Mark Scheme

Additional Sample Assessment Materials

Pearson BTEC Level 3 - Computing

Unit 1: Principles of Computer Science

Edexcel and BTEC Qualifications

Edexcel and BTEC qualifications come from Pearson, the world's leading learning company. We provide a wide range of qualifications including academic, vocational, occupational and specific programmes for employers. For further information visit our qualifications websites at www.edexcel.com or www.btec.co.uk for our BTEC qualifications.
Alternatively, you can get in touch with us using the details on our contact us page at www.edexcel.com/contactus.

If you have any subject specific questions about this specification that require the help of a subject specialist, you can speak directly to the subject team at Pearson.
Their contact details can be found on this link: www.edexcel.com/teachingservices.

You can also use our online Ask the Expert service at www.edexcel.com/ask. You will need an Edexcel username and password to access this service.

Pearson: helping people progress, everywhere
Our aim is to help everyone progress in their lives through education. We believe in every kind of learning, for all kinds of people, wherever they are in the world. We've been involved in education for over 150 years, and by working across 70 countries, in 100 languages, we have built an international reputation for our commitment to high standards and raising achievement through innovation in education. Find out more about how we can help you and your students at: www.pearson.com/uk

Unit 1: Principles of Computer Science – sample marking grid

General marking guidance

- All learners must receive the same treatment. Examiners must mark the first learner in exactly the same way as they mark the last.
- Marking grids should be applied positively. Learners must be rewarded for what they have shown they can do, rather than be penalised for omissions.
- Examiners should mark according to the marking grid, not according to their perception of where the grade boundaries may lie.
- All marks on the marking grid should be used appropriately.
- All the marks on the marking grid are designed to be awarded. Examiners should always award full marks if deserved. Examiners should also be prepared to award zero marks, if the learner's response is not rewardable according to the marking grid.
- Where judgement is required, a marking grid will provide the principles by which marks will be awarded.
- When examiners are in doubt regarding the application of the marking grid to a learner's response, a senior examiner should be consulted.

Specific marking guidance

The marking grids have been designed to assess learner work holistically. Rows in the grids identify the assessment focus/outcome being targeted. When using a marking grid, the 'best fit' approach should be used.

- Examiners should first make a holistic judgement on which band most closely matches the learner's response and place it within that band. Learners will be placed in the band that best describes their answer.
- The mark awarded within the band will be decided based on the quality of the answer, in response to the assessment focus/outcome and will be modified according to how securely all bullet points are displayed at that band.
- Marks will be awarded towards the top or bottom of that band, depending on how they have evidenced each of the descriptor bullet points.

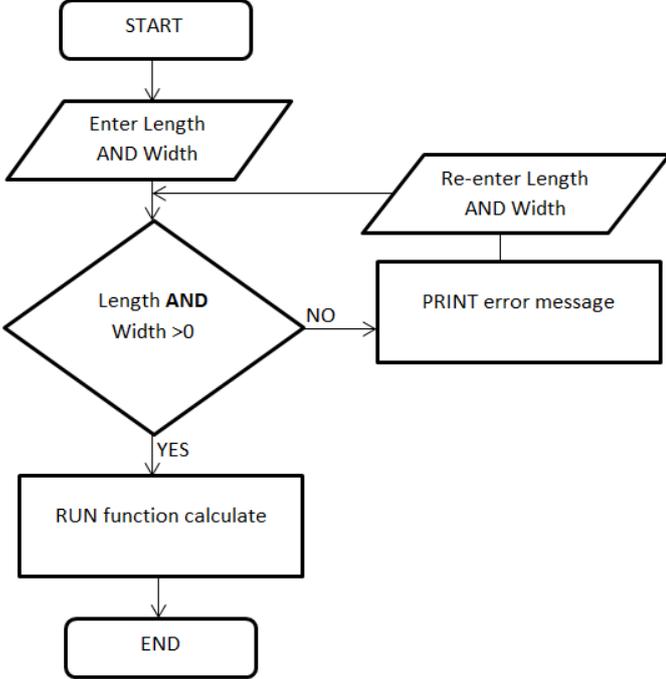| Question Number	Answer	Mark
1(a)	Award **1** mark for any of the following up to a maximum of **2** marks. • Check the length and width are entered • Convert feet to (square) metres • Calculate the amount of square metres/area • Multiply the price for one square metre by the number of square metres required • Deduct the discount if more than 20 square metres are ordered	(2)

