

The logo consists of the letters 'N5' in a large, bold, white font, set against a solid purple square background.

National 5  
Coursework  
Assessment Task



---

# National 5 Computing Science Assignment Assessment task

This document provides information for teachers and lecturers about the coursework component of this course in terms of the skills, knowledge and understanding that are assessed. It must be read in conjunction with the course specification.

**Specimen – valid from session 2023-24 and until further notice.**

This edition: August 2023 (version 1.0)

© Scottish Qualifications Authority 2023

# Contents

Introduction	1
Instructions for teachers and lecturers	2
<b>Marking instructions</b>	<b>7</b>
Instructions for candidates	15

# Introduction

This document contains instructions for teachers and lecturers, and instructions for candidates for the National 5 Computing Science assignment. You must read it in conjunction with the course specification.

This assignment has 40 marks out of a total of 120 marks available for the course assessment.

This is one of two course assessment components. The other component is a question paper.

# Instructions for teachers and lecturers

This is a specimen assessment task for National 5 Computing Science.

SQA publishes a new assessment task on the secure website each academic year. The task is valid for that year only. Once complete, you must send the assignment responses to SQA to be marked.

You must conduct the assignment under a high degree of supervision and control. This means:

- ◆ candidates must be supervised throughout the session(s)
- ◆ candidates must not have access to email or mobile phones
- ◆ candidates must complete their work independently – no group work is permitted
- ◆ candidates must not interact with each other
- ◆ with no interruption for targeted learning and teaching
- ◆ in a classroom environment

You can use any integrated development environments (IDE) that enables candidates to generate evidence – this includes online IDEs. However, the IDE must have a facility that prevents candidates accessing their files and tasks outside the supervised classroom environment.

## Time

Candidates have 6 hours to carry out the assignment, starting at an appropriate point in the course, after all content has been delivered. It is not anticipated that this is a continuous 6-hour session, although it can be, but conducted over several shorter sessions. This is at your discretion.

You have a responsibility to manage candidates' work, distributing it at the beginning and collecting it in at the end of each session, and storing it securely in between. This activity does not count towards the total time permitted for candidates to complete the assignment.

Candidates are prompted to print their work at appropriate stages of the tasks. They can print on an ongoing basis or save their work and print it later. Whatever approach they take, time for printing is not part of the 6 hours permitted for the assignment.

## Resources

Each candidate must have access to a computer system with a high-level (textual) programming language and **either**:

- ◆ a database application or software that can create, edit and run SQL
- ◆ software that can create, edit and run HTML and CSS

This is an open-book assessment. Candidates can access resources such as programming manuals, class notes, textbooks and programs they have written throughout the course. These may be online resources.

You must not create learning and teaching tasks that make use of constructs required in the assessment task, **with the specific purpose of developing a solution that candidates can access during the assignment.**

You can provide candidates with templates, however these templates must only contain general starter code used in learning and teaching (for example, a web page that contains the HTML, title and body elements) – templates must not be tailored to this year's task.

There may be instances where restriction of network use is prohibited (for example, a local authority-managed network with specific limitations). However, it remains your professional responsibility to make every effort to meet the assessment conditions.

## Reasonable assistance

The assignment consists of three independent tasks. They are designed in a way that does not require you to provide support to candidates, other than to ensure that they have access to the necessary resources. Candidates can complete the tasks in any order.

Once the assignment is complete, you must not return it to the candidate for further work to improve their mark. You must not provide feedback to candidates or offer an opinion on the perceived quality or completeness of the assignment response, at any stage.

You can provide reasonable assistance to support candidates with the following aspects of their assignments:

- ◆ printing, collating and labelling their evidence to ensure it is in the format specified by SQA
- ◆ ensuring candidates have all the materials and equipment required to complete the assignment – this includes any files provided by SQA
- ◆ ensuring candidates understand the conditions of assessment and any administrative arrangements around the submission and storage of evidence, and the provision of files
- ◆ technical support

## Evidence

All candidate evidence (whether created manually or electronically) must be submitted to SQA in a paper-based format. The evidence checklist details all evidence to be gathered. You can use it to ensure you submit all evidence to SQA.

You should advise candidates that evidence, especially code, must be clear and legible. This is particularly important when pasting screenshots into a document.

There is no need for evidence to be printed single sided or in colour.

## Alteration or adaptation

The tasks are in PDF and Word formats. Each task is available as a separate file from the secure site. Word files allow candidates to word process their responses to parts of the task.

You must not adapt the assignment in any way that changes the instructions to the candidate and/or the nature and content of the tasks. However, you can make changes to font size, type and colour and to the size of diagrams for candidates with different assessment needs, for example, visual impairment.

If you are concerned that any particular adaptation changes the nature and/or the content of the task, please contact our Assessment Arrangements Team for advice as soon possible at [aarequests@sqa.org.uk](mailto:aarequests@sqa.org.uk).

## Submission

Each page for submission has the number of the assignment task that it refers to, for example 1a, and contains space for candidates to complete their name and candidate number. Any other pages submitted, for example, prints of program listings or screenshots, must have this information added to them.

## Specific instructions for teachers and lecturers: specimen assignment

All candidates must complete task 1 (software design and development) and **either** task 2 (database design and development) **or** task 3 (web design and development).

It is at your discretion how you approach this optionality in assessment. The task your candidates complete might be pre-determined by your progress through the course, or you may be able to let candidates choose which task to complete.

You must follow these specific instructions and ensure that candidates are aware of what you will give them at each stage in the assessment.

Print each task on single-sided paper, where applicable:

- ◆ this allows candidates to refer to information on other pages
- ◆ this helps you manage tasks that are split into more than one part

**Task 1 – part A** requires candidates to complete the analysis of a problem. They must submit their evidence to you before you issue part B.

**Task 1 – part B** is a separate section. This ensures that candidates are not able to access part A and change their responses. They must submit their evidence to you before you issue part C.

