# M269 Exams

## Prsntn 2014J Exam

## Contents

# 1 M269 Prsntn 2014J Exam Qs

## 1.1 M269 2014J Exam Qs

- M269 Algorithms, Data Structures and Computability
- Presentation 2014J Exam
- Date Monday, 8 June 2015 Time 10:00–13:00
- There are **TWO** parts to this examination. You should attempt all questions in **both** parts
- **Part 1** carries **60 marks** — **100 minutes**
- **Part 2** carries **40 marks** — **70 minutes**
- **Note** see the original exam paper for exact wording and formatting — these slides and notes may change some wording and formatting

Go to Exam Soln s

## 1.2 M269 2014J Exam Q Part1

- Answer every question in this part.
- 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.

Go to Exam Soln Part1

## 1.3 M269 2014J Exam Q 1

- **Question 1** Which **two** of the following statements are true ? **(2 marks)**

**A.** A computational thinker can explain why some computational problems do not have a good computational solution.

**B.** The main factor when deciding the best algorithm for a given task is the programming language in which the algorithm is to be implemented.

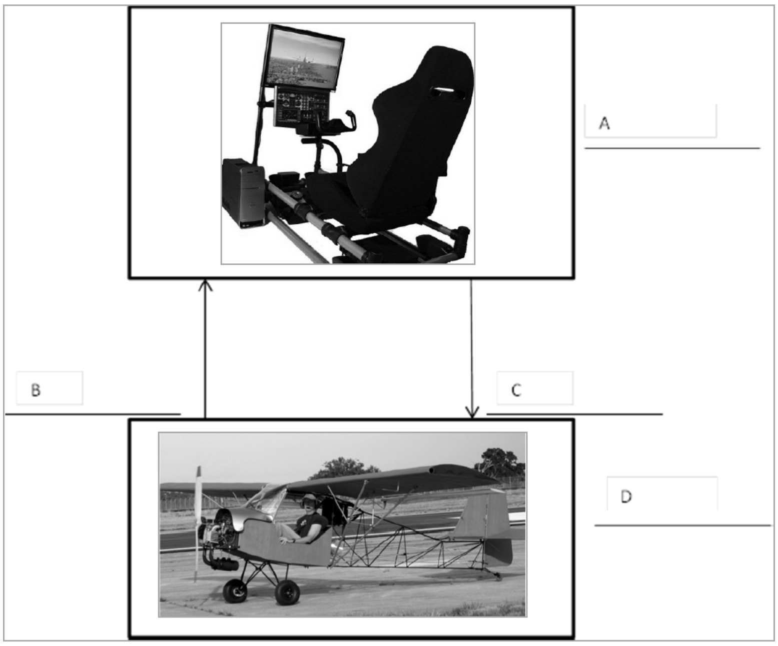
**C.** A decision problem can be defined as a formally stated problem to which the answer is either yes or no.

**D.** The data structure chosen to hold data does not affect the efficiency of an algorithm operating on that data.

Go to Exam Soln 1

## 1.4 M269 2014J Exam Q 2

- **Question 2** The diagram below shows images of a flight simulator for a single seater aircraft and an actual single seater aircraft. **(2 marks)**




- Complete the diagram above by adding an appropriate label (one of the numbers 1 to 8) in each of the spaces indicated by A, B, C and D. The possible answers are shown as 1 to 8 below.

1 ... represented by

2 ... ignores detail of

3 ... solves

4 ... transforms

5 ... model

6 ... part of reality

7 ... computational problem

8 ... layer

Go to Exam Soln 2

## 1.5 M269 2014J Exam Q 3

- **Question 3** An insertion sort is being carried out on the list of integers shown below, so as to arrange the list in ascending numerical order: **(4 marks)**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| start array | 54 | 26 | 93 | 17 | 77 | 44 | 55 | 31 |