Question Number	Answer	Mark
1(b)	Award **1** mark for any of the following up to a maximum of **3** marks. • Number of feet in one metre • How to calculate area/square metres • Price for one square metre • Discount rate (if 20 or more square metres are ordered)	(3)

Question Number	Answer	Mark
1(c)	Award **1** mark for the appropriate variable and **1** mark for a correct data type and an additional explanation. • **Variable**: Length or Width (1) • **Data Type**: Real/Floating Point (1) as the measurement is likely to include a decimal / will need precision (1) • **Variable**: Metres/Square Metres (1) • **Data Type**: Integer (1) as the number required has to be rounded upwards to the nearest whole number (1) • **Variable**: Cost (1) • **Data Type**: Real/Floating Point (1) as the price per square metre contains a decimal (1)	(3)

Question Number	Answer	Mark
1(d)	Award **one** mark for identification and **one** additional mark for appropriate expansion. • **INPUT** to allow the user to enter their measurements (1) so that it can be assigned to a variable (1) • **ROUND** to round the number of metres up (1) to the nearest whole number (1)	(4)

Question Number	Answer	Mark
1(e)	Award **one** mark for identification and **one** additional mark for each appropriate expansion up to **three** marks. An IF statement will only check once (1) whereas a WHILE will check continuously (1) until valid data is entered (1) Accept any other relevant phrasing/wording.	(3)

Question Number	Answer	Mark
1(f)	Award **one** mark for identification and **one** additional mark for each appropriate expansion up to **three** marks. To ensure that the measurements entered are within two boundaries (1) to ensure that the data entered is more reasonable (1) such as measurements should be >0 (and below a pre-set value) (1)	(3)

Question Number	Answer	Mark
1(g)	**POSSIBLE ALGORITHM:** Guidance: 1 mark for each of the following points: • Decision box showing correct logic for checking that the length AND width have been entered (e.g. length AND width >0) (1) • Yes route correctly labelled and leads to running the function 'calculate' (1) • No route correctly labelled, leads to error message and prompts user to re-enter the measurements (1) • Appropriate conventions for a flow chart (e.g. correct BCS symbols, correct data flow) (1)	(4)

Question Number	Answer	Mark
2(a)	Award **one** mark for each correct value in the table up to a maximum of **four** marks. <table><tr><td>Data Entered</td><td>Problem Value</td></tr><tr><td>3</td><td>3 (Variable Inputted)</td></tr><tr><td>yes</td><td>3 (no calculation)</td></tr><tr><td>yes</td><td>2 (3-1=2)</td></tr><tr><td>no</td><td>2 (no calculation)</td></tr><tr><td>yes</td><td>1 (2-1=1)</td></tr></table> Additional guidance- allow follow through for incorrect values that result from correct application of a process that uses previously incorrect value.	(4)

Question Number	Answer	Mark
2(b)	Award **one** mark for identification and **one** additional mark for appropriate expansion up to **four** marks. • Frequently used code only needs to be written once (1) reducing development time (1) • Improves reusability of code/allows sections of code to be shared with others (1) by creating self-contained blocks of code (1) • To make the programming code more efficient/reduces the amount of code to be loaded (1) which therefore increases the performance (of the code) (1) • To reduce the amount of errors in the code (1) if the code needs to be adapted it only needs changing once (1) • Aids problem solving (1) by allowing complex problems to be broken down (deconstruction) (1)	(4)

