Coimisiún na Scrúduithe Stáit
State Examinations Commission

Leaving Certificate 2026

Marking Scheme

Computer Science

Ordinary Level

Note to teachers and students on the use of published marking schemes

Marking schemes published by the State Examinations Commission are not intended to be standalone documents. They are an essential resource for examiners who receive training in the correct interpretation and application of the scheme. This training involves, among other things, marking samples of student work and discussing the marks awarded, so as to clarify the correct application of the scheme. The work of examiners is subsequently monitored by Advising Examiners to ensure consistent and accurate application of the marking scheme. This process is overseen by the Chief Examiner, usually assisted by a Chief Advising Examiner. The Chief Examiner is the final authority regarding whether or not the marking scheme has been correctly applied to any piece of candidate work.

Marking schemes are working documents. While a draft marking scheme is prepared in advance of the examination, the scheme is not finalised until examiners have applied it to candidates' work and the feedback from all examiners has been collated and considered in light of the full range of responses of candidates, the overall level of difficulty of the examination and the need to maintain consistency in standards from year to year. This published document contains the finalised scheme, as it was applied to all candidates' work.

In the case of marking schemes that include model solutions or answers, it should be noted that these are not intended to be exhaustive. Variations and alternatives may also be acceptable. Examiners must consider all answers on their merits, and will have consulted with their Advising Examiners when in doubt.

Future Marking Schemes

Assumptions about future marking schemes on the basis of past schemes should be avoided. While the underlying assessment principles remain the same, the details of the marking of a particular type of question may change in the context of the contribution of that question to the overall examination in a given year. The Chief Examiner in any given year has the responsibility to determine how best to ensure the fair and accurate assessment of candidates' work and to ensure consistency in the standard of the assessment from year to year. Accordingly, aspects of the structure, detail and application of the marking scheme for a particular examination are subject to change from one year to the next without notice.

Marking Scheme – Section C

Structure of the marking scheme for Section C (Programming)

Candidate responses are marked according to different scales, depending on the types of response anticipated. Scales labelled B divide candidate responses into four categories (correct response, almost correct response, partially correct response, and response of no substantial merit), and so on. The scales and the marks that they generate are summarised in this table:

Scale Label	A	B	C
No. of categories	3	4	5
5 mark scale		0, 2, 3, 5	
10 mark scale		0, 3, 7, 10	0, 3, 5, 8, 10
15 mark scale	0, 8, 15		

A general descriptor of each point on each scale is given below. More specific directions in relation to interpreting the scales in the context of each question are given in the scheme, where necessary.

Marking scales – level descriptors

A-scales (3 categories)

- response of no substantial merit
- response about half-right
- correct response

B-scales (4 categories)

- response of no substantial merit
- response with some merit
- almost correct response
- correct response

C-scales (5 categories)

- response of no substantial merit
- response with some merit
- response about half-right
- almost correct response
- correct response

Section A — Short Answer Questions — 54 marks

Answer any **nine** questions.

Question 1

Column A Variable	Column B Output
a	**8**
b	**3**
c	**8**
d	**2 or 2.0**
e	**1**

First correct response 2 marks
Each subsequent correct response 1 mark

Question 2

14

Correct answer 6 marks
Response about half right 4 marks
Response with some merit 2 marks

Question 3

(a)

- **Primary school children**
- **Teachers**

Correct answer 2 marks

(b)

- **Test the software for errors (debugging/user acceptance testing etc)**
- **Check the program meets the original requirements**
- **Any similar acceptable task**

Each correct item 2 marks (max 4 marks)

Question 4

	Yes (ML)	No (Not ML)
A spam filter gets better at spotting junk emails the more users mark messages as spam.	✓	
A programmer writes code to sort numbers into order.		✓
A calculator uses a formula to find the square root of a number.		✓
A voice assistant (like Siri or Alexa) improves at understanding your accent over time.	✓	
A photo app groups pictures of the same person by recognising faces.	✓	
Traffic lights are programmed to change every 60 seconds, no matter how many cars are waiting.		✓

Each correct answer 1 mark

Question 5

(a)

- **It shows only the main parts of the system.**
- **It leaves out unnecessary details.**
- **It uses a diagram instead of showing the real system.**
- **It focuses on how parts link together, not how they are built.**
- **Any other valid response**

Good explanation – clear understanding demonstrated 3 marks
Fair explanation – limited understanding 2 marks

(b)