- For the first four passes of the algorithm (after assuming that a list with one item is already sorted), show the order of the items in the list after that pass:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| (Pass 1) | | | | | | | | |
| (Pass 2) | | | | | | | | |
| (Pass 3) | | | | | | | | |
| (Pass 4) | | | | | | | | |

Go to Exam Soln 3

## 1.6 M269 2014J Exam Q 4

- **Question 4** Consider the guard in the following Python `while` loop header: **(4 marks)**

```
while (b > 8) and not(a < 6 or b <= 8):
```

(a) Make the following substitutions:

P represents `a < 6`

Q represents `b > 8`

complete the following truth table:

| P | Q | ¬Q | P ∨ ¬Q | ¬(P ∨ ¬Q) | Q ∧ ¬(P ∨ ¬Q) |
|---|---|---|---|---|---|
| T | T | | | | |
| T | F | | | | |
| F | T | | | | |
| F | F | | | | |

**(b)** Use the results from your truth table to choose which one of the following expressions could be used as a simpler equivalent to the above guard.

A. `(b > 9 and a < 6)`

B. `not (a < 6 or b > 8)`

C. `(a >= 6 and b <= 8)`

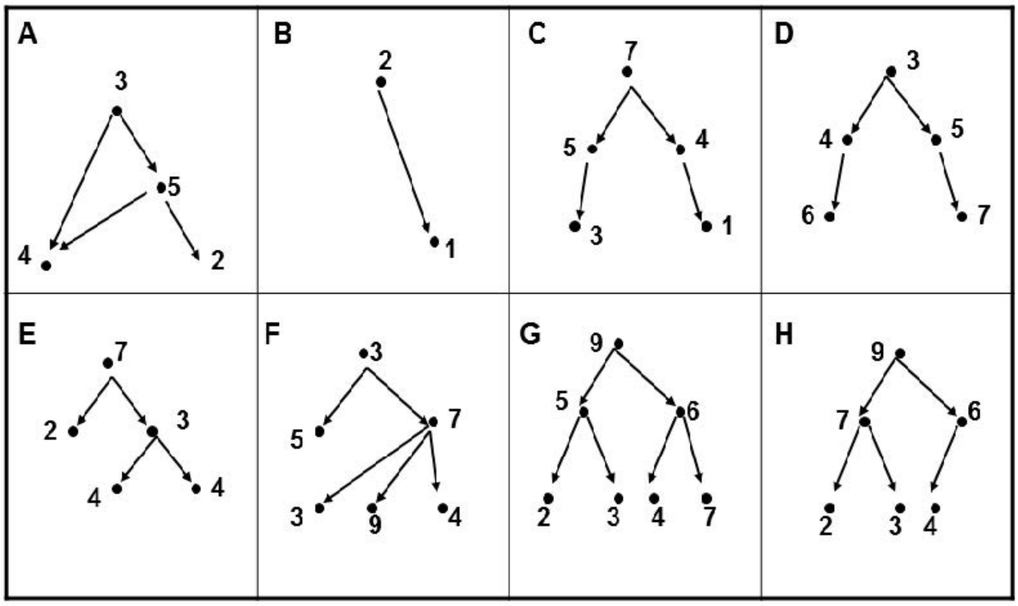
D. `not (a < 6 or b <= 8)`

E. `(a < 6 and b <= 8)`

Go to Exam Soln 4

## 1.7 M269 2014J Exam Q 5

- **Question 5** Consider the diagrams in A–H. In these diagrams, nodes are represented by black dots and edges by arrows. The numbers are the keys for the corresponding nodes. **(4 marks)**




- Answer the following questions. Write your answer on the line that follows each question. In each case there is at least one diagram in the answer but there may be more than one. Explanations are not required.

(a) Which of A, B, C and D **do not** show trees ?

(b) Which of E, F, G and H are binary trees ?

(c) Which of C, D, G and H are complete binary trees ?

(d) Which of C, D, G and H are binary heaps ?

