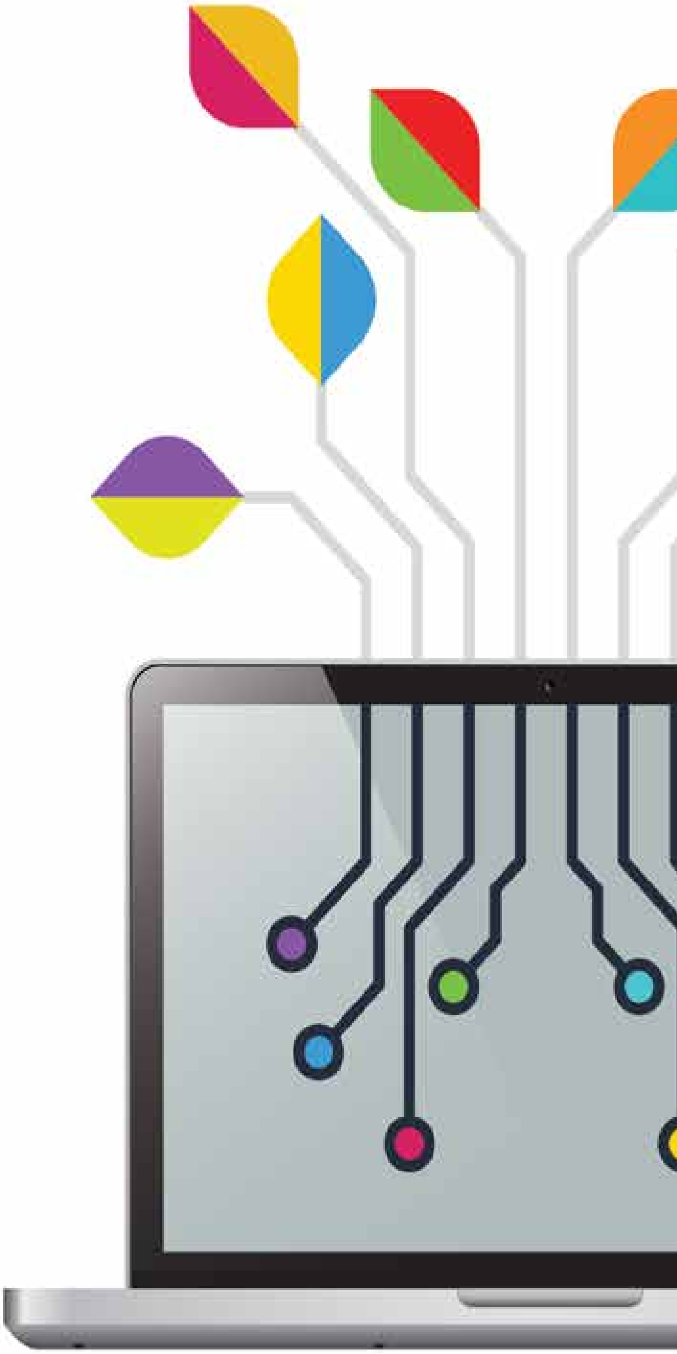
BTEC

Pearson BTEC Level 3 National in Computing

Unit 4: Software Design and Development Project



Sample Assessment Materials (SAMs)

For use with Extended Diploma in Computing

First teaching from September 2016 Issue 1

PEARSON

Edexcel, BTEC and LCCI qualifications

Edexcel, BTEC and LCCI qualifications are awarded by Pearson, the UK’s largest awarding body offering academic and vocational qualifications that are globally recognised and benchmarked.
For further information, please visit our qualification websites at www.edexcel.com, www.btec.co.uk or www.lcci.org.uk. Alternatively, you can get in touch with us using the details on our contact us page at qualifications.pearson.com/contactus

About Pearson

Pearson is the world's leading learning company, with 40,000 employees in more than 70 countries working to help people of all ages to make measurable progress in their lives through learning.
We put the learner at the centre of everything we do, because wherever learning flourishes, so do people. Find out more about how we can help you and your learners at qualifications.pearson.com

References to third-party material made in this specification are made in good faith. We do not endorse, approve or accept responsibility for the content of materials, which may be subject to change, or any opinions expressed therein. (Material may include textbooks, journals, magazines and other publications and websites.)

