** and **D** are true
 - **B** is not true — chaining is a way of resolving collisions
 - **C** is not true — see [What is primary and secondary clustering in hash?, Primary clustering](#)
- (b) The load factor is $0.6 = \frac{6}{10}$

[Go to Q 7](#)

2.10 M269 2016J Exam Soln 8

- (a)



(b) The tree is not balanced since node 51 has balance factor -2 which is outside $-1, 0, 1$

- Note the height definition here is from my notes not M269

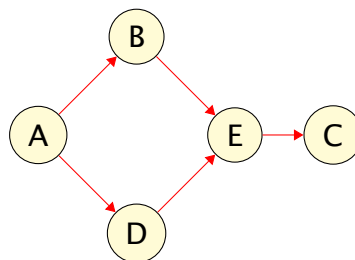
[Go to Q 8](#)

2.11 M269 2016J Exam Soln 9

- (a) For a typical food web, the graph is **directed** because the relation is not symmetric: if A eats B, B doesn't necessarily eat A.
- (b) An adjacency matrix is not a good data structure because it would waste memory: only few of the n^2 matrix cells would be non-zero

[Go to Q 9](#)

2.12 M269 2016J Exam Soln 10



- ABDEC, ADBEC are topological sorts
- ABCDE, ABEDC are not topological sorts
- The graph must be shown with directed edges (arrows)

[Go to Q 10](#)

2.13 M269 2016J Exam Soln 11

- (a) A WFF is *satisfiable* if there is at least one interpretation under which the formula is true — hence a tautology is satisfiable

- (b) The WFF is not a tautology because the formula is not true under all interpretations — it is false when P is true and q is false

P	Q	$(P \vee Q)$	$(P \vee Q) \rightarrow Q$
T	T	T	T
T	F	T	F
F	T	T	T
F	F	F	T

[Go to Q 11](#)

2.14 M269 2016J Exam Soln 12

- (a) **B.** *All dogs are Nicky's pets* translates to:

- $\forall X.(\text{dog}(X) \rightarrow \text{has-pet}(\text{nicky}, X))$
- **A.** $\forall X.(\text{dog}(X) \wedge \text{has-pet}(\text{nicky}, X))$ means
- *All objects are dogs and are Nicky's pets*
- **C.** $\exists X.(\text{dog}(X) \wedge \text{has-pet}(\text{nicky}, X))$ means
- *There is some object which is a dog and is Nicky's pet*

- (b) $\forall X.\exists Y.(\text{dog}(X) \rightarrow \text{feeds}(Y, X))$ means

All dogs are fed by someone

- But not *Somebody feeds all dogs* which would be
- $\exists Y.\forall X.(\text{dog}(X) \rightarrow \text{feeds}(Y, X))$
- This is true because

- (i) If X is not a dog then the implication is true

- (ii) We have $\mathcal{I}(\text{dog}) = \{\text{Rex}, \text{Fifo}\}$ and we have $(\text{Clara}, \text{Rex}) \in \mathcal{I}(\text{feeds})$ and $(\text{Nicky}, \text{Fifo}) \in \mathcal{I}(\text{feeds})$

[Go to Q 12](#)

2.15 M269 2016J Exam Soln 13

	<i>make</i>	<i>model</i>
(a)	Mowlt	Speedy
	Blades	Yard

- *Which models of which makes are electric lawnmowers ?*

- (b)

```
SELECT manufacturer, model
FROM lawnmower CROSS JOIN brand
WHERE make = manufacturer
AND location = 'China';
```

- Also allow

FROM lawnmower, brand

[Go to Q 13](#)

2.16 M269 2016J Exam Soln 14

- A. ✓ If a decision problem is in NP, then it is computable.
- B. The complexity of an algorithm that solves a problem places a lower bound on the complexity of the problem itself.
- C. If the best algorithm we currently have for solving a decision problem has complexity $O(2^n)$, then we know that problem can't be in P.
- D. ✓ If an NP-hard problem A can be Karp-reduced to a problem B, then problem B is NP-hard too.
- E. Every NP-hard problem is also NP-complete.

[Go to Q 14](#)

2.17 M269 2016J Exam Soln 15

- (a) Undecidable: 3. Totality Problem
- (b) Tractable: 2. Sorted?, 4. Path?
- (c) NP-complete: 1. 3SAT Problem

[Go to Q 15](#)

2.18 M269 2016J Exam Soln Part2

- Part 2 solutions

[Go to Q Part2](#)

2.19 M269 2016J Exam Soln 16

- (a) It is a worst-case scenario since there is no early exit from the loop, before returning false.
 - The complexity is $O(n + e)$ since all nodes are visited by the outer loop, and all edges are visited by the linear search through the neighbours of each node.
 - Note that the number of edges, e , could vary from 0 for completely unconnected to $n(n - 1)/2$ in a [Complete graph](#) where every node is connected to every other node
- (b) (i) **Name:** isolatedNodes
 - **Inputs:** an undirected graph *theGraph* (or a Ugraph *theGraph*)

- **Outputs:** *isolated*, a set of nodes
- **Preconditions:** true
- **Postconditions:** all nodes without neighbours in *theGraph* are in *isolated*; each node in *isolated* has no neighbours in *theGraph*

Alternative: a node is in *isolated* if and only if it has no neighbours in *theGraph*

(b) (ii) It would be an inspector because *theGraph* is not in the outputs.

Alternative: because the operation does not create or modify a graph.

- (iii) Initialise *isolated* to the empty set.

Iterate over the nodes of *theGraph* and for each one check if its neighbours is the empty sequence.

If so, add the node to *isolated*.

(c) The data structure is a weighted tree (alternative: acyclic graph).

Prim → Minimum Spanning Tree

The nodes represent the data centres.

The edges represent the cables.

The weights represent the cable lengths.

- To compute the longest path, do any traversal of the tree starting at the gateway node and add the weights of the edges visited.

For an efficient, single-pass algorithm, when visiting a leaf, check if its distance is the maximum so far.

- *Alternative:* calculate the height of the tree with cable lengths

[Go to Q 16](#)

2.20 M269 2016J Exam Soln 17

- There is no definitive answer — here are some points:

1. *Setting the scene with the importance of relational databases:*

- All retailers need to keep data on their products, suppliers and clients, the properties of those entities (e.g. current stock of a product) and their relationships (e.g. who bought which product to issue invoices).
- Storing entities and their properties and relationships is such a generic need across business, government departments and other organisations that so-called relational databases were invented for that purpose.

2. *What are relational databases:*

- It is a data structure that represents each entity type as a table, with one column per property and one row per entity, e.g. a table to represent customers may have columns for their name and address.

- A table can also represent a relation, e.g. a table with customer names and product ids would store who bought what.
- A database can be queried to retrieve information from the database, e.g. which other customers bought a particular book

3. *What is predicate logic and its relation to relational databases:*

- Predicate logic is a formal language to represent unambiguously statements about entities and their properties and relations, e.g. *No customer in Yorkshire bought a polka dot dress.*
- Given information about the existing entities and their properties/relations, it is possible to prove whether a predicate logic statement is true or false.
- A database query is a particular form of a predicate logic statement.
- Running a query is an automated proof: it returns the entities stored in the database that make the statement true; if no entities are returned, the statement is false.

4. *Conclusion:*

- Formal logic helps verifying the correctness of systems, which is important for our daily reliance on them.
- There are limits on what is computable, and a system may be correct but not fit for purpose, so formal logic doesn't suffice for quality assurance.

[Go to Q 17](#)