- **It makes the system easier to understand.**
- **It helps designers focus on important parts only.**
- **It reduces complexity by leaving out details.**
- **It makes planning the system easier.**
- **It helps explain the system to others.**
- **Any other valid advantage**

Good response – clear understanding demonstrated 3 marks
Fair response – limited understanding 2 marks

Question 6

- **Computer modelling is safer because it avoids dangerous real-life tests, such as car crashes or severe weather conditions.**
- **It is cheaper as simulations can be run many times without using real materials or building full prototypes.**
- **It is faster as the process and be sped up and repeated instantly.**
- **Any other valid reason**

For each valid reason:

Good – clear understanding demonstrated	3 marks
Fair – limited understanding	2 marks

Question 7

Pass 2	**12**	**22**	**43**	**15**	**17**	**8**
Pass 3	**12**	**15**	**22**	**43**	**17**	**8**
Pass 4	**12**	**15**	**17**	**22**	**43**	**8**

Each correct row 2 marks

Question 8

Arguments for:

- **Reduce face-to-face communication**
- **Cause people to ignore those physically present**
- **Increase distraction and lack of attention**
- **Lower quality of conversations**
- **Encourage addiction and constant checking**
- **Any valid argument for**

Arguments against:

- **Allow people to stay in touch easily**
- **Help maintain long-distance relationships**
- **Enable instant communication in emergencies**
- **Support social connections through messaging and video calls**
- **Help people feel less lonely**
- **Allow families and friends to share experiences**
- **Any valid argument against**

For each valid argument:

Good – clear understanding demonstrated	3 marks
Fair – limited understanding	2 marks

Question 9

Input devices:

- **Keyboard**
- **Mouse**
- **Microphone**
- **Scanner**
- **Webcam**
- **Touchscreen**
- **Any other valid input device**

Output devices:

- **Monitor**
- **Printer**
- **Speakers**
- **Headphones**
- **Projector**
- **Any other valid output device**

For input/output:

First correct	2 marks
Second correct	1 mark

Question 10

(a)

Discrete Data Example: Number of steps taken, number of floors climbed, number of workouts, number of calories burned, number of notifications, etc.

Continuous Data Example: Heart rate, body temperature, running speed, distance travelled, time duration, GPS position, etc.

Each correct example 2 marks (max. 4 marks)

(b)

- **A list of daily step counts.**
- **A list of heart rate readings over time.**
- **A list of distances travelled for each workout.**
- **A list of calories burned each day.**
- **A list of recorded temperatures during an activity.**
- **Any other valid example of a list**

Correct valid example 2 marks

Question 11

Syntax errors:

Line 2	**Missing bracket**	**close bracket**
Line 6	**No concatenation**	**missing use of , / missing use of +**

Logic error:

Line 4	**This concatenates 2 strings instead of adding 2 integers (e.g. 2 + 3 = 23, instead of 2 + 3 = 5).**
Lines 1, 2	**Variables need to be cast (either on lines 1 and 2 as part of input or on line 4 as part of addition expression).**

Each correct line identified	1 mark
Each error explained	1 mark
Each correct solution to fix	1 mark

Question 12

Character	Can be encoded	Cannot be encoded
e	☑	☐
R	☑	☐
☺	☐	☑
?	☑	☐
4	☑	☐
✓	☐	☑

Each correct item	1 mark

Section B — Long Questions — 76 marks

Answer any **two** questions.

Question 13 — **38 marks (19, 7, 12)**

(a) — **19 marks (13, 4 ,2)**

(i)

Column A Variable	Column B Datatype
`fruits`	**list**
`target`	**string**
`position`	**Integer / int**
`found`	**Boolean / Bool**

First correct response	5 marks
Second correct response	4 marks
Third correct response	2 marks
Fourth correct response	2 marks

(ii) — **4 marks**

orange found after 4 steps.