Go to Exam Soln 5

## 1.8 M269 2014J Exam Q 6

- **Question 6** Consider the following function, which takes an integer n as an argument. You can assume that n is positive. **(4 marks)**

```python
def twiddle(n):
    a = 5
    for i in range(n):
        for j in range(n):
            x = i * i
            y = j * j
        for k in range(n):
            z=i*k
    ans = (x + y + z) * a
    return ans
```

- From the five options below, select the **one** that represents 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. $T(n) = n^2 + 3n + 2$ and $O(n^2)$

B. $T(n) = 2n^3 + n^2 + 2$ and $O(n^3)$

C. $T(n) = 2n^2 + n + 2$ and $O(2n^2)$

D. $T(n) = 3n^2 + 2$ and $O(n^2)$

E. T(n) = 2n + 5 and O(n)

- Now explain how you obtained T(n) and the Big-O complexity.

Go to Exam Soln 6

## 1.9 M269 2014J Exam Q 7

**(a)** Given an alphabet of ACGT and the target string CAGAGAG, select the option below that represents the shift table that would be used by the Sunday string search algorithm.

A.

| A | C | G | T |
|---|---|---|---|
| 1 | 6 | 0 | 8 |

B.

| A | C | G | T |
|---|---|---|---|
| 2 | 1 | 3 | 0 |

C.

| A | C | G | T |
|---|---|---|---|
| 1 | 0 | 2 | 7 |

A.

| A | C | G | T |
|---|---|---|---|
| 2 | 7 | 1 | 8 |

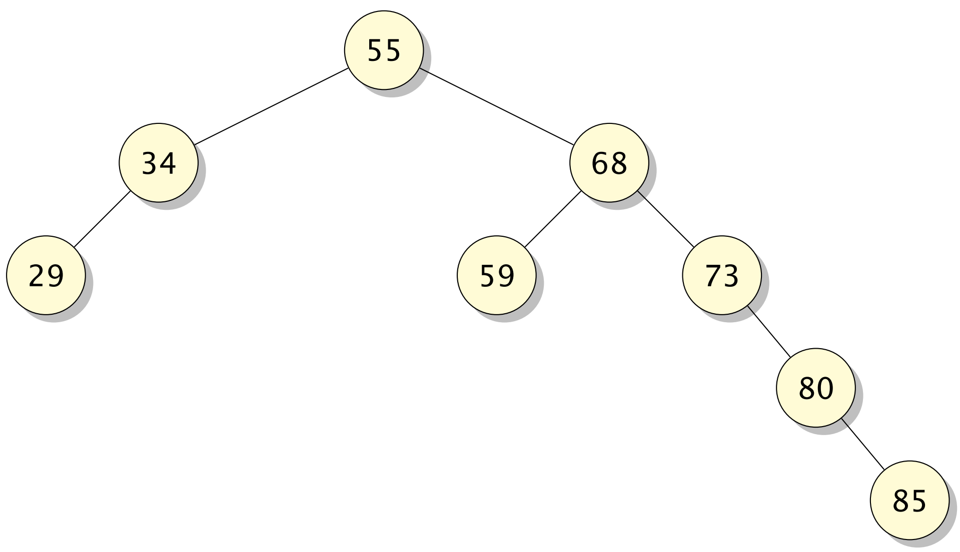
**(b)** Assuming a hash table with 20 slots, using the folding method, with 2 as the size of each part, what would be the hash value of the item 1459862913 ?

- In the box below give the hash value and indicate how you calculated your answer, showing all steps.

Go to Exam Soln 7

## 1.10 M269 2014J Exam Q 8

- Consider the following Binary Search Tree. **(4 marks)**



**(a)** Calculate the balance factors of each node in the above tree and modify the diagram to show these balance factors.

**(b)** Given your calculated balance factors, would this tree need to be rebalanced to be a valid AVL tree? Give your answer and a brief explanation for your answer in the box provided below.