All information in this document is correct at time of publication.

Pearson BTEC Level 3 Nationals

Write your name here

Surname	Forename

Level 3

Computing

Unit 4: Software Design and Development Project

Part S

Marks

Supervised hours 6

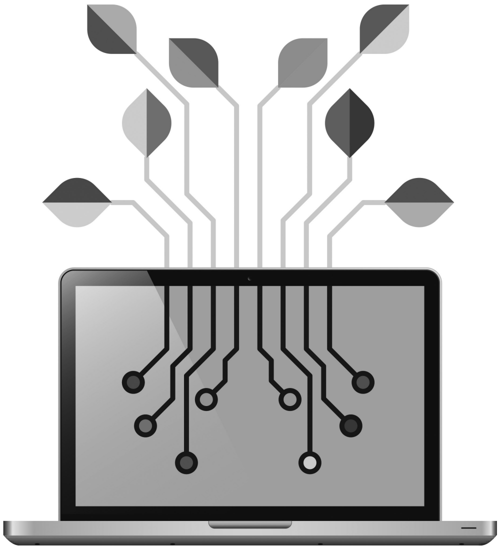
Extended Diploma in Computing
Sample assessment material for first teaching September 2016

Instructions

- This booklet contains material for the completion of the set task under supervised conditions.
- This booklet is specific to each series and this material must only be issued to learners who have been entered to undertake the task in the relevant series.
- This booklet should be kept securely until the start of the 6-hour supervised assessment period.
- This set task should be undertaken during the 1 week assessment period timetabled by Pearson.

Information

- The total mark for this paper is 68.



Paper reference
XXXX/XX
S50113A

1/1/1/1

PEARSON

Instructions to Teachers/Tutors and/or Invigilators

This paper must be read in conjunction with the unit information in the specification and the BTEC Nationals Information for Conducting External Assessments (ICEA) document. See Pearson website for details.

The set task should be carried out under supervised conditions.

Electronic templates for use in activities 3, 4 and 5 will be provided for centres to download for candidate use.

Work should be completed on a computer:

- internet access is not permitted.
- the task must be completed using one of the following programming languages: a C family language or Python 3.4 or later version
- access to a data dictionary for the chosen language is permitted.

During any break, materials must be kept securely.

All learner work must be completed independently and authenticated by the Teacher/Tutor and/or Invigilator before being submitted to Pearson.

Centres are free to arrange the supervised assessment period how they wish provided the 6 hours for producing final outcomes are under the level of control specified, and in accordance with the conduct procedures.

Refer carefully to the instructions in this task booklet and the Information for Conducting External Assessments (ICEA) document to ensure that the assessment is supervised correctly. An authentication statement will be required confirming that learner work has been completed as directed.

Learners must not bring anything into the supervised environment or take anything out without your approval.

Centres are responsible for putting in place appropriate checks to ensure that only permitted material is introduced into the supervised environment.

Maintaining security:

- User areas must only be accessible to the individual learner and to named members of staff.
- Learners can only access their work under supervision.
- Internet access is not permitted.
- Learner work is regularly backed up. Learners will save their work to their folder using the naming instructions indicated in each activity.

- Any work learners produce under supervision must be kept secure.
- Any materials being used by learners must be collected in at the end of each session, stored securely and handed back at the beginning of the next session.

Outcomes for Submission

12 documents will need to be submitted by each learner.