Correct answer	4 marks
Response of some merit	3 marks

(iii) — **2 marks**

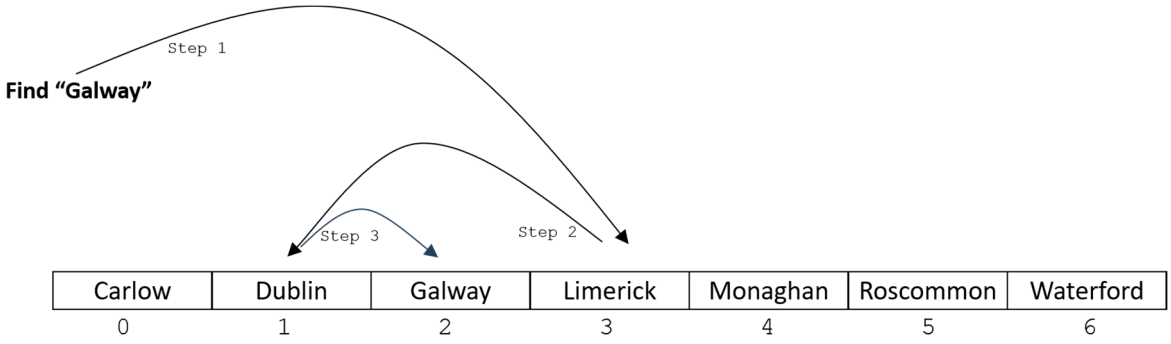
`lower()` is used to convert the user input into lowercase and is also used to convert the items in the list to lowercase. This means that the user can enter lowercase or uppercase or a mixture and the program will work no matter what case is entered. The programme is always comparing lowercase to lowercase.

Good explanation - clear understanding demonstrated	2 marks
Fair explanation - limited understanding	1 mark

(b) **7 marks (3, 4)**

(i) **3 marks**

REPEAT
SELECT the middle element
COMPARE the middle element with "Galway"
If middle element == "Galway" STOP
If middle element > "Galway" DISCARD all items above middle element
If middle element < "Galway" DISCARD all items below middle element



Correct answer	3 marks
Response about half right	2 marks
Response with some merit	1 marks

(ii) **4 (2, 2) marks**

- **Linear search works on unsorted data.**
- **Binary search requires the data to be sorted.**

- **Linear search checks each item one by one.**
- **Binary search halves the list each time.**

- **Linear search is slower for large lists.**
- **Binary search is faster for large lists.**

- **Linear search may check every item.**
- **Binary search checks far fewer items.**

For each difference:

Good description – clear understanding demonstrated	2 marks
Fair description – limited understanding	1 marks

(c) **12 (6, 6) marks**

Sorting Algorithms:

- **Amazon sorts items by delivery speed ("Prime First").**
- **Netflix sorts content into categories ("Top 10", "Trending Now").**
- **eBay sorts auctions by ending soonest or lowest price.**
- **Booking.com sorts hotels by rating or cheapest first.**
- **Accept other relevant answers.**

Searching Algorithms:

- **Netflix uses searching so users can quickly find shows by title, actor, or genre.**
- **Spotify uses searching to find songs, artists or playlists instantly.**
- **Google Maps uses searching to find locations or businesses.**
- **Accept other relevant answers.**

For each algorithm:

Good explanation - clear understanding demonstrated	6 marks
Limited explanation – lacks clarity	5 mark

Question 14 **38 (10, 12, 16) marks**

(a) **10 (6, 4) marks**

(i) **6 (3, 3) marks**

Possible benefits:

- **Faster vote counting – results can be calculated much more quickly than counting paper ballots.**
- **Reduced human error – fewer mistakes compared to manual counting.**
- **More efficient elections – less staff needed to count votes.**
- **Accessibility – can be designed to help voters with disabilities.**
- **Less paper used – more environmentally friendly.**
- **Immediate results – quicker announcement of election outcomes.**
- **Any other valid benefit.**

Possible costs:

- **High financial cost – expensive to buy, store, maintain, and update machines.**
- **Security risks – possibility of hacking, tampering, or software faults.**
- **Technical failures – machines could break down on election day.**
- **Lack of public trust – voters may not trust electronic systems as much as paper ballots.**
- **Training required – staff must be trained to use and manage the machines.**
- **Any other valid cost.**

For each valid answer:

Good explanation - clear understanding demonstrated	3 marks
Fair explanation - limited understanding	2 marks

(ii) **4 (2, 2) marks**

Voters – machines are easy to use/ accessible/ trustworthy/ confidential
Government – machines are secure/ reliable/ affordable
Polling Staff – machines are easy to set up/ clear training given/ count accurately
Political Parties – machines are fair & transparent/ verified results/ accurate/ can't be hacked or tampered