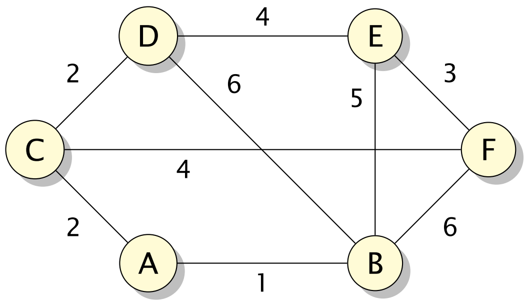
Go to Exam Soln 8

## 1.11 M269 2014J Exam Q 9

- **Question 9** Recall that the structured English for Dijkstra's algorithm is: **(5 marks)**

```
create priority~queue
set dist to 0 for v and dist to infinity
  for all other vertices
add all vertices to priority~queue
ITERATE while priority~queue is not empty
  remove u from the front of the queue
  ITERATE over w in the neighbours of u
    set new~distance to
      dist u + length of edge from u to w
    IF new~distance is less than dist w
      set dist w to new~distance
      change priority(w, new~distance)
```

- Now consider the following weighted graph:




- **Starting from vertex A**, the following table represents the distances from each vertex to A after the second line of structured English is executed for the graph given above (using the convention that the character ∞ represents infinity):

| Vertex | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| Distance | 0 | ∞ | ∞ | ∞ | ∞ | ∞ |

- *Note that neither the table above nor the subsequent tables represent the priority queue.*

- Now, complete the appropriate boxes in the next table to show the distances after the first and second iterations of the `while` loop of the algorithm.

| Vertex | A | B | C | D | E | F |
|---|---|---|---|---|---|---|

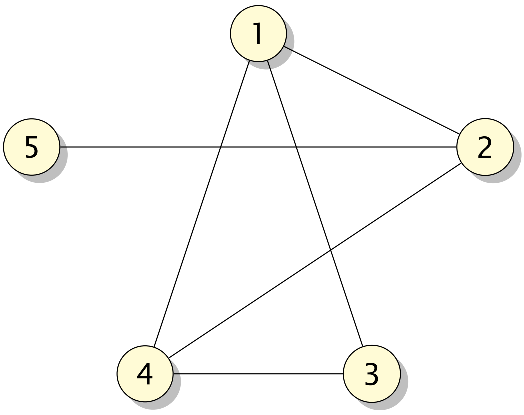
| Distance | 0 | | | | | | First iteration |
|---|---|---|---|---|---|---|---|

| Distance | 0 | | | | | | Second iteration |
|---|---|---|---|---|---|---|---|

Go to Exam Soln 9

## 1.12 M269 2014J Exam Q 10

- **Question 10** Consider the following graph: **(4 marks)**




- From the options below, select the **two** which show possible orders in which the vertices of the above graph could be visited in a *Breadth First Search* (BFS) starting at vertex 2:

A.

| Vertex | 2 | 3 | 4 | 5 | 1 |
|---|---|---|---|---|---|

B.

| Vertex | 2 | 1 | 4 | 5 | 3 |
|---|---|---|---|---|---|

C.

| Vertex | 2 | 1 | 3 | 4 | 5 |
|---|---|---|---|---|---|

D.

| Vertex | 2 | 5 | 1 | 4 | 3 |
|---|---|---|---|---|---|

E.

| Vertex | 2 | 5 | 4 | 3 | 1 |
|---|---|---|---|---|---|

F.

| Vertex | 2 | 1 | 3 | 4 | 5 |
|---|---|---|---|---|---|

Go to Exam Soln 10

## 1.13 M269 2014J Exam Q 11

- **Question 11** **(4 marks)**

**(a)** What does it mean to say that a well-formed formula (WFF) is a *contradiction* ? Use the space below for your answer.

(b) Is the following WFF a contradiction ?