Question Number	Answer	Mark
2(c)	An explanation of why the function is defined before it is called such as: The programming language used is interpreted language (1) so the programming code is run in sequence (from top to bottom) (1) by defining it (the function) first it will be loaded into the memory (1) so it is ready to be executed when called (1) Accept any other relevant phrasing/wording.	(4)

Question Number	Answer	Mark
2(d)	Award **one** mark for identification and **one** additional mark for appropriate expansion up to **three** marks. • The problems variable is declared at the start of the program (1) but it needs to be changed inside a function (1) the global command enables the variables declared outside of a function to be accessed/ updated (by other parts of the program) (1) If not, the problems variable (in part D) would only be assigned the value locally (1) so other functions/other parts of the code would use the original value (1) causing the program to produce incorrect results/not function correctly (1)	(3)

Question Number	Answer	Mark
2(e)	A explanation of why the error has occurred such as : User input is defined as a string (1). The IF statement then compares this variable against an integer/the value 0 (1). As they are not the same data type the comparison cannot be made (1)	(3)

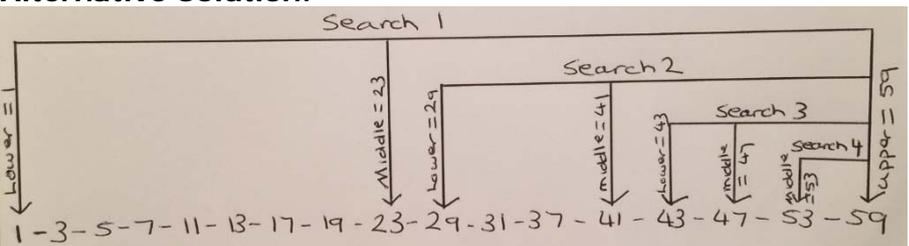| Question Number	Answer	Mark
2(f)	Award **one** mark for identification and **one** additional mark for appropriate expansion up to **three** marks. Not all lines of the code need to be executed (1) as the program will attempt to solve a varying number of problems with a printer (1) therefore the selection statements will execute the correct part of the code (that solves their specific problem) (1)	(3)

Question Number	Answer	Mark
3(a)	Award **one** mark for each identification up to a maximum of **three** marks. • Incorrect operator has been used / a less than operator (<) should have been used • First ELSE should be an IF statement • Incorrect variable has been printed/ status has been misspelt	(3)

Question Number	Answer	Mark
3(b)	Award **1** mark for identification and **1** additional mark for appropriate expansion up to a maximum of **3** marks each. • There are multiple distinct data sets (1) which have to be stored separately but in related structures (1) this structure allows a set of scores for each student that can be accessed by a single identifier (1)	(3)

Question Number	Answer	Mark
3(c)	**Example solution**	

Original Data	93	65	81	62	36	94	45
Insertion A	<u>65</u>	93	81	62	36	94	45
Insertion B	65	<u>81</u>	93	62	36	94	45
Insertion C	<u>62</u>	65	81	93	36	94	45
Insertion D	<u>36</u>	62	65	81	93	94	45
Insertion E	36	62	65	81	93	<u>94</u>	45
Final Sorted Data	36	<u>45</u>	62	65	81	93	94

Award **one** mark for each identification up to a maximum of **six** marks.

- Insertion A shows 65 move to first position
- Insertion B shows 81 move to second position
- Insertion C shows 62 move to first position
- Insertion D shows 36 move to first position
- Insertion E shows 94 move to sixth position
- Final sorted data shows 45 move to second positon

Additional Guidance - Marks should be awarded for correct insertion. Therefore if the previous insertion is incorrect but the candidate has followed through and inserted the number correctly (based on the previous search on the next insertion) then marks should be awarded.

For marks to be awarded for each insertion, the candidate must have written the previous sort/row to show that the insertion has been updated correctly.