Each stakeholder identified	1 mark
Each valid requirement explained	1 mark

(b) **12 (4, 4, 4) marks**

(i) **4 (2, 2) marks**

- **Equitable Use - The design is useful and marketable to people with diverse abilities.**
  The voting machine could be adjustable in height for wheelchair users.
- **Flexibility in Use - The design accommodates a wide range of individual preferences and abilities.**
  The voting machine could have option of buttons, touch screen, voice control, etc.
- **Simple and Intuitive Use - The design is easy to understand, regardless of the user's experience or knowledge.**
  The voting machine should have clear instructions, easy-to-read screens, and intuitive navigation buttons make it easy for everyone to vote without confusion.
- **Perceptible Information - The design communicates necessary information effectively, using different methods (text, sound, visuals).**
  The voting machine communicates effectively to the user, e.g., using text, audio, or visual cues.
- **Tolerance for Error - The design minimises hazards and mistakes and allows users to correct errors.**
  The voting machines might allow voters to review and change choices before final submission.
- **Low Physical Effort - The design can be used comfortably with minimal fatigue.**
  The voting machine might have touch sensitive screen rather than buttons
- **Size and Space for Approach and Use - The design provides appropriate space for reach, manipulation, and use regardless of body size or mobility.**
  The machine is positioned so wheelchair users or people using mobility aids can comfortably reach the screen and controls. There is enough space around the machine to allow users to approach, move, and operate it without obstruction.

For each response:

Correct principle named	1 mark
Good description - understanding demonstrated	1 mark

(ii) **4 (2, 2) marks**

- **Screen reader / text-to-speech support - Reads out instructions and candidate names for visually impaired users.**
- **Adjustable text size - Allows users to increase or decrease font size for easier reading.**
- **High-contrast mode - Improves visibility for users with poor eyesight or colour blindness.**
- **Voice control - Allows users to navigate and vote using voice commands instead of touch.**
- **Simple, uncluttered interface - Large buttons and clear screens reduce confusion and mistakes.**
- **Audio instructions - Step-by-step spoken guidance helps users who struggle with reading.**
- **Confirmation and review screen - Allows users to check and change their vote before submitting.**
- **Multiple language options - Supports users whose first language is not English.**
- **Any other valid answer.**

For each accessibility feature:

Good description - clear understanding demonstrated	2 marks
Fair description - limited understanding	1 mark

(iii) **4 marks**

- **Security and privacy of voter data – ensuring that users' personal information and votes are protected from hacking or misuse. Breaches could compromise voter anonymity or allow manipulation of results.**
- **Accessibility and fairness – the app must be accessible to all eligible voters, including those with disabilities or limited access to smartphones/internet. Otherwise, it could unfairly exclude certain groups.**
- **Integrity of the election – technical errors, bugs, or manipulation could affect the accuracy of vote counting, undermining trust in the democratic process.**
- **Proof of identity – someone could pretend to be someone else and cast their vote.**

Good explanation – clear understanding demonstrated	4 marks
Fair explanation – limited understanding	3 marks

(c) **16 (4, 12) marks**

(i) **4 marks**

- **Algorithms prioritise content the user has interacted with before.**
- **Algorithms push content that is popular or highly shared.**
- **Algorithms create a "filter bubble".**
- **Algorithms use personal data to target political ads.**
- **Any other valid answer.**

Good description – clear understanding demonstrated	4 marks
Fair description – limited understanding	3 marks

(ii) **12 (6, 6) marks**

Arguments For:

- **Protects mental health - Reduces anxiety, depression, and pressure caused by likes, followers, and comparisons.**
- **Reduces cyberbullying - young people are less exposed to online harassment and abuse.**
- **Protects children from harmful content - limits access to inappropriate material, misinformation, or online predators.**
- **Improves focus and education - less distraction from schoolwork and better concentration.**
- **Encourages healthier lifestyles - promotes more face-to-face interaction and physical activity.**
- **Increased social skills - young people become more inclined to speak with people in person.**
- **Any other valid argument for.**

Arguments Against:

- **Limits freedom of expression - young people may feel their right to communicate and express themselves is restricted.**
- **Difficult to enforce - children may lie about their age or use someone else's account.**
- **Social exclusion - teens may feel left out if their friends use social media and they cannot.**
- **Loss of positive uses - social media can be useful for learning, creativity, and staying in touch with friends and family.**
- **Better regulation instead of bans - education and parental controls may be more effective than a full ban.**
- **Any other valid argument against.**

For each valid argument:

Good - clear understanding demonstrated	6 marks
Fair - limited understanding	5 marks