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

- Explain how you arrived at your answer in the space below:

Go to Exam Soln 11

## 1.14 M269 2014J Exam Q 12

- **Question 12** A particular interpretation of predicate logic allows facts to be expressed about cities and people, in particular, facts about who visited and/or liked which cities. In this interpretation, we will make use of the following two sets:
**(6 marks)**

cities = {Adelaide, San Francisco, Mumbai}

persons = {Lin, Derren, Gabi}

- Some of the assignments in the interpretation are given below (where the symbol $\mathcal{I}$ is used to show assignment). The interpretation assigns Lin, Derren and Gabi to the constants *lin*, *derren* and *gabi*.
- $\mathcal{I}(lin)$ = Lin  $\mathcal{I}(derren)$ = Derren  $\mathcal{I}(gabi)$ = Gabi
- The predicates *has_visited* and likes are assigned to binary relations. The comprehensions of the relations are:

$\mathcal{I}(has\_visited)$ = {(A, B) : the person A has visited the city B}

$\mathcal{I}(likes)$ = {(A, B) : the person A likes the city B}

- The enumerations of the relations are:

$\mathcal{I}(has\_visited)$ = {(Lin, Adelaide), (Derren, San Francisco), (Gabi, Mumbai), (Gabi, San Francisco)}

$\mathcal{I}(likes)$ = {(Lin, Mumbai), (Lin, San Francisco), (Derren, Mumbai), (Derren, San Francisco), (Gabi, Adelaide)}

- You will find parts (a) and (b) of this question on the next page, whilst the interpretation is reproduced on the other side of this page.
- In both parts, you are given a sentence of predicate logic and asked to provide an English translation of the sentence in the box immediately following it.
- You also need to state whether the sentence is TRUE or FALSE in the interpretation that is provided on this page, and give an explanation of your answer.
- In your explanation you need to consider any relevant values for the variables, and *show, using the interpretation above, whether they make the quantified expression TRUE or FALSE*.
- When your explanation refers to the interpretation, make sure that you use formal notation.

- So instead of saying that *Lin likes Mumbai according to the interpretation*, write:

  (Lin, Mumbai) $\in \mathcal{I}(likes)$.

- Similarly, instead of *Lin doesn't like Mumbai* you would need to write:

  (Lin, Mumbai) $\notin \mathcal{I}(likes)$.

- **Interpretation to be used for answering Question 12**
- cities = {Adelaide, San Francisco, Mumbai}
- persons = {Lin, Derren, Gabi}
- $\mathcal{I}(lin)$ = Lin $\mathcal{I}(derren)$ = Derren $\mathcal{I}(gabi)$ = Gabi
- $\mathcal{I}(has\_visited)$ = {(Lin, Adelaide), (Derren, San Francisco), (Gabi, Mumbai), (Gabi, San Francisco)}
- $\mathcal{I}(likes)$ = {(Lin, Mumbai), (Lin, San Francisco), (Derren, Mumbai), (Derren, San Francisco), (Gabi, Adelaide)}

**(a)** $\exists X.\neg(likes(lin, X) \wedge likes(derren, X))$

can be translated into English as:

- This sentence is ____ (choose from TRUE/FALSE), because:

**(b)** $\forall X.(likes(gabi, X) \vee has\_visited(gabi, X))$

can be translated into English as:

- This sentence is ____ (choose from TRUE/FALSE), because:

Go to Exam Soln 12

## 1.15 M269 2014J Exam Q 13

- **Question 13** A database contains the following tables, *production_line* and *product*. **(6 marks)**

| *production_line* | |
|---|---|
| *id* | *unit* |
| Line1 | Sportscar |
| Line2 | SUV |
| Line3 | Bus |
| Line4 | Tractor |
| Line5 | Aeroplane |

| *product* | |
|---|---|
| *type* | *unit_price* |
| SUV | 50000 |
| Sportscar | 200000 |
| Bus | 250000 |
| Tractor | 50000 |
| Aeroplane | 30000000 |