**Task 1 – part C** is a separate section. This ensures that candidates are not able to access part B and change their responses.

**Task 2 – part A** requires candidates to analyse a database problem. They must submit their evidence to you before you issue part B.

**Task 2 – part B** is a separate section. This ensures that candidates do not access part A and change their responses. They must submit their evidence to you before you issue part C.

**Task 2 – part C** is a separate section. This ensures that candidates do not access part B and change their responses.

A Microsoft Access file (football.accdb) is provided for candidates to use in part C. If your centre uses a different database management system, you can create the relational database for using the CSV files or the text files provided.

If using the CSV files, you should set up all tables, fields and validation shown in the data dictionaries below. Some validation has been intentionally left blank to be completed by candidates as part of the task. Referential integrity should also be enforced.

The text files contain SQL create and insert statements for each table. If you use the text files, you must add validation (shown in the data dictionaries below), appropriate for your

version of SQL. Some validation has been intentionally left blank, to be completed by candidates as part of the task. Referential integrity should also be enforced.

Entity: Club					
Attribute name	Key	Type	Size	Required	Validation
clubName	PK	text	20	Y	length <= 20
street		text	40	Y	length <= 40
postcode		text	8	Y	length <= 8
formed		date		Y	
league		number		Y	Restricted choice: 1, 2, 3

Entity: Player					
Attribute name	Key	Type	Size	Required	Validation
registration	PK	number		Y	Range: >= 100000 and <= 999999
clubName	FK	text	20	Y	Existing clubName from Club table
forename		text	20	Y	
surname		text	30	Y	
mobileNo		text	12	Y	length = 12
dateOfBirth		date		Y	
position		text	10	Y	Restricted choice: Striker, Midfielder, Defender, Goalkeeper
shirtNumber		number		Y	

**Note:** the date in the SQL statement in task 2f(i) is **not** the error that candidates are asked to identify. If you use an SQL server, the date in the INSERT statement must be edited to the appropriate format.

**Task 3** – requires candidates to edit a website.

A folder named ‘Web files’ is provided. This contains the CSS, HTML and media files candidates need to complete this task. These files must not be renamed and they must remain in the folders provided. However, the case of suffixes may be changed if the environment you work in requires them to be lower or upper case.

Candidates do not need to print completed web pages in colour.

# Marking instructions

In line with SQA's normal practice, the following marking instructions for the National 5 Computing Science assignment are addressed to the marker. They will also be helpful for those preparing candidates for course assessment.

Candidates' evidence is submitted to SQA for external marking.

## General marking principles

Always apply these general principles. Use them in conjunction with the specific marking instructions, which identify the key features required in candidates' responses.

- a Always use positive marking. This means candidates accumulate marks for the demonstration of relevant skills, knowledge and understanding; marks are not deducted for errors or omissions.
- b If a candidate response does not seem to be covered by either the principles or detailed/specific marking instructions, and you are uncertain how to assess it, you must seek guidance from your team leader.
- c Award marks regardless of spelling, as long as the meaning is unambiguous and does not result in a syntax error in implemented code.
- d For design and implementation tasks, a sample response may be shown in the detailed marking instructions. This will not be the only valid response. You must use the detailed marking instructions and additional guidance to ensure that you consider alternative approaches and nuances of different programming languages. If in doubt, you should refer to your team leader.
- e A correct response can be negated if the candidate includes an extra, incorrect response which demonstrates they do not know the correct answer. For example, in a 'state' question where the only correct answer is 'white', if the candidate answers 'white orange', you should not award the mark.
- f If a candidate puts a score through their entire response to a question and makes a further attempt, you should only mark the further attempt. If no further attempt is made and the original is legible, you should mark the original response.
- g In the marking instructions, if a word is underlined then it is essential; if a word is in brackets() then it is not essential. Words separated by / are alternatives.

# Specific marking instructions

## Task 1 – software design and development

Task	Expected response	Max mark	Additional guidance
1a	<ul style="list-style-type: none"><li>◆ number of words (in sentence) to be entered</li><li>◆ each word in the sentence</li></ul>	2	For the first bullet point, accept input that would end a conditional loop, as the user could be asked if they wish to enter another word.
1b	Design showing: <ul style="list-style-type: none"><li>◆ input of words in loop</li><li>◆ running total of word lengths in loop</li><li>◆ average calculated outside loop using “valid number of words” shown in the design</li></ul>	3	Variables names used in design will vary.

Task	Expected response	Max mark	Additional guidance	
1c	<ul style="list-style-type: none"> <li>◆ array of strings used to store individual words</li> <li>◆ total (number of characters) initialised to 0</li> </ul>	2		
	Input validation	Conditional loop used	1	
		Loop conditions correct	1	Conditions in loop may be either pre or post condition: <ul style="list-style-type: none"> <li>◆ Until <math>\geq 1</math> and <math>\leq 20</math></li> <li>◆ While <math>&lt; 1</math> or <math>&gt; 20</math></li> </ul> or any other valid answer.
		Input inside loop	1	If no validation is attempted, one of the four input validation marks may be awarded for a single input.
		Error message inside validation loop	1	Must be more than duplication of original input message.
	Loops	<ul style="list-style-type: none"> <li>◆ two fixed loops (input and output) which match design</li> <li>◆ correct number of iterations (number of words)</li> </ul>	2	Award 1 mark if a conditional loop and counter is used for input.
	Use of length function to calculate number of characters		1	
	Running total used to calculate total number of characters		1	
	Average word length calculated correctly		1	Total number of characters divided by number of words.
	Display words on individual lines		1	
If statement	If conditions both correct: <ul style="list-style-type: none"> <li>◆ <math>&lt; 5</math></li> <li>◆ <math>\geq 5</math> and <math>\leq 7</math></li> </ul>	1		

