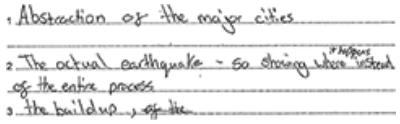
Mark scheme

Question			Answer/Indicative content	Marks	Guidance
1		i	1 mark per bullet to max 2 • Removing unnecessary detail • Simplifying a complex problem • Focussing on only the necessary/main parts	2	**Examiner's Comments** Many candidates could give a definition for abstraction with many specifying that it removed unnecessary detail to simplify a problem.
		ii	1 mark per bullet to max 3 e.g. • The physical game **board/squares** can be replaced with a grid • Physical **dice** can be replaced with two numbers • Physical player **counters** can be replaced with simple 2D graphics • Physical instruction **cards** can be replaced with text messages • Physical **timer** can be replaced with a simple text based timer	3	Allow other suitable examples that are relevant to the scenario. **Examiner's Comments** Some candidates struggled to use the context of the question to identify components such as the dice, counters, squares, cards or the timer. Candidates need to be able to apply knowledge of abstraction to give suitable responses within context.
		iii	1 mark per bullet to max 2 e.g. • Reduce programming time/cost • Simpler to solve the problem • Program requires less memory / computational power / faster program execution • Reduces complexity of programming code • Allows Eve to focus on core aspects	2	**Examiner's Comments** Many candidates successfully identified at least one reason why abstraction should be used, such as decreasing the computational processing demands for the solution. Some candidates did however repeated definitions of abstraction, rather than giving the advantages of using it. Another common error was to give an example of how it would benefit a player of the game rather than focusing on the design stage.
			Total	**7**	
2		i	1 mark for definition, 1 mark each for each example of use to max 2 (3 overall) **Definition:**	3	Allow any reasonable examples for this scenario For the example use, allow 2 marks

			• Removal of unnecessary detail **Example use:** • E.g. simplifying scenery • E.g. removing internal features of a vehicle that are not needed • E.g. simplify physics for vehicle movement • E.g. The vehicles may not be drawn to scale		for stating a valid example of abstraction with an expansion. For example,"simplify track (1) by taking out the bumps in the road (1)" would be given two marks. **Examiner's Comments** Many candidates scored two marks for defining abstraction and giving one example of its application to the game, but fewer managed to identify a second example for full marks.
		ii	1 mark each to max 2 • Simplifies the problem / algorithm / programming code • Faster to create the program code • Final program uses less memory/processor time • Programmer can focus on core aspects of the game • Completed game will be simpler for end users to understand / play	3	Do not accept a reiteration of a definition of abstraction. **Examiner's Comments** Some candidates reiterated a definition of abstraction and did not focus on the benefits of abstraction as the question required. Most candidates struggled to give more than one or two benefits. There were also a number of unqualified responses such as 'saves time' that did not specify 'saving coding development time'.
			Total	**6**	
3	a		1 mark each • Compare the first element (rainbow) to search item / clouds • If it is equal to the search item return index / found • If it is not equal move to the next element • Repeat until either search item / clouds is equal / or the end of the list has been reached	3	Allow answers by example from the given dataset **Examiner's Comments** Most candidates scored the majority of the marks available and demonstrated a clear understanding of a linear search. Many candidates answered by example with values from the given list.
	b		1 mark for: the data is not in order/sorted	1	**Examiner's Comments** Most candidates correctly identified the requirement for data to be sorted/ordered for a binary search to

					work. 'Organised' was too vague and was not accepted.
			Total	**4**	
4		i	1 mark for each description to max 2 and 1 mark for example e.g. • Removal of unnecessary detail… • ….to allow programmers to focus on core aspects of the problem…. • ….simplifies a complex problem Examples, e.g: • Treasure objects are replaced with text labels / no images of treasure are used • Island is set of coordinates and no info as to environment/layout and other objects	3	Allow other suitable examples that are relevant to the treasure game. **Examiner's Comments** Many candidates identified that abstraction simplified or removed unnecessary detail to gain some marks, but then found it harder to give an example relevant to the scenario. Many candidates gave examples that related to graphics whereas the scenario explicitly stated that the game was text based.
		ii	1 mark each to max 3 e.g. • Reduces programming time • Reduces complexity of code (through abstraction by generalisation) • Reduces amount of memory required / computational power • Simplifies the problem so it's easier to solve / understand (by recognising common patterns) • Allows programmers to focus on core aspects of the problem	3	**Examiner's Comments** Many candidates gained some marks, but most found it difficult to identify three distinct reasons, and repetition was often observed in responses. Again, some candidate answers did not provide suitable levels of qualification for the points given.
			Total	**6**	
5			1 mark per bullet to max 3 Any reasonable abstraction e.g. • will not be to scale not life size • will exclude features e.g. people, road markings etc	3	**Examiner's Comments** Most candidates demonstrated that they knew what abstraction was, but a significant number struggled to make clear qualified points. Some candidates gave definitions of abstraction as the removal of