Activity 1: *activity1flowchart*:

- *in their chosen software format*
  and
- *as a PDF.*

Activity 2: *activity2pseudocode*:

- *in their chosen software format*
  and
- *as a PDF.*

Activity 3: *activity3testlog*:

- *in their chosen software format*
  and
- *as a PDF.*

Activity 4: *activity4code*:

- *as a .txt file*
  and
- *as a PDF.*

Activity 4: *activity4testlog*:

- *in their chosen software format*
  and
- *as a PDF.*

Activity 5: *activity5evaluation*:

- *in their chosen software format*
  and
- *as a PDF.*

A fully completed authentication sheet must be completed by each learner.

Instructions for Learners

Read the set task information carefully.

This contains all the information you need to complete each activity within the set task.

Plan your time carefully to allow for the preparation and completion of all the activities.

You will complete the activities within the set task under supervision and your work will be kept securely during any breaks taken.

You must work independently throughout the supervised assessment period and should not share your work with other learners.

You may use a calculator and will have access to a computer. All activities must be completed using a computer.

Internet access is not allowed.

The task must be completed using one of the following programming languages: a C family language or Python 3.4 or later version.

You will have access to a data dictionary for your chosen programming language.

Your teacher/tutor may clarify the wording that appears in this task but cannot provide any guidance in completion of the task.

This task must be completed under supervision in timetabled sessions provided by your centre. It is likely that you will be given more than one timetabled session to complete these tasks.

Outcomes for Submission

You will need to submit 12 documents on completion of the supervised assessment period:

Activity 1: *activity1flowchart*:

- in your chosen software format
  and
- as a PDF in folder for submission.

Activity 2: *activity2pseudocode*:

- in your chosen software format
  and
- as a PDF in folder for submission.

Activity 3: *activity3testlog*:

- in your chosen software format
  and
- as a PDF in folder for submission.

Activity 4: *activity4code*:

- as a .txt file
  and
- as a PDF.

Activity 4: *activity4testlog*:

- in your chosen software format
  and
- as a PDF in folder for submission.

Activity 5: *activity5evaluation*:

- in your chosen software format
  and
- as a PDF in folder for submission.

A fully completed authentication sheet must also be completed; any prepared notes do not need to be submitted with the final outcomes to Pearson.

Set Task Brief

You are asked to use your software design, development, testing and evaluation understanding and skills to produce a program that meets the client's requirements.

A local gym has commissioned you as a software developer to write a program that will assess a gym member's requirement to maintain their current weight accurately.

You need to create a program that will give a gym member information about:

- their current basal metabolic rate (BMR)
- their current body mass index (BMI)
- their target BMI
- the number of kilocalories to maintain their current weight.

You will design, implement and test your program. You will also need to justify and evaluate your decisions.

When designing and developing the solution ensure:

- it is efficient and robust
- it provides the accurate daily intake requirement for a gym member to maintain their body mass index (BMI)
- the BMR calculation is given to 2 decimal places
- the BMI calculation is given to 1 decimal place
- the kilocalorie requirement output is shown rounded to a whole number
- there is a text output to show the member's:
    - current BMR
    - current BMI
    - target BMI.

Information

You are provided with information to use when designing and developing your program:

- the revised Harris-Benedict equation used to calculate Basal Metabolic Rate
- the formula to use to give the recommended daily kilocalorie intake to maintain current weight for men and women
- the formula to calculate BMI
- the standard BMI categories
- current gym membership age, weight and height profile.

Calculating the Basal Metabolic Rate (BMR) using the revised Harris-Benedict equation

Men	BMR = 88.362 + (13.397 x weight in kg) + (4.799 x height in cm) – (5.677 x age in years)
Women	BMR = 447.593 + (9.247 x weight in kg) + (3.098 x height in cm) – (4.330 x age in years)

Calculating the recommended daily kilocalorie intake to maintain current weight

Individual's level of exercise	Calculation of daily intake required (kilocalories)
Little to no exercise	BMR x 1.2
Light exercise (1–3 days per week)	BMR x 1.375
Moderate exercise (3–5 days per week)	BMR x 1.55
Heavy exercise (6–7 days per week)	BMR x 1.725
Very heavy exercise (twice per day, extra heavy workouts)	BMR x 1.9

Calculating BMI

BMI = Weight (kg) / (Height (m) x Height (m))

Standard BMI categories

Underweight = <18.5
Normal weight = 18.5–24.9
Overweight = 25–29.9
Obesity = BMI of 30 or greater.

Ideal gym member BMI = 22.

Current gym membership age, weight and height profile

Membership age range: 14–100 years of age.

Weight range: 30–250 kg.

Height range: 120 – 210 cms.

Set Task

You must complete ALL activities within the set task.

You are reminded that you are to produce your documents using a computer and to save your documents in your folder ready for submission using the formats and naming conventions indicated.

Activity 1

Produce a flow chart, using British Computing Society symbols, to plan the logic and processes for the program.

Save your flow chart in your folder for submission as ***activity1flowchart***:

- *in your chosen software format*
  and
- *as a PDF*.

You are advised to spend approximately 70 minutes on this activity.

Total for Activity 1 = 10 marks

Activity 2

Produce pseudocode that a software developer could use to create the program.

Save your pseudocode in your folder for submission as ***activity2pseudocode***:

- *in your chosen software format*
  and
- *as a PDF*.

You are advised to spend approximately 70 minutes on this activity.

Total for Activity 2 = 10 marks

Activity 3

Produce a test log to plan the testing of your complete program, including test data and expected result.

Show your test planning by completing the test log section of the given 'Test log and evaluation' document.

*Save your completed Test log and evaluation document in your folder for submission as **activity3testlog**:*

- *in your chosen software format*
  and
- *as a PDF.*

You are advised to spend approximately 35 minutes on this activity.

Total for Activity 3 = 6 marks

Activity 4

You are ready to write and test your program. Use your flowchart, pseudocode and test log to help write and test your program.

Write and test your program.

You should:

- write a program that meets the scenario requirements, in a C family or Python V3.4 or later version programming language
- test your solution to ensure that it functions as expected, recording this testing in your 'Test Log and evaluation' document.

Your evidence should include:

- a copy of your code containing annotations/comments
- a copy of your Test log and evaluation document.

Save your code in your folder for submission as ***activity4code*** as a:

- *.text file*
  and
- *PDF.*

Save your Test log and evaluation document in your folder for submission as ***activity4log***:

- *in your chosen software format*
  and
- *as a PDF.*

You are advised to spend approximately 2 hours 20 minutes on this activity.

Total for Activity 4 = 30 marks

Activity 5

Evaluate your program solution.

You should cover:

- how well your solution meets the requirements of the scenario
- the quality and performance of your program
- the choices you made about coding conventions
- the changes you made during the development process.

Write your response in the evaluation section of your 'Test log and evaluation' document.

Save your Test log and evaluation document to your folder for submission as ***activity5evaluation***:

- *in your chosen software format*
  and
- *as a PDF.*

You are advised to spend approximately 45 minutes on this activity.

Total for Activity 5 = 12 marks

END OF TASK **TOTAL FOR TASK = 68 MARKS**

Example of the documents you need to complete.

Document for Activities 3, 4 and 5

Test log and evaluation (add additional rows and extend the evaluation space as required)

Program language the product is to be produced in (tick box for language used):

Python □ C Family □

Test Number	Purpose of test	Test Data	Expected Result	Actual Result	Comments
Software evaluation (Activity 5 response):					

Unit 4: Software Design and Development Project - 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 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 judgment 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 within 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 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.

Mark Scheme: Unit 4 Software Design and Development Project

Assessment focus	Band 0	Band 1	Band 2	Band 3	Max. mark
Activity 1: **Flowchart**	**0**	**1-3**	**4-7**	**8-10**	**10**
	No rewardable material	Use of British Computer Society (BCS) symbols is limited or mostly inaccurate. Flowchart breaks down requirements into component parts that are not relevant or arbitrary. The flowchart shows limited coverage of inputs, outputs and processes using inconsistent or inappropriate naming conventions Links between component parts are incomplete or inappropriate with limited procedures for handling unexpected events	British Computer Society (BCS) symbols used throughout with some inaccuracies. Flowchart breaks down requirements into component parts that are relevant, but lack detail. The flowchart shows coverage of most inputs, outputs and processes using some naming conventions appropriate to the scenario consistently. Links between component parts are complete, but may be inefficient with accurate procedures for handling some unexpected events	British Computer Society (BCS) flowchart symbols used accurately throughout. Flowchart breaks down requirements into component parts that are detailed and relevant. The flowchart shows full coverage of inputs, outputs and processes using naming conventions appropriate to the scenario consistently. Links between component parts are complete and efficient with accurate and robust procedures for handling unexpected events	

Assessment focus	Sub Task	Band 0	Band 1	Band 2	Band 3	Max. mark
Activity 2: **Producing pseudocode and testing**		**0**	**1-3**	**4-7**	**8-10**	**10**
	Pseudo code	No rewardable material	Structure of Pseudocode shows some use of appropriate hierarchies and indentation but clarity and/or readability is limited. The sequence of processes is partially incomplete or incorrect which would result in incorrect outcomes. Pseudocode uses some inappropriate and/or inconsistent naming conventions. Pseudocode includes imprecise use of logical operations, which would lead to a solution that is inaccurate and/or incomplete.	Structure of pseudocode makes use of appropriate hierarchies and indentation to provide some clarity and readability but these are not always consistent. The sequence of processes is complete but the suggested solution is inefficient and/or may result in minor errors in outcomes. Pseudocode uses appropriate naming conventions but there may be some inconsistencies. Pseudocode includes precise use of most logical operations, which would lead to a complete solution with some inaccuracies.	Structure of pseudocode shows appropriate and consistent use of hierarchy and indentation, providing clarity and readability The sequence of processes is complete and efficient and would result in the correct outcome(s). Pseudocode uses appropriate and consistent naming conventions. Pseudocode includes precise use of logical operations, which would lead to a complete and accurate solution.	

Note: the Assessment Grid for activity 3 (Test Plan) is given after the activity 4 Program Assessment Grid, so that all testing activities are on one page.

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4	Max. mark
Activity 4	**0**	**1-6**	**7 - 12**	**13-18**	**19 -24**	**24**
Program	No rewardable material	Limited use of accurate syntax and indentation appropriate for the chosen language. Organisation has structure that lacks logic and commenting is vague, making maintenance of the code by a third party difficult. Code is inefficient; uses limited appropriate and accurate programming conventions. Uses imprecise logical operations to create a program which may not function or compile and/or may have major errors that prevent the program from meeting the given criteria. Program outputs may contain inaccuracies and/or provide limited information so a novice user would experience difficulty in using the program. Program uses minimal validation and checking procedures resulting in a program with limited capacity to reduce errors or handle unexpected events**.**	Program uses mostly accurate syntax and indentation throughout, appropriate to the chosen language. Organisation has some logical structure and some of the commenting of the code is informative but not always clear, allowing it to be maintained by a third party with minor difficulties. Code is efficient in some places; uses mostly appropriate programming conventions, with minor inaccuracies. Uses logical operations with some precision to create a functional program that meets most of the given criteria with minimal errors. Program outputs are accurate and mostly informative so a novice user would experience minor difficulties in using the program. Program uses some accurate validation and checking procedures, resulting in a program that minimises the most common errors and handles some unexpected events.	Program uses mostly accurate syntax and indentation throughout, appropriate to the chosen language. Organisation has logical structure and commenting is informative, but not always clear, allowing for the code to be maintained by a third party. Code is efficient; uses appropriate and accurate programming conventions throughout. Uses logical operations with some precision to create a functional program that meets the given criteria with minimal errors. Program outputs are accurate and mostly informative allowing a novice user to use the program. Program uses accurate validation and checking procedures, resulting in a program that minimises errors and handles unexpected events**.**	Program uses accurate syntax and indentation throughout, appropriate to the chosen language. Organisation has logical structure and commenting is consistently clear and informative, allowing for the code to be easily maintained by a third party. Code is highly efficient and optimised; uses appropriate and accurate programming conventions throughout. Uses precise logical operations throughout to create a fully functional, error-free program that meets the given criteria. Program outputs are accurate and informative allowing a novice user to easily use the program. Program uses accurate validation and checking procedures throughout, resulting in a robust program that minimises errors and handles unexpected events.	