(6)

Question Number	Answer	Mark
3(d)	Answers will be credited according to the learner's demonstration of knowledge and understanding of the material using the indicative content and levels descriptors below. The indicative content that follows is not prescriptive. Answers may cover some/all of the indicative content but should be rewarded for other relevant answers. Indicative content: • Insertion sorts will sort the data faster as the number of scores to be sorted is small. If the number of scores in the array was larger than a quick sort would be more efficient as insertion sorts could possibly need to shift all numbers in the array upwards, therefore taking longer to complete the sort. • Insertion sorts are easy to create the programming code for. Although the number of scores is likely to increase over an academic year the number of scores in the sort is unlikely to reach an amount that will decrease the performance of the insertion sort. Therefore it will be more time efficient for the programmer to create an insertion sort as quick sorts can take a long time to create the code for. As quick sorts would not give a better performance on the small amount of scores within the array, it may not be worth spending more time creating the code. • There is a small possibility that the test scores may already be in order. This is ideal for insertion sorts as the best case for this sort is when the list is already/mostly sorted. However, this is actually the worst case for quick sorts as it can be difficult to choose a pivot that allows the data to be sorted efficiently, which can therefore decrease performance of the sort. • Insertion sorts are less memory intensive. It only needs one extra storage location to temporarily score a number before it is inserted into the correct position. Quick sorts rely heavily on recursion and therefore more memory is used up running more code and carrying out more comparisons to put numbers into the correct position, which can decrease performance.	(8)

Mark scheme (award up to 8 marks) refer to the guidance on the cover of this document for how to apply levels-based mark schemes*.

Level	Mark	Descriptor
Level 0	0	No rewardable material.
Level 1	1-2	Technical vocabulary is used but it is not used appropriately to support arguments in relation to the issues of the question. Issues are identified but chains of reasoning are not made,

| | | leading to a superficial understanding of the relative importance of issues to the scenario.

Arguments are not linked to the given scenario. |
|---------|-----|---|
| Level 2 | 3-5 | Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question.

There is consideration of relevant issues, using logical chains of reasoning, but it does not reflect on their relative importance to the given scenario.

Various elements of the question are considered but it does not always link arguments to the given scenario. |
| Level 3 | 6-8 | Fluent and accurate technical vocabulary is used to support arguments that are relevant to the issues of the question.

There is a balanced and wide-ranging consideration of relevant issues, using coherent and logical chains of reasoning that show a full awareness of their relative importance to the given scenario.

Various elements of the question are carefully considered and arguments are clearly linked to the given scenario. |

Question Number	Answer	Mark
4a	Answers will be credited according to the learner's demonstration of knowledge and understanding of the material using the indicative content and levels descriptors below. The indicative content that follows is not prescriptive. Answers may cover some/all of the indicative content but should be rewarded for other relevant answers. **Example Solution:** Step 1 <table><tr><th>Lower Boundary</th><th>Middle Boundary</th><th>Upper Boundary</th><th>Comments</th></tr><tr><td>1</td><td>23</td><td>59</td><td>The whole list is used in the search. 1 is the lower boundary, 59 is the upper boundary and so therefore the middle boundary is 23.</td></tr></table> Step 2 <table><tr><th>Lower Boundary</th><th>Middle Boundary</th><th>Upper Boundary</th><th>Comments</th></tr><tr><td>29</td><td>41</td><td>59</td><td>The number 53 does not exist between the lower (value 1) and middle boundary (value 23) so therefore this part of the array is eliminated from the search. The new lower boundary therefore changes to 29, 59 continues to be the upper boundary and the new middle boundary is 41.</td></tr></table> Step 3 <table><tr><th>Lower Boundary</th><th>Middle Boundary</th><th>Upper Boundary</th><th>Comments</th></tr><tr><td>43</td><td>47</td><td>59</td><td>The number 53 does not exist between the lower (value 29) and middle boundary (value 41) so therefore this part of the array is eliminated from the search. The new lower boundary therefore changes to 43, 59 continues to be the upper boundary and the new middle (value) boundary is 47.</td></tr></table> Step 4 <table><tr><th>Lower Boundary</th><th>Middle Boundary</th><th>Upper Boundary</th><th>Comments</th></tr><tr><td>N/A</td><td>53</td><td>59</td><td>The number 53 does not exist between the lower (value 43) and middle boundary (value 47) so therefore this part of the array is eliminated from the search. As there are only two values left in the array, there is no lower boundary, the new middle boundary is 53 and 59 continues to be the upper boundary. As 53 is the middle boundary the value has been found.</td></tr></table> **Alternative Solution:** 	(6)