- For the following SQL query, give the table returned by the query.

```sql
SELECT id, unit_price
FROM production_line CROSS JOIN product
WHERE unit = type AND unit_price < 300000;
```

| |
|---|
| |

- Write an SQL query which answers the question *Which products cost exactly 50000?* The answer should be the following table:

| *type* |
|---|
| SUV |
| Tractor |

| |
|---|
| |

Go to Exam Soln 13

## 1.16 M269 2014J Exam Q 14

- **Question 14** Consider the following axiom schema and rules: **(4 marks)**

| *Axiom schema* | $\{\mathbf{A}\} \vdash \mathbf{A}$ |
|---|---|
| *Rules* | $\dfrac{\Gamma \vdash \mathbf{A} \wedge \mathbf{B}}{\Gamma \vdash \mathbf{A}}$ ($\wedge$-elimination left) |
| | $\dfrac{\Gamma \vdash \mathbf{A} \wedge \mathbf{B}}{\Gamma \vdash \mathbf{B}}$ ($\wedge$-elimination right) |
| | $\dfrac{\Gamma \vdash \mathbf{A} \quad \Gamma \vdash \mathbf{B}}{\Gamma \vdash \mathbf{A} \wedge \mathbf{B}}$ ($\wedge$-introduction) |
| | $\dfrac{\Gamma \cup \{\mathbf{A}\} \vdash \mathbf{B}}{\Gamma \vdash \mathbf{A} \rightarrow \mathbf{B}}$ ($\rightarrow$-introduction) |
| | $\dfrac{\Gamma \vdash \mathbf{A} \quad \Gamma \vdash \mathbf{A} \rightarrow \mathbf{B}}{\Gamma \vdash \mathbf{B}}$ ($\rightarrow$-elimination) |

- Complete the following proof by filling in the two boxes. You can use any of the above as appropriate.

| | | |
|---|---|---|
| 1. | $\{(V \wedge W)\} \vdash (V \wedge W)$ | [Axiom schema] |
| 2. | ?? ?? | [1 $\wedge$-elimination right] |
| 3. | $\emptyset \vdash (V \wedge W) \rightarrow W$ | ?? ?? |

Go to Exam Soln 14

## 1.17 M269 2014J Exam Q 15

**(a)** Complete the following sentence in the box below:

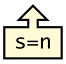
The statement *If a computational problem is in NP, then it must be intractable* may be false because

| |
|---|
| |

**(b)** Consider the following Turing Machine:

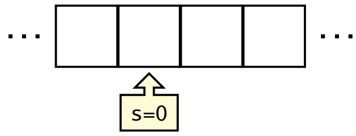
| | $ | EMPTY |
|---|---|---|
| 0 | | WRITE $<br>MOVE RIGHT<br>NEXT STATE 1 |
| 1 | WRITE blank<br>MOVE LEFT<br>NEXT STATE 0 | WRITE $<br>MOVE RIGHT<br>NEXT STATE 0 |

- Assume that the starting state is 0 and that the input tape consists of empty squares.
- One square is marked as the current square with the tape head, shown here as

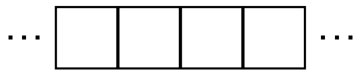


where n is the current state

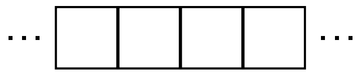
- So, initially the tape looks as follows:

Initial tape:



- Note it the original exam the tape cells and head are denoted by ASCII symbols
- Using the same notation, write down what the tape looks like after each of the next two steps of the computation.
- Use the boxes below for this.
- After the first step

First step:


- After the second step

Second step:


Go to Exam Soln 15

# 2 M269 Prsntn 2014J Exam Solns

## 2.1 M269 2014J 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 Q s

## 2.2 M269 2014J Exam Soln Part1

- Part 1 solutions

Go to Exam Q Part1

## 2.3 M269 2014J Exam Soln 1

- A, C

Go to Exam Q 1

## 2.4 M269 2014J Exam Soln 2

A 5 model

B 1 represented by

C 2 ignores detail of

D 6 part of reality

- See Unit 1 section 3.2 *Computational thinking and abstraction*

Go to Exam Q 2

## 2.5 M269 2014J Exam Soln 3

- The sorted part of the list is filled in pale blue with the next item to be inserted in pale red

