VICTORIAN CURRICULUM AND ASSESSMENT AUTHORITY

2015 VCE Algorithmics (HESS) examination report

General comments

In the 2015 VCE Algorithmics (HESS) examination, the first examination for the study, students achieved scores across the range of available marks and were able to demonstrate their knowledge. In general, students performed better in Section A than in Section B.

In Section A, most students answered Questions 2, 3, 13, 18 and 20 correctly.

In Section B, some students demonstrated a good understanding of the key knowledge and skills. However, often students did not address all aspects of a question. In extended-response questions, such as Questions 5, 9, 11 and 12, it may have been helpful for students to structure their answer in order to manage the various components of their answers.

Specific information

Note: Student responses reproduced in this report have not been corrected for grammar, spelling or factual information.

This report provides sample answers or an indication of what answers may have included. Unless otherwise stated, these are not intended to be exemplary or complete responses.

The statistics in this report may be subject to rounding resulting in a total more or less than 100 per cent.

Section A – Multiple-choice questions

The table below indicates the percentage of students who chose each option. The correct answer is indicated by shading.

Question	% A	% B	% C	% D	% No Answer	Comments
1	10	1	80	9	0	
2	0	0	100	0	0	
3	1	97	0	2	0	
4	7	15	1	77	0	
5	49	27	5	17	2	If the graph is already a tree, then it is already a minimal spanning tree.
6	22	23	49	6	1	
7	68	9	17	6	0	
8	3	61	32	2	1	
9	3	66	15	15	1	
10	68	9	8	15	0	
11	80	8	6	7	0	
12	14	7	75	5	0	
13	1	5	1	93	0	

Question	% A	% B	% C	% D	% No Answer	Comments
14	65	2	20	11	1	
15	11	10	75	2	1	
16	52	30	11	5	2	
17	2	13	1	84	0	
18	5	91	3	1	0	
19	1	78	11	9	0	
20	7	3	0	90	0	

Section B

Question 1

Marks	0	1	2	3	Average
%	8	27	54	12	1.7

Only those functions on the natural numbers that are computable by a Turing machine can possibly be manually computed by any device. A limitation of the Church-Turing thesis is that it is unproven and could be refuted if a device were created that had greater computational power than the Turing machine.

Most students were able to explain the meaning of the statement. Many students made reference to features absent from Turing machines but did not explicitly tie these into how the thesis might be refuted.

Question 2a.

Marks	0	1	2	Average
%	17	23	59	1.4

It was necessary to contrast a characteristic of the two abstract data types; providing non-comparative definitions was not sufficient.

The following is an example of a high-scoring response.

An array is an ADT that stores elements by an integer index. A dictionary or associative array stores elements by a 'key' element that can be used to retrieve items

Question 2b.

Marks	0	1	2	Average
%	3	27	70	1.7

Any example where there was an ordering that could be overridden was accepted. A common mistake was to compare one example of the use of a priority queue with one example of the use of a queue.

The following is an example of a high-scoring response.

A priority queue is more suitable when items have different priorities, such as important emails in an inbox or VIP access to special events.

Question 3

Marks	0	1	2	3	Average
%	57	24	15	4	0.7

This problem is decidable as the number of ways that an *n*-gon can be divided into *n*-2 triangles is computable, either by brute-force generation or by a dynamic programming approach that students may have encountered when seeing Catalan numbers.

Many students incorrectly discussed the tractability of the problem. Some confused intractability with undecidability or used the two terms interchangeably. Some students confused the decidability of this specific problem with the decidability of arbitrary statements.

Question 4a.

Marks	0	1	2	3	4	Average
%	11	16	34	22	17	2.2

A description of an algorithm (for example, breadth-first search, depth-first search) and how it relates to finding the monkey was required. Responses should have included an acknowledgment of the need for full exploration and an explanation of how the algorithm met the need to find the monkey quickly.

The following is an example of a high-scoring response.

> *Shara could use depth-first search. DFS will arbitrarily pick nodes to visit next, backtracking when no new node is reached. This makes it possible to find the monkey very quickly this way, although it could potentially take a long time due to its random selection of nodes from those available to visit. In the worst case, the algorithm will require O(|V| + |E|) running time before the position of the monkey is found (assuming it must check all nodes and edges).*

Question 4b.