Task	Expected response	Max mark	Additional guidance
	If structure matches design	1	Award mark for use of else if/nested if statements.  Do not award mark for use of three separate if statements.
	Display reading age message	1	
1d(i)	Evidence of testing the supplied test data.	1	The average word length is 8.4 Long words - suitable for senior readers  Award mark for the correct message being displayed, even if typing errors made during input of words results in a different average.
1d(ii)	<ul style="list-style-type: none"> <li>◆ a four-word sentence that has an average word length of less than 5</li> <li>◆ a three-word sentence that has an average word length between 5 and 7</li> </ul>	2	
1e	<p>Evaluation of the following:</p> <ul style="list-style-type: none"> <li>◆ one efficiency or one inefficiency in own program code</li> <li>◆ comment on one aspect of readability in the own code</li> </ul>	2	<p>Efficiency examples could include comparison of:</p> <ul style="list-style-type: none"> <li>◆ array vs multiple variables</li> <li>◆ nested ifs vs individual ifs</li> <li>◆ use of a loop vs replication of code</li> </ul> <p>Evaluation of readability must contain an element of evaluation rather than simple statements of terms. For example “I have used white space to highlight structures in my program” <b>not</b> “I have used white space”. The candidate’s code must also show evidence of this for a mark to be awarded.</p>

## Task 2 – database design and development

Task	Expected response	Max mark	Additional guidance
2a	<p>The remaining attributes of club:</p> <ul style="list-style-type: none"> <li>◆ club name</li> <li>◆ date formed</li> <li>◆ league</li> </ul> <p>The remaining attributes of player:</p> <ul style="list-style-type: none"> <li>◆ registration</li> <li>◆ mobile number</li> <li>◆ date of birth</li> <li>◆ position</li> <li>◆ shirt number</li> <li>◆ (club name)</li> </ul>	2	<p>Award 1 mark for identifying the remaining attributes of club.</p> <p>Award 1 mark for identifying the remaining attributes of player.</p> <p>Names given to club and player details could differ from bullet list.</p> <p>Ignore club name if included in the player details, as this is how the tables will be implemented.</p>
2b	<ul style="list-style-type: none"> <li>◆ PK – club name in Club entity</li> <li>◆ PK – registration in Player entity</li> <li>◆ FK – club name in Player entity</li> </ul>	2	<p>Award 1 mark for identifying both PKs.</p> <p>Award 1 mark for identifying the FK.</p>
2c	<p>shirtNumber validation — range <math>\geq 1</math> AND <math>\leq 25</math></p>	1	<p>Evidence must show that validation has been added to the database.</p>

Task	Expected response	Max mark	Additional guidance
2d(i)	<ul style="list-style-type: none"> <li>◆ UPDATE Player</li> <li>◆ SET clubName= Dundee North", shirtNumber = 24</li> <li>◆ WHERE registration = 814209</li> </ul>	<b>3</b>	Do not accept use of the player forename and surname for the WHERE criteria, as there are two players of the same name in the database.
2d(ii)	<ul style="list-style-type: none"> <li>◆ SELECT forename, surname, mobileNo, Club.clubName FROM Club, Player</li> <li>◆ WHERE position = "Goalkeeper"</li> <li>◆ AND league = 1</li> <li>◆ AND Club.clubName = Player.clubName</li> </ul>	<b>4</b>	<p>Do not award first bullet if extra fields are included in SELECT clause.</p> <p>ClubName in SELECT could also be the FK, Player.clubName</p>
<p>Do not award any marks if the candidate's SQL is created by an application.</p> <p>MS Access design view example:</p> <pre>SELECT Player.forename, Player.surname, Player.mobileNo, Club.clubName FROM Club, Player WHERE ((Club.clubName)=[Player].[clubName])AND((Player.[position])= "Goalkeeper")AND ((Club.[league])=1</pre>			
2e	The SQL statement finds all players with the required shirtNumber rather than just the strikers.	<b>1</b>	Accept code-based answer if candidate states that: position = 'striker' is missing from the WHERE clause.
2f(i)	Unknown	<b>1</b>	
2f(ii)	<ul style="list-style-type: none"> <li>◆ Referential integrity has been implemented.</li> </ul> <p style="text-align: center;"><b>or</b></p> <ul style="list-style-type: none"> <li>◆ The database should prevent a player being added to the database if they do not have a valid club.</li> </ul>	<b>1</b>	

### Task 3 – web design and development

Task	Expected response	Max mark	Additional guidance
3a	<p>The user would like to see:</p> <ul style="list-style-type: none"> <li>◆ where the club meets</li> <li>◆ when the club meets</li> <li>◆ photographs of each leader</li> <li>◆ a paragraph about each leader</li> <li>◆ a list of activities for this week</li> <li>◆ a list of activities for next week</li> <li>◆ some additional information on events</li> <li>◆ the cost of each activity</li> </ul>	2	Award 1 mark for each bullet. Maximum 2 marks.
3b	<ul style="list-style-type: none"> <li>◆ All page content included: <ul style="list-style-type: none"> <li>— main heading</li> <li>— a link to external webpage (video)</li> <li>— a graphic called “armsling.jpg”</li> <li>— a paragraph of text</li> <li>— a numbered list</li> </ul> </li> <li>◆ Page content in a logical order with: <ul style="list-style-type: none"> <li>— heading at top</li> <li>— paragraph before list</li> </ul> </li> </ul>	2	
3c	<ul style="list-style-type: none"> <li>◆ heading, paragraph and graphic added</li> <li>◆ numbered (ol) list added</li> <li>◆ link to external webpage</li> <li>◆ layout matches candidates design</li> </ul>	4	Marks should be awarded from code, not the printout of webpage as viewed in a browser.