	Award one mark for each identification up to a maximum of **six** marks	
	• Identifying that 4 different searches are required.	
	• Step 1 - Identify 1 as the lower boundary, 23 as the middle boundary and 59 as the upper boundary.	
	• Step 2 - Identifying 29 as the lower boundary, 41 as the middle boundary and 59 as the upper boundary.	
	• Step 3 - Identifying 43 as the lower boundary, 47 as the middle boundary and 59 as the upper boundary.	
	• Step 4 - Identifying 53 as the middle boundary and 59 as the upper boundary.	
	• Correctly identifying no lower boundary / the same lower and middle boundary on step 4.	
	Additional Guidance - Marks should be awarded for correct searching. Therefore if the previous step is incorrect but the candidate has followed through and applied on the next steps correctly then marks should be awarded.	

Question Number	Answer	Mark
4(b)	Award **one** mark for identification and **one** additional mark for appropriate expansion up to a maximum of **3** marks. **Line 6** (1) The address would not be positioned correctly (1) as the centre tag has been misspelled (1) **Line 9** (1) The details heading would not be displayed (1) as the tag has been displayed incorrectly (1) **Line 18** (1) The email address would not appear in bold (1) as the strong tag has been misspelled (1) **Line 22** (1) The image would not be displayed (1) as the file type of the image has been incorrectly identified (1)	(3)

Question Number	Answer	Mark
4(c)	Answers will be credited according to the learner's demonstration of knowledge and understanding of the material using the indicative content and levels descriptors below. The indicative content that follows is not prescriptive. Answers may cover some/all of the indicative content but should be rewarded for other relevant answers. Indicative content: • HTML does not allow server-side processing. In order for customers to view previous animal treatments the programming code will need to search for their data on a central server. However, HTML code is run on the client side (on their browser) and so therefore does not store or have the functionality to display stored data. • The surgery needs to display dynamic data that is relevant to each customer; however, HTML is a markup language and so therefore is only ideal for displaying static information. Therefore HTML can only display the same information for all customers who view the website. • Although HTML has the ability to carry out some conditional statements for different web browsers, HTML is not a true programming language and so therefore the code would need to incorporate other languages such as Java Script that have increased programming functionality to give the website increased user interactivity.	(6)

Mark scheme (award up to 6 marks) refer to the guidance on the cover of this document for how to apply levels-based mark schemes*.

Level	Mark	Descriptor
Level 0	0	No rewardable material.
Level 1	1-2	Technical vocabulary is used but it is not used appropriately to support arguments in relation to the issues of the question. Issues are identified but chains of reasoning are not made, leading to a superficial understanding of the relative importance of issues to the scenario. Arguments are not linked to the given scenario.
Level 2	3-4	Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question.

| | | There is consideration of relevant issues, using logical chains of reasoning, but it does not reflect on their relative importance to the given scenario.

Various elements of the question are considered but it does not always link arguments to the given scenario. |
| Level 3 | 5-6 | Fluent and accurate technical vocabulary is used to support arguments that are relevant to the issues of the question.

There is a balanced and wide-ranging consideration of relevant issues, using coherent and logical chains of reasoning that show a full awareness of their relative importance to the given scenario.

Various elements of the question are carefully considered and arguments are clearly linked to the given scenario. |

Question Number	Answer	Mark
4(d)	Answers will be credited according to the learner's demonstration of knowledge and understanding of the material using the indicative content and levels descriptors below. The indicative content that follows is not prescriptive. Answers may cover some/all of the indicative content but should be rewarded for other relevant answers.

DATA STRUCTURE
- Currently, the appointment data is stored within the actual code. This therefore means the surgery will need to have various copies of the programming code for each day. Therefore, over time, this could prove problematic.
- Also, by expanding the programming code, it increases the likelihood of appointment data being accidentally lost as this is stored within the programming code.
- To avoid this, the programmer may want to use file handling so the programming code / structure could be adapted without actually changing the appointment data (as this will be stored in a separate file).

CODE COMMENTS
- There are very limited comments provided by the original programmer.
- The comments that have been provided do not actually inform a third party what the | |

code does.
A third party may not understand the code to be able to adapt it or will have to spend time working out what the code actually does before they can carry out maintenance work on it.
- However, an experienced programmer would probably be able to work out what the code does, but may have to spend time annotating the code before they carry out maintenance.
- If the code is expanded and new code comments continue to be limited/poor then it could make it problematic in the future to maintain the code.

BLOCKS
- The programmer has generally structured the code well into 4 main blocks.
- This breaks the code down and makes it easier to navigate around the main parts of the program.
- A third party can therefore clearly see the main focus for each part of the programming code. Block A shows lists that have been set up, Block B shows variables being set, Block C contains a function that will search for appointments and, finally, Block D allows for user input.

LOOPING
- The programmer has made good use of while and for loops to make the code very efficient.
- The number of lines in the code is therefore significantly reduced, which will therefore reduce the amount of code that needs to be maintained.
- If one part of the part of the programming code is changed then it's 'knock on' affect will be smaller as there is less code that may also need to be changed as a result.

FUNCTIONS
- The programmer has created a function which is a 'self-contained' block of code.
- This therefore means that if the code needs to be changed, it only needs to be changed once within the function but can be run many times within the code.
- This function could also be reused in the

		future if the program was expanded to improve future efficiency.	
		INDENTATION • The programmer has made very good use of indentation in order to separate different statements in the program. • The indents allow a third party to clearly see the flow of the program, which will help the programmer to work out what the code does and which parts of the code relate to each other. **VARIABLE/FUNCTION NAMES** • The programmer has chosen poor variable/function names within the code. The variable names do not accurately reflect the data that is being stored by the program. • This can make it difficult to keep track of the variables throughout the program and make it difficult to understand why that variable is being used and how it could be used again in the future if the code was expanded. • The code uses a global variable approach, which may cause problems if the code is expanded. If more functions are developed in the future then it's more difficult to maintain the integrity of the data stored in the variable as it could be accessed and changed by many different functions.	(12)

Mark scheme (award up to 12 marks) refer to the guidance on the cover of this document for how to apply levels-based mark schemes*.

Level	Mark	Descriptor
Level 0	0	No rewardable material.
Level 1	1-4	Technical vocabulary is used but it is not used appropriately to support arguments in relation to the issues of the question. Issues are identified but chains of reasoning are not made, leading to a superficial understanding of the relative importance of issues to the scenario. No conclusion is presented or is generic.
Level 2	5-8	Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question. There is consideration of relevant issues, using logical chains of reasoning, but it does not reflect on their relative importance to the given scenario.

		An attempt at a conclusion is presented that links arguments to the given scenario but is not justified in that it does not reflect the careful consideration of both sides of the argument.
Level 3	9-12	Fluent and accurate technical vocabulary is used to support arguments that are relevant to the issues of the question. There is a balanced and wide-ranging consideration of relevant issues, using coherent and logical chains of reasoning that show a full awareness of their relative importance to the given scenario. A fully justified conclusion is presented, that links arguments to the given scenario and that reflects the careful consideration of both sides of the argument, leading to a reasoned decision.

For more information on Edexcel qualifications, please visit our website
www.edexcel.com