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| (Pass 1) | 26 | 54 | 93 | 17 | 77 | 44 | 55 | 31 |
| (Pass 2) | 26 | 54 | 93 | 17 | 77 | 44 | 55 | 31 |
| (Pass 3) | 17 | 26 | 54 | 93 | 77 | 44 | 55 | 31 |
| (Pass 4) | 17 | 26 | 54 | 77 | 93 | 44 | 55 | 31 |

Go to Exam Q 3

## 2.6 M269 2014J Exam Soln 4

**(a)** Truth table

| P | Q | ¬Q | P ∨ ¬Q | ¬(P ∨ ¬Q) | Q ∧ ¬(P ∨ ¬Q) |
|---|---|---|---|---|---|
| T | T | F | T | F | F |
| T | F | T | T | F | F |
| F | T | F | F | T | T |
| F | F | T | T | F | F |

**(b)** The only row that has True in the final column for the guard is

P = F and Q = T

- This is equivalent to

  $\neg P \wedge Q$

- $\rightarrow \neg(P \vee \neg Q)$ hence answer D

Go to Exam Q 4

## 2.7 M269 2014J Exam Soln 5

- TODO: Q5 Solution

Go to Exam Q 5

## 2.8 M269 2014J Exam Soln 6

- TODO: Q6 Solution

Go to Exam Q 6

## 2.9 M269 2014J Exam Soln 7

**(a)** D TODO: Q7 (a) solution explanation

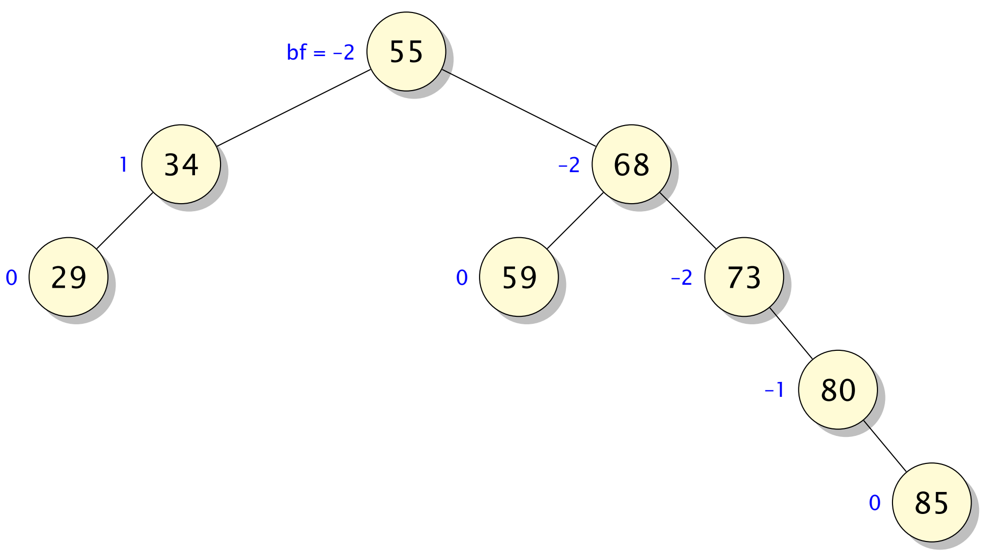
**(b)** Hash value = 1

since (14 + 59 + 86 + 29 + 13) % 20 = 1

Go to Exam Q 7

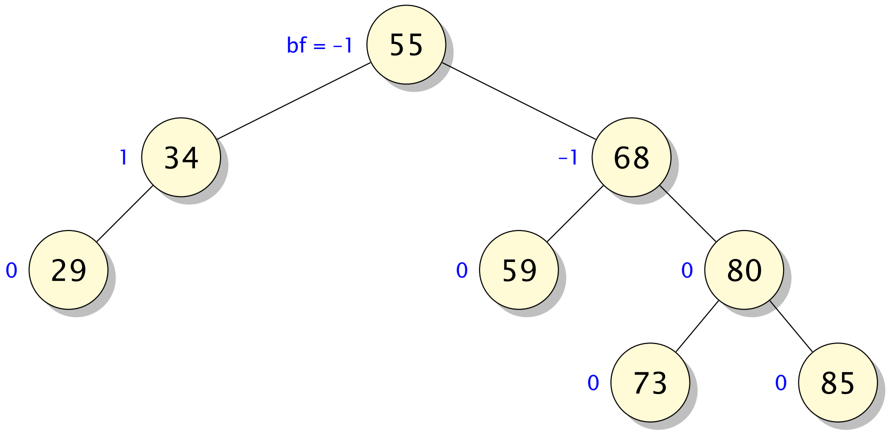
## 2.10 M269 2014J Exam Soln 8

**(a)** Tree with balance factors



**(b)** The tree is not a valid AVL tree since some of the balance factors are outside the range $[-1, 0, +1]$

- The tree must have become unbalanced as a result of inserting 85
- Rebalancing would mean a left rotation around the node 73 as below (this diagram was not required in the exam)




Go to Exam Q 8

## 2.11 M269 2014J Exam Soln 9

- The completed table

| Vertex | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|
| Distance | 0 | 1 | 2 | ∞ | ∞ | ∞ | First iteration |