Question 15 **38 (19, 13, 6) marks**

(a) **19 (10, 9) marks**

(i) **10 (4, 4, 2) marks**

- **Embedded systems do one specific task, while general-purpose computers can do many tasks.**
- **Embedded systems run one fixed program, but general-purpose computers let users install lots of software.**
- **Embedded systems use limited hardware (small memory, low power), while general-purpose computers have powerful processors and lots of RAM.**
- **Embedded systems often have little or no user interface, while general-purpose computers have full interfaces like keyboards, touchscreens, and operating systems.**
- **Embedded systems are built into another device (e.g., microwave, car), but general-purpose computers work as standalone machines.**
- **Embedded systems must respond in real time, while general-purpose computers do not always need strict timing.**
- **Embedded systems are usually cheaper and smaller, while general-purpose computers are larger and more expensive.**
- **Any other valid advantage.**

First correct response	4 marks
Second correct response	4 marks
Third correct response	2 marks

(ii) **9 marks**

Column A **Hardware**	**Column B** **Digital or Analogue**
Power button on a dishwasher	**Digital**
A microphone receiving sound	**Analogue**
A temperature sensor	**Analogue**
A security switch to detect if a door is open or closed	**Digital**
An outdoor motion detection light	**Digital**
A street light that turns on based on darkness levels	**Analogue**

First three correct answers	2 marks each
Each subsequent correct answer	1 mark each (max. 3)

(b) **13 marks (9, 4)**

(i) **9 marks**

Column A Conditions	Column B Status (on/off)
Temperature = 12°C, Humidity = 55%	**Dehumidifier: OFF** **Cooler: OFF** **Alarm: OFF**
Temperature = 3°C, Humidity = 80%	**Dehumidifier: ON** **Cooler: OFF** **Alarm: ON**
Temperature = 18°C, Humidity = 30%	**Dehumidifier: OFF** **Cooler: ON** **Alarm: ON**

Each correct ON / OFF state 1 mark

(ii) **4 marks**

Inputs:

- **Temperature sensor / thermometer**
- **Humidity sensor**

Outputs:

- **Dehumidifier**
- **Cooler**
- **Alarm**

Each correct device identified 1 mark

(c) **6 (4, 2) marks**

(i) **4 (2, 2) marks**

- **Incorrect system decisions - if the temperature reading is missing, the system may not turn the fan on or off at the correct time.**
- **System acting unnecessarily - the system may assume unsafe conditions and turn devices on or off when it is not needed.**
- **Increased energy - fans or dehumidifier systems could run longer than required, wasting electricity.**
- **Reduced reliability of the system - frequent errors reduce trust in the monitoring system.**
- **Delayed fault detection - real environmental problems might go unnoticed if sensor data is missing or incorrect.**
- **Any other valid problem.**

For each valid answer:

Good description – clear understanding demonstrated	2 marks
Fair description – limited understanding	1 mark

(ii) **2 marks**

- **Data validation check - the system can check if values are within a normal range and reject or flag invalid data like "Error".**
- **Automatic re-reading of sensors - if a reading is missing or invalid, the system can take another reading immediately.**
- **Use the last valid reading or average reading - if data is missing, the system can temporarily use the previous correct value or an average of all values**
- **Alert the user - end a warning when sensor errors occur so the problem can be fixed quickly.**
- **Any other valid suggestions.**

Good description – clear understanding demonstrated	2 marks
Fair description – limited understanding	1 mark

Section C — Programming — 80 marks

Question 16 — **80 (60, 20) marks**

(a) — **60 (15, 10, 10, 10, 10, 5) marks**

Possible solution:

```
1  # Question 16(a)
2  # Examination Number:
3
4  print("GAA Scoreboard Calculator") #(i)
5
6  team_a_goals = int(input("Enter Team A goals: ")) #(iii)
7  team_a_points = int(input("Enter Team A points: ")) #(iii)
8  team_a_total = team_a_goals * 3 + team_a_points    #(ii) /(iii)
9  print("Team A total score:", team_a_total) #(ii)
10
11 team_b_goals = int(input("Enter Team B goals: ")) #(iv)
12 team_b_points = int(input("Enter Team B points: "))
13 team_b_total = team_b_goals * 3 + team_b_points
14 print("Team B total score:", team_b_total)
15
16 if team_a_total > team_b_total:
17     difference = team_a_total - team_b_total
18     print("Result: Team A wins")
19     print("Winning margin: ", difference)
20 elif team_b_total > team_a_total:
21     difference = team_b_total - team_a_total
22     print("Result: Team B wins")
23     print("Winning margin: ", difference)
24 else:
25     print("It was a draw")
```

(i) — **15 marks (A-15 scale)**

15 marks	**Correct response** Correct implementation using solution above or similar.
8 marks	**Response with some merit** Any other reasonable attempt.