Marks	0	1	2	3	Average
%	21	34	27	18	1.5

Students were expected to incorporate the debris into the data structure; for example, through a change in edge weights. This could lead to a prioritisation of options based on the least amount of debris. Discussion of the consequences of the modification was then expected.

The following is an example of a high-scoring response.

> *Each edge could be assigned a weight which corresponds with how difficult it is for the robot to move through the drain. The robot could then choose what untraversed drain to travel through by choosing which one at any given point in time is the least weight.*

Question 4c.

Marks	0	1	2	Average
%	8	26	66	1.6

Expanding the search area would increase the time required to find the monkey because the robot needs to search a greater number of buildings and tunnels.

A common misconception was that the time complexity would increase.

Question 5

Marks	0	1	2	3	4	Average
%	6	9	21	40	24	2.7

Most students were able to determine that Dijkstra's algorithm was problematic. Some reasons included its potential to produce incorrect results when negative edge weights were present, or the unsuitability of Dijkstra's algorithm in providing a maximal score, as it is a minimum distance algorithm.

The following is an example of a high-scoring response.

> *Dijkstra's computes the lowest value for the weight of a single source shortest path. In this situation the maximum value is desired which is the opposite of what Dijkstra's will return.*

Question 6a.

Marks	0	1	2	3	Average
%	5	49	14	32	1.8

A dictionary would be a more suitable data structure. Retrieving an element in a dictionary takes O(1) time, while retrieving a specific item in a list will take O(n) time.

High-scoring responses discussed the Big-O complexity of finding a specific element in dictionaries in comparison to lists.

Question 6b.

Marks	0	1	2	Average
%	30	36	34	1.1

A sorted array or list (kept sorted by name) was the preferred choice as a dictionary does not support listing in order. Browsing a list of names in alphabetical order requires an ADT that supports sorting.

Most students struggled to gain full marks due to inaccurate descriptions of the ADT under discussion. Some students stated a list or array without specifying that it would be sorted.

Question 7a.

Marks	0	1	2	3	4	Average
%	8	3	39	11	39	2.7

The four boundary conditions were 0, 1, 2^{20} and $2^{20} + 1$.

Common incorrect responses included 2, $2^{20} - 1$ and 2^{21}.

Question 7b.

Marks	0	1	2	Average
%	53	16	31	0.8

$|E| >= |V| - 1$ a tree (smallest number of edges for a connected graph)

$|E| = \frac{|V|(|V| + 1)}{2}$ or $\frac{|V|(|V| - 1)}{2}$ all connected (largest number of edges)

A common incorrect response was to provide answers not expressed in terms of |V|.

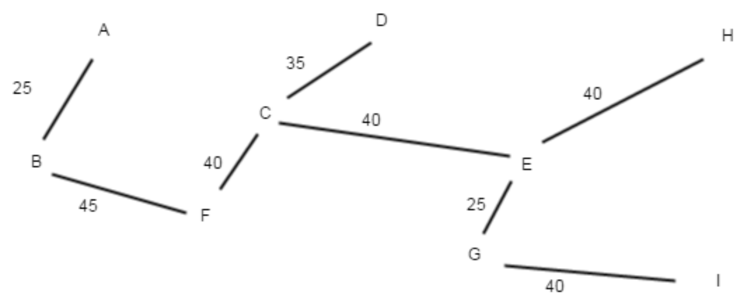
Question 8a.

Marks	0	1	Average
%	6	94	1

Prim's algorithm was the expected answer; however, Kruskal's algorithm was also accepted.

Question 8b.

Marks	0	1	2	3	4	Average
%	1	0	1	3	95	3.9



A number of different minimum spanning trees could be drawn.

Question 9

Marks	0	1	2	3	4	Average
%	3	9	21	36	30	2.8

The following is an example of a high-scoring response.

Turing machines have:

An infinite one-dimensional tape divided into cells, where each cell contains a specific symbol from a finite alphabet.
A head, which reads the cell at a position on the tape, and moves left or right after writing to that cell.
A state register, which stores the current state the Turing machine is in, from a finite set of states including the halting state.
A finite set of instructions, which determine the action of the machine.