| Distance | 0 | 1 | 2 | 7 | 6 | 7 | Second iteration |

- The node being processed has a red background
- Nodes with a final label have a green background
- Neighbours of the node being processed have blue text

Go to Exam Q 9

- The complete iterations

| Vertex | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|
| Distance | 0 | 1 | 2 | ∞ | ∞ | ∞ | First iteration |
| Distance | 0 | 1 | 2 | 7 | 6 | 7 | Second iteration |
| Distance | 0 | 1 | 2 | 4 | 6 | 6 | Third iteration |
| Distance | 0 | 1 | 2 | 4 | 6 | 6 | Fourth iteration |
| Distance | 0 | 1 | 2 | 4 | 6 | 6 | Fifth iteration |
| Distance | 0 | 1 | 2 | 4 | 6 | 6 | Sixth iteration |

Go to Exam Q 9

## 2.12 M269 2014J Exam Soln 10

- B, D
- A *Breadth First Search* (BFS) from 2 must visit its neighbours in some order first

Go to Exam Q 10

## 2.13 M269 2014J Exam Soln 11

- TODO: Solution 11

Go to Exam Q 11

## 2.14 M269 2014J Exam Soln 12

- TODO: Solution 12

Go to Exam Q 12

## 2.15 M269 2014J Exam Soln 13

- TODO: Solution 13

Go to Exam Q 13

## 2.16 M269 2014J Exam Soln 14

- Completed proof

1. $\{(V \wedge W)\} \vdash (V \wedge W)$    [Axiom schema]
2. $\boxed{\{(V \wedge W)\} \vdash W}$    [1 $\wedge$-elimination right]
3. $\emptyset \vdash (V \wedge W) \rightarrow W$    $\boxed{2 \ \rightarrow\text{-introduction}}$

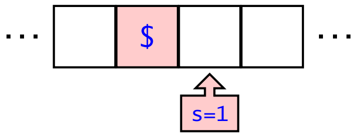
- Note $\emptyset$ is a symbol for the empty set (in LaTeX \varnothing)
- You could also use $\{\}$ (or even leave a blank, but that would not be good practice)

Go to Exam Q 14

## 2.17 M269 2014J Exam Soln 15

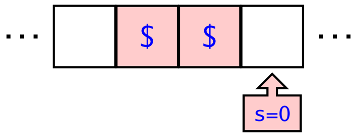
- After the first step

First step:




- After the second step

Second step:




Go to Exam Q 15