Task	Expected response	Max mark	Additional guidance
3d(i)	The audio element required to play 'mixExample.mp3' has been added correctly.	1	Marks should be awarded from code, not the printout of webpage as viewed in a browser.
3d(ii)	<ul style="list-style-type: none"> <li>◆ Page Background - lightblue (#ADD8E6)</li> <li>◆ Middle &lt;div&gt; element also - lightblue (#ADD8E6)</li> <li>◆ &lt;h2&gt; styled: <ul style="list-style-type: none"> <li>— text size 14</li> <li>— font Arial</li> </ul> </li> </ul>	3	Marks should be awarded from CSS and HTML code, not the printout of webpage as viewed in a browser.
3e	Some of the text is not visible following the change of background to lightblue.	1	<p>Any correct answer relating to candidate's website, evidenced by their code.</p> <p><b>Note:</b> Accept answer of 'no errors found' if candidate indicates that they corrected the code.</p>
3f	<ul style="list-style-type: none"> <li>◆ website is not fit for purpose.</li> <li>◆ one reason: <ul style="list-style-type: none"> <li>— no photographs of each leader</li> <li>— no information about when the club meets</li> <li>— no paragraph about each leader</li> </ul> </li> </ul>	2	

[END OF MARKING INSTRUCTIONS]

# Instructions for candidates

This assessment applies to the assignment for National 5 Computing Science.

This assignment has 40 marks out of a total of 120 marks available for the course assessment.

It assesses the following skills, knowledge and understanding:

- ◆ applying aspects of computational thinking across a range of contexts
- ◆ analysing problems within computing science across a range of contemporary contexts
- ◆ designing, implementing, testing and evaluating digital solutions (including computer programs) to problems across a range of contemporary contexts
- ◆ demonstrating skills in computer programming
- ◆ applying computing science concepts and techniques to create solutions across a range of contexts

Your teacher or lecturer will let you know if there are any specific conditions for doing this assessment.

In this assessment, you have to complete two short practical tasks.

You must complete task 1 (software design and development) and **either** task 2 (database design and development) **or** task 3 (web design and development).

You may complete the tasks in any order.

## Advice on how to plan your time

You have 6 hours to complete the assignment. Marks are allocated as follows:

- |  |          |                |
|--|----------|----------------|
| ◆ Task 1 – software design and development | 25 marks | (63% of total) |
| <b>AND EITHER</b>                          |          |                |
| ◆ Task 2 – database design and development | 15 marks | (37% of total) |
| <b>OR</b>                                  |          |                |
| ◆ Task 3 – web design and development      | 15 marks | (37% of total) |

You can use this split as a guide when planning your time for each of the two tasks.

## Advice on gathering evidence

As you complete each task, you must gather evidence as instructed.

Your evidence, especially code, must be clear and legible. This is particularly important when you paste screenshots into a document. You can print code from the software environment or copy and paste this into other packages such as notepad or Word.

Use the evidence checklist provided to make sure you submit everything necessary at the end of the assignment. Ensure your name and candidate number is included on all your evidence.

Evidence may take the form of printouts of code, screenshots, typed answers, hand-written answers or drawings of diagrams and designs.

## Advice on assistance

This is an open-book assessment. This means that you can use:

- ◆ any classroom resource as a form of reference (for example programming manuals, class notes, and textbooks) – these may be online resources
- ◆ any files you have previously created throughout the course

The tasks are designed so you can complete them independently, without any support from your teacher or lecturer. This means that you:

- ◆ cannot ask how to complete any of the tasks
- ◆ cannot access any assignment files outside the classroom

# Computing Science assessment task: evidence checklist

You should complete the checklist for task 1 and either task 2 or task 3.

## Task 1 – software design and development

Evidence		Tick
1a	Completed task 1 sheet identifying the missing inputs	
1b	Completed task 1 sheet showing the expanded design for ‘calculate average length of words in sentence’	
1c	Printout of your program code	
1d (i)	Printout evidence of the test run showing inputs and outputs	
1d (ii)	Completed task 1 sheet showing the completed test table	
1e	Completed task 1 sheet with evaluation	

## Task 2 – database design and development

Evidence		Tick
2a	Completed task 2 sheet showing the analysis of inputs	
2b	Completed task 2 sheet showing the primary and foreign keys in the data dictionary	
2c	Printout or screenshot showing correct validation implemented for shirtNumber field	
2d (i)	Printout of SQL statement to change player’s information	
	Printout of the updated ‘Player’ table	
2d(ii)	Printout of SQL statement to display a list of suitable players	
	Printout of the output from the SQL statement	
2e	Completed task 2 sheet explaining why the output is not correct	
2f (i)	Completed task 2 sheet identifying the value that produces an error	
2f (ii)	Completed task 2 sheet explaining why this error is expected	

## Task 3 – web design and development

Evidence	Tick
3a	Completed task 3 sheet identifying two end-user requirements
3b	Completed task 3 sheet showing a wireframe design for the ‘First Aid’ page
3c 3d (i) 3d (ii)	Printout of HTML and CSS code: <ul style="list-style-type: none"><li>◆ firstAid.html</li><li>◆ soundMixing.html</li><li>◆ home.html</li><li>◆ styles.css</li></ul>
3e	Completed task sheet 3 discussing the results of testing the website
3f	Completed task sheet 3 with evaluation of fitness for purpose

Please follow the steps below before handing your evidence to your teacher or lecturer:

- ◆ Check you have completed all parts of task 1 and either task 2 or task 3.
- ◆ Label any printouts and screenshots with the task number (for example 1c, 2a).
- ◆ Clearly display your name and candidate number on each printout.

## Task 1: software design and development (part A)

The measurement of how easy a sentence is to read is called the reading age. One method of calculating this is to use the average length of the words in the sentence.

So the longer the average word length, the higher the reading age.

### Program analysis

A program is required to calculate the approximate reading age (junior, teen or senior) of a single sentence with a maximum of 20 words. The user will enter each word of the sentence, one at a time. The program will then calculate the average word length and display an appropriate message.