Some students provided characteristics that were too similar. For example, a head that can read from the tape and a head that can write to the tape.

Question 10ai.

Marks	0	1	2	3	Average
%	21	22	8	50	1.9

START	1	1	1	1	1
1	X	1	2	X	1
1	1	2	4	4	5
1	2	4	X	4	X
1	3	7	7	11	11
1	4	X	7	18	END 29

A common misconception was to work down and across the grid, adding 1 each time.

Question 10aii.

Marks	0	1	Average
%	47	53	0.5

29 paths

Question 10b.

Marks	0	1	2	3	4	5	6	7	8	9	Average
%	13	17	12	12	9	9	7	8	9	5	3.6

The following is an example of an iterative solution:

```
For each cell in the top row of the grid
    If cell value = X
            Set cell value to 0
        Else
            Set cell value to 1
For each cell in the first column of the grid
        If cell value = X
            Set cell value to 0
        Else
            Set cell value to 1
Set nrow to 2
While nrow <= n
        Set ncol to 2
        While ncol <= n
            If cell value in row nrow and column ncol = X
                Set cell value to 0
            Else
                Set cell value in row nrow and column ncol
                        to (cell value in row nrow1-1 and column ncol)
                         + (cell value in row nrow1 and column ncol-1)
            Set ncol to ncol +1
        Set now to nrow + 1
Return the value in row n and column n
```

A variety of algorithms that correctly calculated the number of different paths from the top left corner of the grid to the bottom right corner of the grid were accepted. Only a small number of students attempted a recursive solution.

Question 11

Marks	0	1	2	3	4	5	6	7	8	Average
%	9	15	26	19	10	8	11	3	0	2.9

Responses needed to discuss the connection between computer security and NP-Completeness, the application of DNA computing to NP-Complete problems, the reducibility of solutions to NP-Complete problems to other NP-Complete problems and any threat that DNA computing poses to computer security.

Student responses often noted the ability of DNA computing to utilise massive parallel processing and included some discussion of the connection between encryption and the NP class of problems. Discussion of the biochemistry of DNA computing was not necessary.

Question 12

Marks	0	1	2	3	4	Average
%	39	34	18	6	2	1

Applying backtracking to the travelling salesman problem requires that the problem be represented as a decision tree, choosing a root node and incrementally adding nodes not currently in the path to it until every node is in the path. The last node must connect back to the root. Backtracking is then used to test every combination of paths. When a solution is found, record its cost if it is cheaper than any other currently found solution. Whenever a decision creates a path with a cost that is greater than the best currently found solution, cease further exploration of that path.

This question was not answered well. Many students attempted to describe the concept of backtracking without referring back to the travelling salesman problem.

Question 13a.

Marks	0	1	2	3	4	5	6	Average
%	25	12	12	9	8	19	16	2.8

f([–1,0,4,5,10,11,15,20],**0**,**7**)

f([–1,0,4,5,10,11,15,20],**0**,**3**)

f([–1,0,4,5,10,11,15,20],**2**,**3**)

f([–1,0,4,5,10,11,15,20],**2**,**2**)

Some students incorrectly attempted to show the control flow of the algorithm within their diagram.

Question 13b.

Marks	0	1	Average
%	83	17	0.2

The algorithm returns i, such that A[i] = i if it exists, otherwise it returns –1.

A number of students incorrectly gave the answer to the specific problem in Question 13a.

Question 13c.

Marks	0	1	2	Average
%	69	23	7	0.4

$T(n) = T(n/2) + O(1)$, if $n > 1$
$T(1) = O(1)$ if $n = 1$

Many students did not provide the base case.

Question 14a.

Marks	0	1	Average
%	57	43	0.4

The graph has a negative cost cycle.

If the shortest path cost to a node decreases on the n-th iteration of the Bellman-Ford algorithm, then this will indicate the presence of a negative cost cycle. Some students indicated that the graph would have a negative edge, but this response was not sufficient.

Question 14b.

Marks	0	1	2	Average
%	51	22	28	0.8

The Bellman-Ford algorithm is unsuitable. As each subsequent iteration will continue to reduce the shortest path cost, there is no shortest path for these two nodes in G.