

Unit 2: Systems Software (2b. Applications Generation, A Level Only Content)

Marks: /22

Answer all the questions.

1. Livid Lizards is a computer game in which players get to fire lizards from a cannon to knock down walls. Players get to pick different types of lizards, each with qualities and special powers.

The game is coded using an object-oriented language. Below is the code for the lizard class:

```
class Lizard

    private speed
    private mass
    private size

    public procedure new(givenSpeed, givenMass, givenSize)
        speed=givenSpeed
        mass=givenMass
        size=givenSize
    endprocedure

    public function breakBlock(brick)
        if speed*mass>=brick.getStrength() then
            speed= ((speed*mass)-brick.getStrength()) /mass;
            return true
        else
            return false
        endif
    endfunction

    ...
    ...
    ...

endclass
```

The game uses a 2D graphics library. Explain why a linker would need to be used after compilation.

[3]

2. State **three** benefits of using library routines when a program is written.

1

2

3

[3]

3. Describe what happens during syntax analysis, when code is compiled.

[5]

4. A software development team is writing a word game.

The team is using Rapid Application Development.


The software team use a prebuilt library to create the Graphical User Interface.

(i) Give **two** advantages to the software team of using a library.

1 -----

2 -----

[2]

(ii)  The program is compiled. Explain the process of compilation including how code from the library becomes part of the finished program, justifying why each stage is necessary.

Question		Answer/Indicative content	Marks	Guidance
1		<ul style="list-style-type: none"> The user running the program will not necessarily have the library installed on their machine (1) therefore the relevant code needs to be included within the final executable (1) – it is the job of the linker to combine this code (1). 	3	Up to 3 marks for a valid explanation.
		Total	3	
2		Relatively error free / has already been tested Ready to use / saves time/already been written Used multiple times / common tasks / reduces repeated code Programmer expertise Different source languages	3	Examiner's Comments Excellently answered, very few candidates had any problem with this question and most candidates gained at least two out of the three marks.
		Total	3	
3		2 Marks from this section <ul style="list-style-type: none"> Statements / tokens are checked... ... against the rules / grammar of the language valid example given 3 Marks from this section <ul style="list-style-type: none"> Errors reported as a list Error diagnostics given Detail added to symbol table... ...eg data type / scope / address Receives output from lexical analysis / passes code to code generation 	5	No syntax check Examiner's Comments Again the Principal Examiner was looking for more technical knowledge with this question and it was apparent from the range of answers given that this was one question that showed true understanding of the subject.
		Total	5	

Question			Answer/Indicative content	Marks	Guidance
4		i	<p>Saves time / money as prewritten (1)</p> <p>Draws on expertise of other programmers (1)</p> <p>Pre-tested (so likely to work) (1)</p> <p>Can have been written in a different language (1)</p> <p>(Max 2)</p>	<p>2</p> <p>(AO1.2)</p>	<p>Examiner's Comments</p> <p>Those candidates who cited generic advantages of using subroutines as opposed to library routines did not gain credit. The question asked for advantages to the team of using a library.</p>
		ii	<p>Mark Band 3–High Level (7–9 marks)</p> <p>The candidate demonstrates a thorough knowledge and understanding of how source code is compiled and library code incorporated. The material is generally accurate and detailed.</p> <p>The candidate is able to apply their knowledge and understanding directly and consistently to the context provided. Evidence / examples will be explicitly relevant to the explanation.</p> <p>The candidate provides a thorough discussion which is well balanced. Evaluative comments are consistently relevant and well-considered.</p> <p>There is a well-developed line of reasoning which is clear and logically structured. The information presented is relevant and substantiated.</p>	<p>AO1.1 (2)</p> <p>AO1.2 (2)</p> <p>AO2.1 (2)</p> <p>AO3.3 (3)</p> <p>9</p>	<p>Points may include but are not limited to:</p> <p>AO1 Knowledge and Understanding</p> <p>The compiler is effectively a group of programs.</p> <p>The stages of compilation are: lexical analysis, syntax analysis, code generation and optimisation.</p> <p>A linker is then used to combine the object code with the library code to make the final executable.</p> <p>AO2.1 Application</p> <p>Source code is input into a compiler program.</p> <p>The first stage is lexical analysis in which..</p> <p>Comments and whitespace are removed</p> <p>Variables, and subroutines stored in symbol table</p> <p>Which also holds data such as scope and data type</p> <p>Code is converted to a series of tokens</p>

Question	Answer/Indicative content	Marks	Guidance
	<p>Mark Band 2–Mid Level (4–6 marks) The candidate demonstrates reasonable knowledge and understanding of how source code is compiled and library code incorporated; the material is generally accurate but at times underdeveloped.</p> <p>The candidate is able to apply their knowledge and understanding directly to the context provided although one or two opportunities are missed. Evidence / examples are for the most part implicitly relevant to the explanation.</p> <p>The candidate provides a sound discussion, the majority of which is focused. Evaluative comments are for the most part appropriate, although one or two opportunities for development are missed.</p> <p>There is a line of reasoning presented with some structure. The information presented is in the most part relevant and supported by some evidence.</p> <p>Mark Band 1–Low Level (1–3 marks) The candidate demonstrates a basic knowledge of how source code is compiled and / or library code incorporated; the material is basic and contains some inaccuracies. The candidate makes a limited attempt to apply acquired knowledge and understanding to the context provided.</p> <p>The candidate provides a limited discussion which is narrow in focus. Judgments if made are weak and unsubstantiated. The information is basic and communicated in an unstructured way. The information is supported by limited evidence and the relationship to the evidence may not be clear.</p>		<p>The series of tokens and symbol table is passed onto the next stage, syntax analysis: Here the code is checked to ensure it follows the rules of the language.</p> <p>This is often accomplished by placing the tokens into a (abstract syntax) tree.</p> <p>Where it breaks the rules of the language errors are generated.</p> <p>If no rules are broken then it's passed on to the next stage...</p> <p>..Which is code generation.</p> <p>Here the object code (accept machine code) is created.</p> <p>(i.e. the binary that is executed by the processor)</p> <p>This code may be inefficient..</p> <p>.. it may contain unnecessary instructions or groups of instructions that can be replaced by simpler ones.</p> <p>Code from the library is likely already compiled.</p> <p>And may well have been written in a different language to the main program.</p> <p>The main program source code will have contained lines importing the library code.</p> <p>A program called a linker can incorporate the code from the library with the main program... ...into a single executable file. An alternative approach is for the main executable to link to the compiled library code (i.e. dynamic linking).</p>

Question	Answer/Indicative content	Marks	Guidance
	<p>0 marks No attempt to answer the question or response is not worthy of credit.</p>		<p>AO3.3 Evaluation Lexical analysis is necessary to put the code into a format which can be read and processed (i.e. parsed) by the syntax analyser.</p> <p>Syntax Analysis is necessary to ensure the code is valid in as much as it meets all the structural rules of the language. This guarantees it will run (though it might not do as expected and may still have occurrences of runtime errors).</p> <p>Code generation is necessary to turn the code into a format that the processor can understand (i.e. binary machine code).</p> <p>The code optimisation whilst not necessary, does ensure the code runs quicker or using less memory.</p> <p>Linking is necessary to ensure the library code is incorporated into the final program.</p> <p>Examiner's Comments Candidates were assessed on the quality of their extended response in this question. Many candidates explained the stages of compilation very well. Some went on to describe how code from the library becomes part of the finished program equally well. Few justified why each stage was necessary. Many candidates scored well on this question.</p>
	Total	11	