

# F452 Programming Techniques and Logical Methods

Question		Expected Answers	Marks	
1	(a)	MemberID: <ul style="list-style-type: none"> <li>• String/Text/Alphanumeric</li> <li>• 5</li> </ul> Name: <ul style="list-style-type: none"> <li>• String/Text/Alphanumeric</li> <li>• 10 – 30</li> </ul> DateJoined: <ul style="list-style-type: none"> <li>• Date</li> <li>• 2 or 4 or 8</li> </ul> CurrentAverage <ul style="list-style-type: none"> <li>• Real/Floating Point/Single/Double</li> <li>• 4 or 8</li> </ul> GamesPlayed <ul style="list-style-type: none"> <li>• Integer</li> <li>• 2, 4 or 8</li> </ul>	[10]	
	(b)	<ul style="list-style-type: none"> <li>• Answers in part a added up</li> <li>• Multiply by 2000 members</li> <li>• Add 10% (for overheads)</li> <li>• Divide by 1000(or 1024) to get kB</li> <li>• Answer between 45kB and 126kB</li> </ul>	[5]	
	(c)	(i)	Any 4 of: <ul style="list-style-type: none"> <li>• Records are arranged in order of a primary <u>key</u>...</li> <li>• ... which in this case will be MemberID</li> <li>• An index is kept which is used to jump to <u>groups/blocks</u> of records</li> <li>• Eg the index could hold the positions of the first record with letters A, B, C etc</li> <li>• The index must be in the same order as the records</li> <li>• Mention of multiple indices</li> </ul>	[4]
		(ii)	Any 2 of: <ul style="list-style-type: none"> <li>• Given the large number of records...</li> <li>• ... accessing a specific record is faster</li> <li>• ... as you do not have to search sequentially from the beginning.</li> </ul>	[2]
	(d)	(i)	<ul style="list-style-type: none"> <li>• “GamesPlayed &lt;= 50” is TRUE (so take left branch)</li> <li>• “DateJoined &lt; 1 year ago” is FALSE (so take right branch)</li> <li>• Category = Improver</li> </ul>	[3]
		(ii)	<ul style="list-style-type: none"> <li>• “GamesPlayed &lt;= 50” is FALSE (so take right branch)</li> <li>• “CurrentAverage &lt; 180” is FALSE (so take right branch)</li> <li>• Category = Pro</li> </ul>	[3]

Question		Expected Answers	Marks
	(e)	<ul style="list-style-type: none"><li>• CurrentAverage <math>\geq</math> 200 (or equivalent) at * ...</li><li>• ... in a rhombus</li><li>• YES and NO labels are present</li><li>• Category = Pro for correct branch/if CurrentAverage <math>\geq</math> 200</li><li>• Category = Improver for correct branch/if CurrentAverage <math>&lt;</math> 200</li><li>• Flowchart reconnected correctly</li><li>• Include diagram...</li></ul> (Accept other solutions which produce the correct result.)	<b>[6]</b>

Question		Expected Answers	Marks
2	(a)	<ul style="list-style-type: none"> <li>Sequence</li> </ul>	[1]
	(b)	(i) <ul style="list-style-type: none"> <li>Any 3 of:               <ul style="list-style-type: none"> <li>(A variable) which holds an item of data...</li> <li>... which is supplied/passed to a subroutine/procedure/function</li> <li>It is given an identifier when the subroutine is defined</li> <li>It is substituted by an actual value when the subroutine is called</li> </ul> </li> </ul>	[3]
		(ii) <ul style="list-style-type: none"> <li>L, W, T</li> </ul>	[1]
	(c)	<ul style="list-style-type: none"> <li>a = 8</li> <li>b = 5</li> <li>c = 40 (allow follow through)</li> <li>d = 44 (allow follow through)</li> </ul>	[4]
	(d)	Example: 06 OUTPUT "You will need" + d + " tiles." Award marks for any 2 of: <ul style="list-style-type: none"> <li>Concatenation has been used correctly</li> <li>Output is user friendly/sentence or label + value</li> </ul>	[2]
	(e)	<p><b>High level response [6-8 marks]</b>            Candidates will answer the question with complete and comprehensive explanations of how to rewrite the code, justifying each point made.            The information will be presented in a structured and coherent form, which may include snippets of programming code as illustrations of points made. There will be few, if any, errors in spelling, grammar and punctuation. Technical terms will be used appropriately and correctly.</p> <p><b>Medium level response [3-5 marks]</b>            Candidates will answer the question showing an awareness of a number of techniques for internally documenting code, with some reference to the code provided.            The information will be presented in a structured format, giving examples of code to illustrate the points being made with few explanations or justifications. There may be occasional errors in spelling, grammar and punctuation. Technical terms will be mainly correct.</p> <p><b>Low level response [0-2 marks]</b>            Candidates will demonstrate a limited understanding. A few techniques for writing easily maintainable program code may be mentioned, but not related to the code provided. Information may be a list of points, with few or no descriptions. Information will be poorly expressed and there will be a limited, if any, use of technical terms. Errors of grammar, punctuation and spelling may be intrusive.</p>	

Question		Expected Answers	Marks
	(e) cont'd	<p>Points that may be made:</p> <ul style="list-style-type: none"> <li>• The variables/procedures should be given more descriptive names such as length, width, tile/findNumberOfTiles, etc using consistent conventions making it easier to tell what the variables represent/procedures do</li> <li>• Code should be indented to show program constructs/blocks for example PROCEDURE/END PROCEDURE making it easier to trace the code and check for incorrect blocks</li> <li>• Add comments to the code and separate it into logical sections making the code easier to read</li> </ul>	<b>[8]</b>