			• will only show what is relevant e.g. buildings		unnecessary detail, but this did not answer the question. This was a contextual question that required relevant and qualif ied points. Responses such as 'the weather' were insufficient, whereas 'removal of the effects of weather' was a valid qualified point. Exemplar 2  This exemplar shows a response that has unqualified points that do not show how abstraction can be applied. It highlights factors that could be abstracted, but does not tell us how.
			Total	**3**	
6			1 mark for each component e.g. • Allocating cards to each player • Generating the deck • Managing whose turn it is to play • Checking won	3	Accept any reasonable component **Examiner's Comments** The specification requires candidates to be able to identify elements of computational thinking. As such, candidates are expected to be able to think procedurally and to be able to identify the components of a problem. While analysis of the problem given in context led most candidates to identify valid components, many struggled to read the scenario and to give relevant points. For instance, many reiterated aspects of checking if a move was valid, which was already given in the question.
			Total	**3**	
7		i	Max 1 mark for each definition e.g. **Term** | **Definition**	2	**Examiner's Comments** While many candidates accurately recited definitions for abstraction and decomposition there were equally many that presented very vague and unqualified responses. E.g. For abstraction: 'Simplifying a problem'

			<table><tr><td>Abstraction</td><td>Removal of unnecessary components / focus on only necessary components</td></tr><tr><td>Decomposition</td><td>Breaking down a problem into subproblems</td></tr></table>		without specifying how it was simplified was insufficient. For decomposition: 'Breaking a problem into smaller parts' – without specifying that the 'smaller parts' are sub-problems rather than saying sequences of instructions or loops was insufficient.
		ii	1 mark for each e.g. • Removal of visual elements such as buildings on the ground • Simplification of controls • Focus on important elements such as weather, height, speed	3	**Examiner's Comments** Many candidates gave detailed and relevant examples that were suitable within the context of a flight simulator. However, there were many unqualified responses that were too vague. Unqualified responses such as 'terrain' did not go far enough to explain how the terrain in the simulation would differ from reality. To gain marks candidates had to make it clear exactly how the difference identified differed between reality and the simulation.
		iii	1 mark for each to max 2 e.g. • Reduce memory requirements • Reduce processing requirements • Simplify the problem being solved	2	**Examiner's Comments** It was noticeable that many candidates reiterated definitions of abstraction rather than identifying the actual reasons why abstraction is used, thus scoring no marks for not answering the question. Again, there were also many vague and unqualified responses. Responses such as 'focus on key aspects' did not identify the reason why abstraction is used in the design, whereas clearing specifying 'focus on key aspects allows coding/development time to be saved' would make the reason clear.
			Total	**7**	
8			1 mark for definition • Removal of unnecessary detail / Simplification to allow development of a program more easily 1 mark to max 2 for application e.g.	3 AO1.1 (1) AO2.1 (1) AO2.2 (1)	Allow other suitable examples that are relevant to the scenario in the question.

			• The actual movements are represented by vertices/lines • State of the move is represented by a letter/symbol rather than the actual move position • Tree does not show details about what the moves are		
			Total	**3**	
9		i	1 mark per bullet up to a maximum of 3 marks, e.g: • Classroom displays have been removed • People have been removed / simplified with symbols • Wall/carpet colours/details have been removed • Diagram is not to scale.	3 A02.1 (3)	Allow other suitable responses that are applied to the scenario in the question.
		ii	1 mark per bullet up to a maximum of 2 marks for each benefit (4 marks maximum in total), e.g: • Reduced development time (1) as factors that can detract from the program can be ignored (1) • Program more likely to solve the problem (1) as unnecessary aspects will not detract from the main purpose of the program (1) • Reduces complexity of programming code (1) therefore can run on lower spec computers(1)	4 A01.2 (2) A02.1 (2)	
			Total	**7**	