Assessment Focus	Sub Task	Band 0	Band 1	Band 2	Band 3	Max Mark
		0	1-2	3-4	5-6	6
Testing	Test Plan Activity 3	No rewardable material	Test plan is too narrow to confirm a working solution including limited normal, abnormal and/or extreme data. Expected results are generic or mostly inaccurate based on identified test data.	Test plan is adequate to confirm a working solution, including some normal, abnormal and extreme data. Expected results are and accurate based on identified test data, but may lack detail.	Test plan is thorough, including a range of normal, abnormal and extreme data. Expected results are specific and accurate based on identified test data.	
		0	1-2	3-4	5-6	6
	Testing Activity 4	No rewardable material	Testing shows evidence of a limited or linear development process, with minimal identification and resolution of errors. Comments show a limited understanding of errors that were found, and how they were fixed.	Testing shows evidence of an iterative development process that identifies and resolves some errors, but problems may persist. Comments show partial understanding of errors that were found, and how they were fixed.	Testing shows evidence of an iterative development process that identifies and resolves errors and improves efficiency. Comments show a clear and detailed understanding of errors that were found, and how they were fixed.	

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4	Max. mark
Activity 5 Evaluation	**0**	**1-3**	**4-6**	**7-9**	**10- 12**	
	No rewardable material	Superficial understanding of relevant technical concepts with some inaccuracies. Limited or unsupported justification of changes made during the development process. Limited or unsupported justification of coding conventions selected. Limited links between aspects of the solution and the requirements of the scenario. Limited or unsupported judgments about the quality and performance of the program. Technical vocabulary is used but it is not used appropriately to support arguments.	Some accurate and relevant understanding of technical concepts. Some valid justification, which may lack support, of changes made during the development process. Some valid justification, which may lack support, of coding conventions selected. Some logical links between aspects of the solution and the requirements of the scenario but may lack clarity. Some valid judgments which may lack support about the quality and performance of the program. Mostly accurate technical vocabulary is used to support arguments.	Mostly accurate and detailed understanding of relevant technical concepts. A valid and mostly supported justification of changes made during the development process. A valid and mostly supported justification of coding conventions selected. Makes some logical coherent links between aspects of the solution and the requirements of the scenario. Makes valid and mostly supported judgments about the quality and performance of the program. Accurate technical vocabulary is used to support arguments.	Accurate and detailed understanding of relevant technical concepts throughout. A valid and fully supported justification of changes made during the development process. A valid and fully supported justification of coding conventions selected. Makes logical coherent links between aspects of the solution and the requirements of the scenario throughout. Makes valid and fully supported judgments about the quality and performance of the program. Fluent and accurate technical vocabulary is used to support arguments.	**12**

Total mark = 68

For more information about Edexcel, BTEC or LCCI qualifications
visit qualifications.pearson.com

Pearson Education Limited. Registered in England and Wales No. 872828
Registered Office: 80 Strand, London WC2R 0RL
VAT Reg No GB 278 537121