1a Complete the analysis below by identifying the missing inputs.

(2 marks)

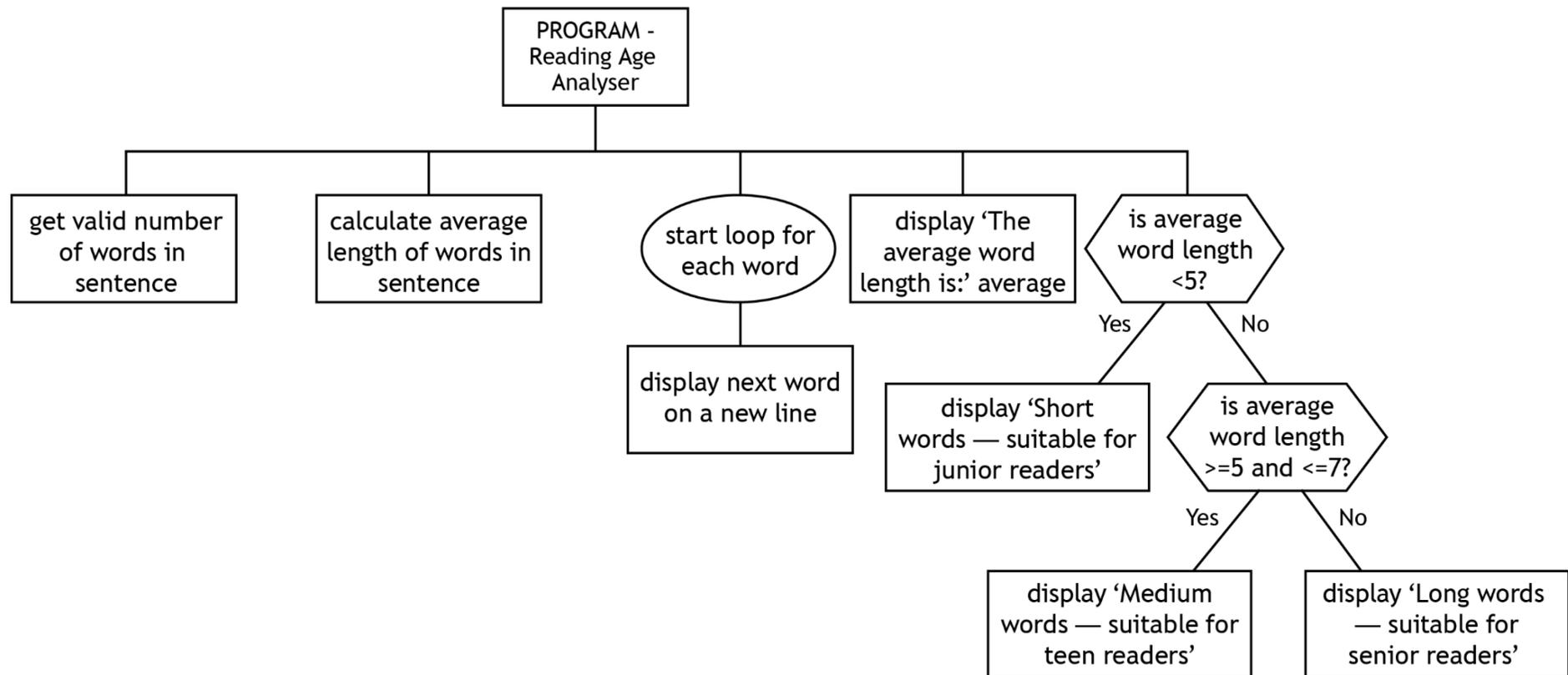
<b>Inputs</b>  ◆  ◆
<b>Processes</b> ◆ calculate the average word length (total number of characters divided by number of words) ◆ decide which message to display
<b>Outputs</b> ◆ each word in the sentence displayed on a new line ◆ one appropriate message from: — Short words – suitable for junior readers — Medium words – suitable for teen readers — Long words – suitable for senior readers

- ◆ Check your answers carefully, as you cannot return to part A after you hand it in.
- ◆ When you are ready, hand part A to your teacher or lecturer and collect part B.

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

# Task 1: software design and development (part B)

## Program design (structure diagram)

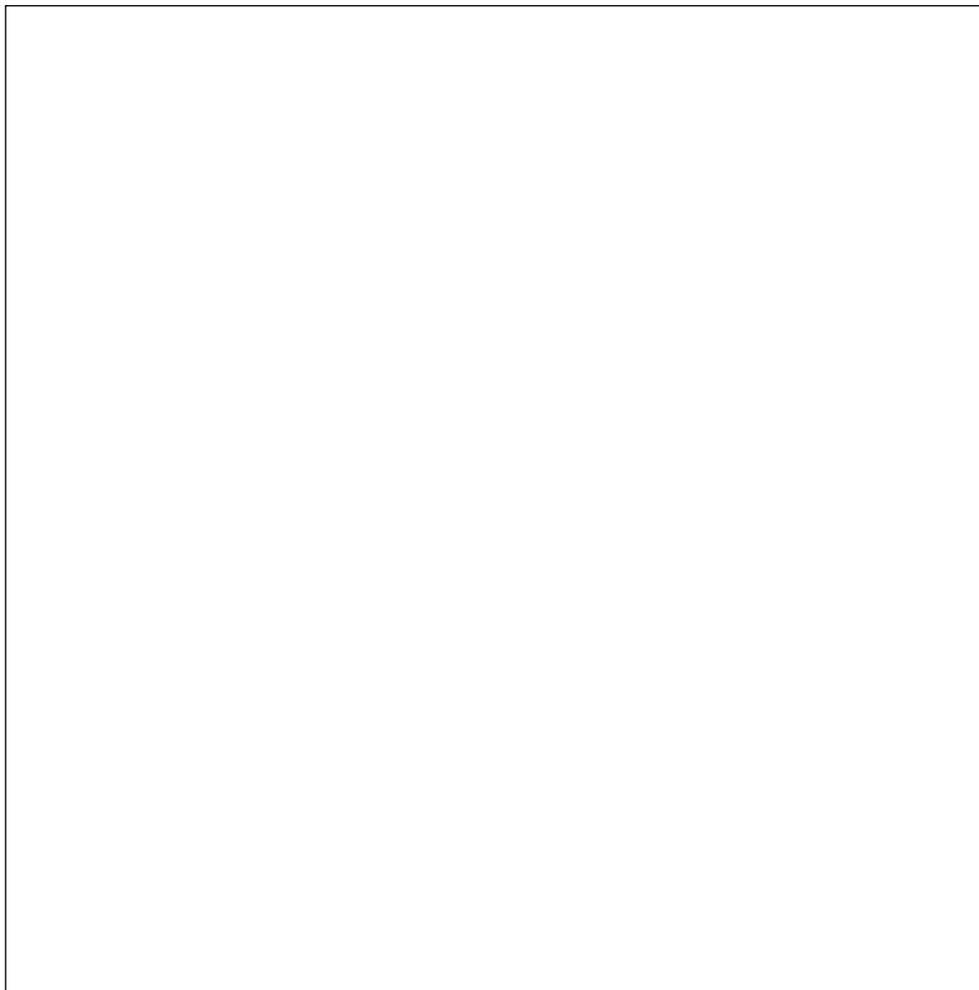


1b The structure diagram contains the following process:

calculate average  
length of words in  
sentence

Using the information provided in the program analysis, expand the design to show how this process could be carried out. You can use a flowchart, structure diagram or pseudocode design.

(3 marks)

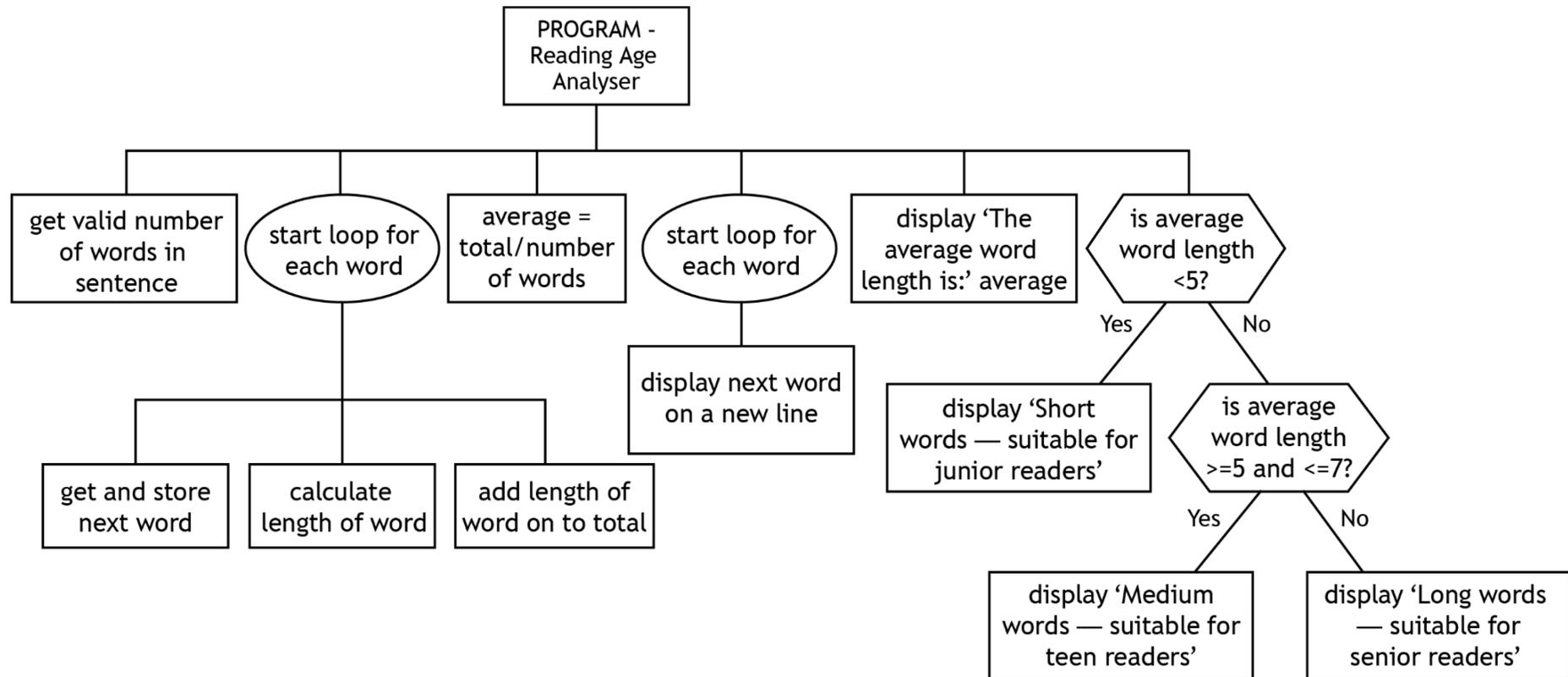


- ◆ Check your answers carefully, as you cannot return to part B after you hand it in.
- ◆ When you are ready, hand part B to your teacher or lecturer and collect part C.

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

# Task 1: software design and development (part C)

## Program design (completed structure diagram)



1c Using the program analysis and the design, implement the program in a language of your choice.

Ensure the program matches the completed structure diagram.

Print evidence of your program code.

**(15 marks)**

1d(i) You should test your program to ensure it produces the expected output.

Use the following data to check that the message ‘Long words – suitable for senior readers’ is displayed:

Number of words: 5  
Words in sentence: distressed  
tourists  
wandering  
around  
aimlessly

Run your program to show that it produces the correct message.

Print evidence of the test run showing inputs and outputs.

**(1 mark)**

(ii) Additional test data is required to check that the other two messages are also displayed correctly.

Complete the test table below with data that could be used to produce the other two messages.

Test data	Expected results
Number of words: 4 Sentence:	‘Short words – suitable for junior readers’ is displayed.
Number of words: 3 Sentence:	‘Medium words – suitable for teen readers’ is displayed.

**(2 marks)**

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

1e With reference to your code, evaluate your program by commenting on the following:

Efficiency of your program code	<b>(1 mark)</b>
Readability of your program code	<b>(1 mark)</b>

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

## Task 2: database design and development (part A)

The Scottish Amateur Women's Football Association keeps details of clubs and players who play in their leagues.

Each club plays matches in one of three football leagues. Each club's football pitch is identified by an address (street and postcode). The date each club was first formed is recorded. The Association's rules state that no club is allowed to have the same name as another.

Players are given a unique registration number. The Association records their forename, surname, date of birth and the club they play for. Players are required to supply a mobile phone number, so they can be contacted about match fixtures or cancellations. Each club is required to inform the Association of each player's shirt number and their preferred playing position (Striker, Midfielder, Defender or Goalkeeper).

Players cannot be registered unless they are a member of one of the Association's clubs.

2a The Association wants to create a database to keep club and player details.

Complete the missing club and player details in the analysis of inputs table below:

(2 marks)

Club details:	Player details:
Street Postcode	Forename Surname

- ◆ Check your answers carefully, as you cannot return to part A after you hand it in.
- ◆ When you are ready, hand part A to your teacher or lecturer and collect part B.

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

## Task 2: database design and development (part B)

2b Complete the data dictionary below, by identifying the primary and foreign keys in the relational database.

(2 mark)

Entity: Club					
Attribute name	Key	Type	Size	Required	Validation
clubName		text	20	Y	length <= 20
street		text	40	Y	length <= 40
postcode		text	8	Y	length <= 8
formed		date		Y	
league		number		Y	Restricted choice: 1,2,3

Entity: Player					
Attribute name	Key	Type	Size	Required	Validation
forename		text	20	Y	
surname		text	30	Y	
registration		number		Y	Range: >= 100000 and <= 999999
clubName		text	20	Y	Existing clubName from Club table
mobileNo		text	12	Y	length = 12
dateOfBirth		date		Y	
position		text	10	Y	Restricted choice: Striker, Midfielder, Defender, Goalkeeper
shirtNumber		number		Y	Range: >= 1 and <= 25

- ◆ Check your answers carefully, as you cannot return to part B after you hand it in.
- ◆ When you are ready, hand part B to your teacher or lecturer and collect part C.

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

## Task 2: database design and development (part C)

- 2c Your teacher or lecturer will provide you with a database file containing two linked tables.

Entity: Club					
Attribute name	Key	Type	Size	Required	Validation
clubName	PK	text	20	Y	length <= 20
street		text	40	Y	length <= 40
postcode		text	8	Y	length <= 8
formed		date		Y	
league		number		Y	Restricted choice: 1,2,3

Entity: Player					
Attribute name	Key	Type	Size	Required	Validation
forename		text	20	Y	
surname		text	30	Y	
registration	PK	number		Y	Range: >= 100000 and <= 999999
clubName	FK	text	20	Y	Existing clubName from Club table
mobileNo		text	12	Y	length = 12
dateOfBirth		date		Y	
position		text	10	Y	Restricted choice: Striker, Midfielder, Defender, Goalkeeper
shirtNumber		number		Y	Range: >= 1 and <= 25

Using the data dictionary above, complete the relational database by adding the required validation to the shirtNumber field.

(1 mark)

Print evidence to show that you have added the validation to the database, to match the data dictionary requirements.

2d(i) Noreen Glass, registration number 814209, has moved teams from Aviemore Aces to Dundee North. She will play in the number 24 shirt at her new club.

Implement **one** SQL statement that will make the required changes to Noreen's information.

**(3 marks)**

Print evidence of the SQL statement and the 'Player' table, clearly showing the change you have implemented.

(ii) The Association would like to invite suitable players to a goalkeeper coaching day.

Implement an SQL statement that will only display a list of club names, players' full names and mobile phone numbers for all league 1 goalkeepers.

**(4 marks)**

Print evidence of the SQL statement and the output.

2e The Association's rules state that players who play in the 'Striker' position are given a shirt number between 10 and 15.

Test the following SQL statement, which is intended to identify strikers who do not have the correct shirt number:

```
SELECT forename, surname
FROM Player
WHERE shirtNumber <10
OR shirtNumber > 15;
```

Explain why the output is not correct.

**(1 mark)**

2f The following SQL statement produces an error when executed.

```
INSERT INTO Player  
VALUES (220745, "Unknown", "Erin", "Smith", "07993 874657",  
"31/05/1999", "Striker", 23);
```

(i) Identify the value in the SQL statement that produces an error.

(1 mark)

(ii) Explain why this error is expected if the database is fit for purpose.

(1 mark)

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

## Task 3: web design and development

Montpel Youth Club would like to create a website to advertise their club.

The website should contain information about where and when the club meets. Pictures of each club leader should be displayed on the home page, along with a short paragraph detailing their experience of working with young adults.

Each night, leaders run an activity. Lists of the activities for the current and following weeks need to be kept up-to-date on the website. The club also wants to include some additional activity information. For example, for the first aid activity, the club would like a page showing what members learned at this activity.

Club members pay to attend some of the activities. Prices should be listed on the website.

3a Identify **two** end-user requirements for this website.

End-user requirement 1	(1 mark)
End-user requirement 2	(1 mark)

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

First aid is one of the activities run by the youth club. The 'First Aid' page will include content of what is covered at the activity, as follows:

- ◆ a link to a website which contains a video of how to make an arm sling
- ◆ a main heading 'How to make an arm sling'
- ◆ a numbered list detailing the steps required to put someone's arm into a sling
- ◆ a graphic called 'armsling.jpg'
- ◆ a paragraph of text introducing the numbered list

3b Draw a wireframe design showing how you would position the above content on the page.

(2 marks)



Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

Your teacher or lecturer will provide you with a copy of the unfinished website.

Open this and look carefully at:

- ◆ the home page
- ◆ the content of each page and the incomplete 'First Aid' page
- ◆ the hyperlinks within the website

3c The 'First Aid' page currently has no content. Open the 'firstAid.html' file in order to edit it.

Implement your design using HTML. The file 'armsling.jpg' has been provided within the website files.

**(4 marks)**

The additional information required to complete the 'link to a website', 'paragraph of text' and 'numbered list' is listed below:

- ◆ video page link address

<https://www.sja.org.uk/get-advice/first-aid-advice/how-to/how-to-make-an-arm-sling/>

- ◆ paragraph of text

When someone hurts their arm or shoulder badly, it is important to hold the injury still using a sling. Instructions detailing how to put a sling on a patient are given below.

- ◆ numbered list

- 1 Before applying a sling, check for cuts and make sure any bleeding has stopped.
- 2 For forearm slings, use padding for the injured arm and tie the sling around the patient's neck on the uninjured side.
- 3 For shoulder or collarbone slings, drape the long side of the bandage down from the shoulder on the uninjured side, bring it over your patient's arm and tie it behind their back.
- 4 Make sure the sling keeps your patient's arm in place but is not so tight that it limits blood flow.
- 5 If there is severe bleeding or if you suspect a dislocated joint or broken bone, see a doctor immediately.

3d(i) Sound mixing is another activity that the youth club has created a web page for.

The activity leader would like to add a sound mixing example to this page.

Open the soundMixing.html page in order to edit it.

Using the file 'mixExample.mp3', edit the 'Sound Mixing' page to add this sound.

**(1 mark)**

3d(ii) User feedback suggested changes to how the website looks.

Open the 'home.html' and 'styles.css' files in order to edit them.

Using external styles, make all the following changes:

- ◆ the background colour of the page should be "lightblue" (#ADD8E6)
- ◆ the middle 'Facilities' part of the page should be styled with the same colour as the page background
- ◆ the h2 headings should be reduced to size 14 and the font changed to Arial

**(3 marks)**

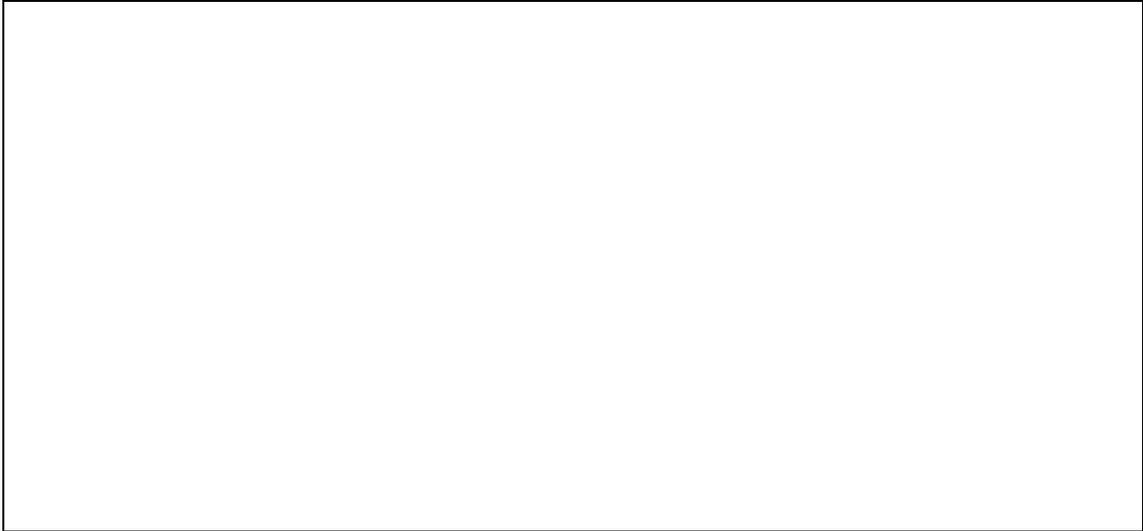
Print evidence of your code from these edited files:

- ◆ firstAid.html
- ◆ soundMixing.html
- ◆ home.html
- ◆ styles.css

3e Test the website to ensure that all media displays correctly and that any hyperlinks work as expected.

Discuss the results of testing the website.

**(1 mark)**



Candidate name\_\_\_\_\_ Candidate number\_\_\_\_\_

3f Initial analysis identified the following functional requirements for the website.

The website should display:

- ◆ where the club meets
- ◆ when the club meets
- ◆ photographs of each leader
- ◆ a paragraph about each leader
- ◆ a list of activities for this week
- ◆ a list of activities for next week
- ◆ additional information on activities, where required
- ◆ the cost of some activities

Use the above analysis to evaluate the finished website in terms of fitness for purpose.

(2 marks)

Candidate name \_\_\_\_\_ Candidate number \_\_\_\_\_

# Copyright acknowledgements

## Electronic files:

- ◆ Nejc Vesel/Shutterstock.com
- ◆ Rawpixel.com/Shutterstock.com
- ◆ Alejandro Magaña/Mixkit.com

# Administrative information

---

Published: August 2023 (version 1.0)

---

## History of changes

Version	Description of change	Date

## Security and confidentiality

This document can be used by SQA approved centres for the assessment of National Courses and not for any other purpose.

This document may only be downloaded from SQA's designated secure website by authorised personnel.

© Scottish Qualifications Authority 2023