(ii) — **10 marks (B-10 scale)**

10 marks	**Correct response** Correct implementation using solution above or similar.
7 marks	**Almost correct response** Correct implementation using solution above or similar but with minor syntax or semantic error.
3 marks	**Response with some merit** Any other reasonable attempt.

(iii) **10 marks (C-10 scale)**

10 marks	**Correct response** Correct implementation using solution above or similar.
8 marks	**Almost correct response** Correct implementation using solution above or similar but with minor syntax or semantic error.
5 marks	**Response about half-right** Partially correct implementation using solution above or similar but with significant syntax or semantic error.
3 marks	**Response with some merit** Any other reasonable attempt.

(iv) **10 marks (C-10 scale)**

10 marks	**Correct response** Correct implementation using solution above or similar.
8 marks	**Almost correct response** Correct implementation using solution above or similar but with minor syntax or semantic error.
5 marks	**Response about half-right** Partially correct implementation using solution above or similar but with significant syntax or semantic error.
3 marks	**Response with some merit** Any other reasonable attempt.

(v) **10 marks (C-10 scale)**

10 marks	**Correct response** Correct implementation using solution above or similar.
8 marks	**Almost correct response** Correct implementation using solution above or similar but with minor syntax or semantic error.
5 marks	**Response about half-right** Partially correct implementation using solution above or similar but with significant syntax or semantic error.
3 marks	**Response with some merit** Any other reasonable attempt.

(vi) **5 marks (B-5 scale)**

5 marks	**Correct response** Correct implementation using solution above or similar.
3 marks	**Almost correct response** Correct implementation using solution above or similar but with minor syntax or semantic error.
2 marks	**Response with some merit** Any other reasonable attempt.

(b) **20 marks**

Possible solution:

```
# Question 16(b)
# Examination Number:

# Display the program title
print("Team Data App")
print()

# Get the team name from the user
team_name = input("Enter team name: ")

# Initialise counters for wins, losses, and draws
wins = 0
losses = 0
draws = 0

# Loop to get results for 5 games
for i in range(5):
    result = input("Result (W/L/D): ")

    # Count the result
    if result == "W":
        wins = wins + 1
    elif result == "L":
        losses = losses + 1
    elif result == "D":
        draws = draws + 1

# Display the results
print()
print("*** " + team_name + " ***")
print("Win: ", wins)
print("Loss: ", losses)
print("Draw: ", draws)
```

	High level of achievement **All** of the following implemented correctly and efficiently	**Moderate level of achievement** Reasonable attempt to implement at least two of each of the following	**Low level of achievement** Poor attempt to complete any of the following
Program Inputs (5 marks)	• Team name read in correctly • Five results read in correctly • Variable initialisation (i.e. wins, losses, draws etc.) and use of assignment statements (5 marks)	• Team name read in correctly • Five results read in correctly • Variable initialisation (i.e. wins, losses, draws etc.) and use of assignment statements (4 marks)	• Team name read in correctly • Five results read in correctly • Variable initialisation (i.e. wins, losses, draws etc.) and use of assignment statements (3 marks)
Processing & Logic (5 marks)	• Correct logic to count wins • Correct logic to count losses • Correct logic to count draws (5 marks)	• Correct logic to count wins • Correct logic to count losses • Correct logic to count draws (4 marks)	• Correct logic to count wins • Correct logic to count losses • Correct logic to count draws (3 marks)
Program Output (5 marks)	• Program displays "Team Data App" • Outputs team name with asterisk on either side • Outputs correct numbers of wins, losses and draws (5 marks)	• Program displays "Team Data App" • Outputs team name with asterisk on either side • Outputs correct numbers of wins, losses and draws (4 marks)	• Program displays "Team Data App" • Outputs team name with asterisk on either side • Outputs correct numbers of wins, losses and draws (3 marks)
Programming Standards (5 marks)	• Program executes correctly with no syntax or runtime errors • Program meets all requirements • Program design is well explained with comments (5 marks)	• