



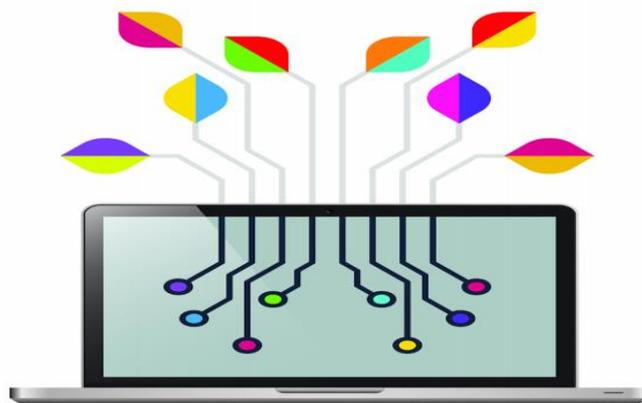
Pearson



Mark Scheme (Results)

Summer 2017

BTEC Level 3 National in Computing
Unit 1: Principles of Computer Science
(31768H)



BTEC Qualifications

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.btec.co.uk for our BTEC qualifications.

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

June 2017

31768H_1706_MS

All the material in this publication is copyright

© Pearson Education Ltd 2017

Unit 1: Principle of Computer Science

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
1a	<p>Solution</p> <pre> BEGIN INPUT age <input type="text" value="IF"/> age >=14 AND <=16: Discount = 30% <input type="text" value="ELIF"/> age =17 <input type="text" value="OR"/> =18: Discount = 20% <input type="text" value="ELIF"/> age => 50: Discount = 40% <input type="text" value="ELSE"/> : Discount = 0% END </pre> <p>Award 1 mark for any of the following up to 3marks:</p> <ul style="list-style-type: none"> • IF/ELIF/ELSEIF (1) in box 2 • OR(1) in box 3 • ELSE (1) in box 5 	(3)

Question Number	Answer	Mark
1b	<p>An explanation to contain three from:</p> <p>(Input/Age) needs to be converted to a number/integer (1)</p> <p>So calculations can take place / comparisons can be made to the numbers/(age) constants (1)</p> <p>Data types are not compatible (1)</p> <p>to avoid an error / inaccurate results (1)</p>	(3)

Question Number	Answer	Mark
1c	<p>Award one mark for identification of an event and two additional marks for appropriate handler up to 3 marks for each, such as:</p> <p>Event: Textbox clicked / get focus (1) Handler: change the box colour (1) to show where data is about to be entered (1).</p> <p>Event: Textbox click out / lose focus / field exit (1) Handler: a tick to appear(1) to confirm successful data entry (1).</p> <p>Event: Mouse hover over a button (1) Handler: onscreen help / tip text to appear (1) to inform the user what the button does (1).</p> <p>Event: Calculate button clicked(1) Handler: Validation can be run (1) to check that the member details have been entered / are present (1).</p> <p>Event: Peak / Off-Peak clicked (1) Handler: ensure that if one radio button is already selected (1) the other becomes unselected (1).</p> <p>Accept other suitable suggestions.</p> <p>Additional Guidance:</p> <ul style="list-style-type: none"> • Each event must clearly state an action (e.g. selected, clicked, hovered, typed, click out etc) • The event must be related to the event handler for maximum marks • Still award marks for event handlers if no event type has been stated. 	(6)

Question Number	Answer	Mark
1d	<p>An explanation to contain four from:</p> <ul style="list-style-type: none"> • The main loop has overall control of the program (1) • The main loop will run repeatedly (throughout the program)(1) • The main loop continues until an event/trigger takes places (1) • The correct call back function is called (1) • The code in the function is executed (1) • then the main loop continues (for other events) (1) <p>Additional guidance Allow alternative terminology for function. Allow examples of events from figure 1b to be used.</p>	(4)

Question Number	Answer	Mark
1e	<p>Award one mark for identification and one additional mark for appropriate expansion up to 3 marks, such as:</p> <ul style="list-style-type: none"> • The program responds to individual user actions/interactions (1) which will run a specific block of code (in any order) (1) rather than running all code sequentially (1) • The program responds to individual user actions (1) such as mouse clicks or data entry (1) which can be assigned to specific tasks / functions (1). • The developing environments tend to be visual (1) allowing the developer to draw the interface (1) and auto generate associated code (1). <p>Accept any alternative wording/phrasing</p>	(3)

Question Number	Answer	Mark
1f	<p>Award one mark for identification and one additional mark for appropriate expansion up to 3 marks for each.</p> <ul style="list-style-type: none"> • All data within a set must be the same data type (1) which could mean that each piece of data would have to be stored in a different set/cannot have identifiers assigned (1) which will reduce efficiency (1). • Data can be stored in a random order within a set (1) therefore data would be structured differently for each member stored (1) making it hard to locate data (1). • A set cannot store duplicated values (1) which will prove difficult when storing Boolean data types / such as...member name (1) meaning multiple data structures may be needed (1). <p>Additional guidance Points and expansions can be combined from different mark points if correct</p>	(3)

Question Number	Answer	Mark								
2(a)	<p>1 Mark for each of the following points:</p> <table border="1" data-bbox="400 389 1251 669"> <thead> <tr> <th data-bbox="400 389 906 421">Sally's Requirements</th> <th data-bbox="906 389 1251 421">Built-in Function</th> </tr> </thead> <tbody> <tr> <td data-bbox="400 421 906 501">The amount of money to be saved needs to be entered.</td> <td data-bbox="906 421 1251 501">INPUT</td> </tr> <tr> <td data-bbox="400 501 906 582">Only accept the first two digits entered for the interest rate.</td> <td data-bbox="906 501 1251 582">TRUNCATION</td> </tr> <tr> <td data-bbox="400 582 906 669">Only allow a number between 1 and 20 to be entered for the interest rate.</td> <td data-bbox="906 582 1251 669">RANGE</td> </tr> </tbody> </table> <p>ADDITIONAL GUIDANCE: Allow specific function names that are used in different programming languages. For example 'slice' in box 2 (which is a built-in function in Python used for truncation)</p>	Sally's Requirements	Built-in Function	The amount of money to be saved needs to be entered.	INPUT	Only accept the first two digits entered for the interest rate.	TRUNCATION	Only allow a number between 1 and 20 to be entered for the interest rate.	RANGE	(3)
Sally's Requirements	Built-in Function									
The amount of money to be saved needs to be entered.	INPUT									
Only accept the first two digits entered for the interest rate.	TRUNCATION									
Only allow a number between 1 and 20 to be entered for the interest rate.	RANGE									

Question Number	Answer	Mark
2b	<p>EXAMPLE SOLUTION:</p> <pre> graph TD Start([Start]) --> Input1[/BankAccounts = Input Number of Bank Accounts/] Input1 --> Input2[/Savings = Input Savings/] Input2 --> Input3[/Rate = Input Interest/] Input3 --> Process1[Interest = (Savings/100)*Rate] Process1 --> Process2[Total = Savings+Interest] Process2 --> Output[/Output Total/] Output --> Process3[BankAccounts = BankAccounts-1] Process3 --> Decision{BankAccounts = 0} Decision -- NO --> Input3 Decision -- YES --> End([End]) </pre>	(6)

Mark scheme (award up to 6 marks)		
Level	Mark	Descriptor
Level 0	0	No rewardable material.
1	1-2	<p>Structure of the algorithm uses some appropriate hierarchies/subdivision but clarity and/or readability is limited.</p> <p>Variable/object/process names are inappropriate and/or inconsistent</p> <p>Use of logical operations and sequence/structure of processes demonstrate limited accuracy.</p> <p>There is limited use of accepted conventions</p> <p>A partial and/or highly inefficient solution has been achieved.</p>
2	3-4	<p>Structure of the algorithm uses mostly appropriate hierarchies/subdivision to provide some clarity and readability.</p> <p>Variable/object/process names are mostly appropriate but there is some inconsistency</p> <p>Use of logical operations and sequences/structure are mostly accurate with only minor errors.</p> <p>Accepted conventions have been applied but there are some inconsistencies.</p> <p>An almost complete/inefficient solution has been achieved.</p>
3	5-6	<p>Structure of the algorithm uses appropriate and consistent hierarchies/subdivision providing clarity and readability.</p> <p>Variable/object/process names are appropriate and used consistently</p> <p>Use of logical operations and sequences/structures are accurate throughout.</p> <p>Accepted conventions have been used consistently</p> <p>A full and efficient solution been achieved.</p>

Question Number	Answer	Mark
2c	<p>Award one mark for identification and one additional mark for appropriate expansion up to 4 marks for each.</p> <p>Sally could use a loop (1) which will calculate the interest for each bank account (1) for the given number of comparisons/bank accounts (1) which therefore increases the efficiency/reduces duplication of code (1).</p> <p>The number of bank accounts to be compared is unknown / may change for every use (1) and a different interest rate may be needed for each bank account (1). However the same calculations are performed for each bank account (1) so an iteration means the same code can be used for each bank account (1).</p> <p>Additional Guidance Allow 'the program will repeatedly use the same block of code' or similar for loop</p>	(4)

Question Number	Answer	Mark
2d	<p>A linked explanation to contain any three from:</p> <ul style="list-style-type: none"> • Easy to determine what the variables are actually used for • Minimise errors by helping the programmer to select the variables • Other programmers will understand/be able to use the code • Flow through the program can be tracked more easily • Makes the code easier to maintain/expand/ aids debugging 	(3)

Question Number	Answer	Mark
2e	<p>Possible response:</p> <ul style="list-style-type: none"> • The variable AccountNo is set to 3 to repeat the sorting process for the three different amounts • The current code has three amounts stored in the actual code and so if the number of amounts change then this variable will also need. • A variable Amounts[item] will store the first number and Amounts[item+1] will store the second number. This is ideal as this sort will only compare two numbers at once. • If the second number is greater than the first number it is held in the Temp variable and the second number is copied to the first number's position in the list. The value that is held in the Temp variable is then copied to the second number's position in the list. • XOR operators could be used to carryout calculations to change the numbers so they become the other number however could overflow errors and reduce the efficiency. • After the first switch / comparison the AccountNo variable is reduced by 1 to repeat the process until all numbers have been sorted. This will ensure that the correct number of passes. 	(6)

Level	Mark	Description
0	0	
1	1-2	<p>Technical vocabulary is used but is not used appropriately to support arguments in relation to the issues of the question.</p> <p>Issues are identified but chains of reasoning are not made leading to a superficial understanding of the relative importance of issues to the context of the scenario.</p>
2	3-4	<p>Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question.</p> <p>A consideration of relevant issues using logical chains of reasoning but does not reflect upon their relative importance to the context of the scenario.</p>
3	5-6	<p>Fluent and accurate technical vocabulary is used to support arguments that are relevant to the issues of the question.</p> <p>A balanced and wide ranging consideration of relevant issues using coherent and logical chains of reasoning that shows a full awareness of their relative importance to the context of the scenario.</p>

Question Number	Answer	Mark
3a	<p>EXAMPLE SOLUTION: BEGIN</p> <p>FOR each line in file: number = 1 READ line of file position0=year position1=firstname position2=surname initial=position1[0] position3=initial username= position0+position3+position2+number WHILE username already exists: number=number+1 username= position0+position3+position2+number ENDWHILE PRINT username ENDFOR</p> <p>END</p> <p>Credit alternative solutions that use correct logic and would produce the expected outcome.</p>	(8)
Mark scheme (award up to 8 marks)		
Level	Mark	Descriptor
Level 0	0	No rewardable material.
1	1-2	<p>Structure of the algorithm uses some appropriate hierarchies/subdivision but clarity and/or readability is limited.</p> <p>Variable/object/process names are inappropriate and/or inconsistent</p> <p>Use of logical operations and sequence/structure of processes demonstrate limited accuracy.</p> <p>There is limited use of accepted conventions</p> <p>A partial and/or highly inefficient solution has been achieved.</p>
2	3-5	<p>Structure of the algorithm uses mostly appropriate hierarchies/subdivision to provide some clarity and readability.</p> <p>Variable/object/process names are mostly appropriate but there is some inconsistency</p> <p>Use of logical operations and sequences/structure are mostly accurate with only minor errors.</p> <p>Accepted conventions have been applied but there are some inconsistencies.</p> <p>An almost complete/inefficient solution has been achieved.</p>
3	6-8	Structure of the algorithm uses appropriate and consistent hierarchies/subdivision providing clarity and readability.

	<p>Variable/object/process names are appropriate and used consistently</p> <p>Use of logical operations and sequences/structures are accurate throughout.</p> <p>Accepted conventions have been used consistently</p> <p>A full and efficient solution been achieved.</p>
--	---

Question Number	Answer	Mark
3b	<p>Award one mark for identification and one additional mark for appropriate expansion up to 4 marks.</p> <p>Starts at first item/index in list (1) compares list item to username (1) repeats for each item/index (1) until a match is found (1)</p>	(4)

Question Number	Answer	Mark
3c	<p>Possible responses:</p> <p>GENERAL</p> <ul style="list-style-type: none"> Procedural programming can run the same code from top to bottom for each understand. The could would be exactly the same as each username is constructed using the same rules. <p>STATEMENTS</p> <ul style="list-style-type: none"> Examples include: The different parts of the text file can be assigned to different variables such as the year, first name and surname), they could be used to set the initial number to 1. Statements are ideal when using a procedural programming language the code will work through each of the statements from the start to the end of the code. <p>PROCEDURES</p> <ul style="list-style-type: none"> When the statements have been setup, the procedures can then be used to carry out actions / calculations. A single procedure may run / use data stored by the individual statements. Examples include: A procedure could be used to combine the different parts into one single username, they could increase the number by one if the username already exists before it is checked again. 	

	<p>BLOCKS</p> <ul style="list-style-type: none"> • A block is a region of the code that treat a group of statements as one whole unit. • Examples include: may use a block to declare variables, loop code that will automatically execute the same block of code that generates different username, loop code to keep creating usernames to ensure its uniqueness. 	(8)
Level	Mark	Description
0	0	
1	1-2	<p>Technical vocabulary is used but is not used appropriately to support arguments in relation to the issues of the question.</p> <p>Issues are identified but chains of reasoning are not made leading to a superficial understanding of the relative importance of issues to the context of the scenario.</p>
2	3-5	<p>Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question.</p> <p>A consideration of relevant issues using logical chains of reasoning but does not reflect upon their relative importance to the context of the scenario.</p>
3	6-8	<p>Fluent and accurate technical vocabulary is used to support arguments that are relevant to the issues of the question.</p> <p>A balanced and wide ranging consideration of relevant issues using coherent and logical chains of reasoning that shows a full awareness of their relative importance to the context of the scenario.</p>

Question Number	Answer	Mark
4a	<p>Award one mark for line number and one additional mark for appropriate expansion up to 3 marks for each.</p> <ul style="list-style-type: none"> • Error 1: Line 59 (1) (When the new 'log' button is pressed) the form is not displayed (1) because the hide mode has been selected. (1) 	

	<ul style="list-style-type: none"> • Error 2: Line 74 (1) (When the 'quit' button is pressed) the application will not close (1) because the run mode has been selected. (1) <p>ADDITIONAL GUIDANCE: Give marks for suitable descriptions even if they have incorrectly identified the line number.</p> <p>Marks can be awarded for descriptions that identify specific code error (1) and how to correct the code (1)</p> <p>Allow Line 58 for Error 1 Allow Line 72 for Error 2</p>	(6)
--	--	-----

Question Number	Answer	Mark
4b	<p>Possible responses:</p> <p>CALCULATING THE LIFT TIME</p> <ul style="list-style-type: none"> • Requirement 1 has been met because line 28 of the programming code calculates how long the lift took by <code>end_time - start_time</code>. • However, there is no check on whether the start / end times have actually been entered and this could lead to inaccurate data being logged & extra faults being logged. <p>CREATING A UNIQUE ID</p> <ul style="list-style-type: none"> • Requirement 2 has been met because line 18 generates a unique identifier by counting how many lines are already in the log text file. • However, if the text file is ever modified and logs / lines are deleted from the text file then this could lead to the same testID being generated more than once. <p>LOGGING ALL TESTS</p> <ul style="list-style-type: none"> • Requirement 3 is mostly, as all require details are written apart from the date that the test was carried out. <p>CREATING A LOG</p> <ul style="list-style-type: none"> • Requirement 4 has been met. Line 30 compares the time taken against the target time. If the time taken is equal to 	

	<p>or less than the target time then a log is created in a text file called 'log.'</p> <p>CREATING A FAULT LOG</p> <ul style="list-style-type: none"> Requirement 5 has not been met. If the time taken is over the target time then a log is recorded in a text file called 'faultlog.' However, the log is also recorded in the 'log' text file. If the time taken is over the target time, then a log should only have been created in the 'faultlog'. 	(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.
1	1-2	<p>Technical vocabulary is used but is not used appropriately to support arguments in relation to the issues of the question.</p> <p>Issues are identified but chains of reasoning are not made leading to a superficial understanding of the relative importance of issues to the context of the scenario.</p> <p>No conclusion is presented or is generic.</p>
2	3-5	<p>Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question.</p> <p>A consideration of relevant issues using logical chains of reasoning but does not reflect upon their relative importance to the context of the scenario.</p> <p>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.</p>
3	6-8	<p>A balanced explanation of both sides of the argument is presented using fluent and accurate technical vocabulary. The points made are discussed in a balanced way that reflects their relative importance to the given scenario. A clear and justified conclusion is presented that reflects a thorough consideration of both sides of the argument.</p> <p>Fluent and accurate technical vocabulary are used to support arguments that are relevant to the issues of the question.</p> <p>A balanced and wide ranging consideration of relevant issues using coherent and logical chains of reasoning that shows a full awareness of their relative importance to the context of the scenario.</p> <p>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.</p>

Question Number	Answer	Mark
4c	<p>Possible responses:</p> <p>Code structure</p> <ul style="list-style-type: none"> • In some languages, the variables have to be declared at the start of the code while in others they can be declared just before they are used. • Not knowing the structure could increase the amount of development time and could increase errors/need for debugging <p>Built-in functions</p> <ul style="list-style-type: none"> • Different languages have different built-in functions / function names. • A programmer may also consider if previously made functions can be imported. These can significantly reduce the workload. • They are also likely to be well tested which therefore will decrease the amount of problems that the programmer will have to debug. • If a function is not available then this will have to be created by the programmer, which therefore increases the chance of errors. <p>Technical support</p> <ul style="list-style-type: none"> • Some languages have more technical support available. • Open source languages may have more books / online materials / forums / code banks for you to use. The programmer is likely to be able to solve problems quicker, which will decrease the development time. • Closed source languages may have less technical documentation to use, which means the programmer will have less sources of help. <p>Debugging facilities:</p> <ul style="list-style-type: none"> • Some languages may have better debugging facilities than others. • Increased debugging facilities may highlight exactly / roughly where errors are making it easier to locate and fix efforts. • Minimal debugging facilities may require the programmers to debug the problems on their own. <p>Memory management</p>	

	<ul style="list-style-type: none"> • Some languages require programmers to manage the computer's memory themselves. • This therefore will increase the workload for the programmers and increase the knowledge needed to be able to implement this. The programmers may be required to add extra comments / documentation into the code to make it maintainable. 	(12)
<p>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*.</p>		
Level	Mark	Descriptor
Level 0	0	No rewardable material.
1	1-4	<p>Technical vocabulary is used but is not used appropriately to support arguments in relation to the issues of the question.</p> <p>Issues are identified but chains of reasoning are not made leading to a superficial understanding of the relative importance of issues to the context of the scenario.</p> <p>Does not link arguments to the given scenario.</p>
2	5-8	<p>Accurate technical vocabulary is used to support arguments but not all arguments are relevant to the issues of the question.</p> <p>A consideration of relevant issues using logical chains of reasoning but does not reflect upon their relative importance to the context of the scenario.</p> <p>Considers the various elements of the question and but does not always link arguments to the given scenario.</p>
3	9-12	<p>Fluent and accurate technical vocabulary is used to support arguments that are relevant to the issues of the question.</p> <p>A balanced consideration of relevant issues using coherent and logical chains of reasoning that shows a full awareness of their relative importance to the context of the scenario.</p> <p>Carefully considers the various elements of the question and Links arguments to the given scenario.</p>

Pearson Education Limited. Registered company number 872828
with its registered office at Edinburgh Gate, Harlow, Essex CM20 2JE

Ofqual



Llywodraeth Cynulliad Cymru
Welsh Assembly Government



Rewarding Learning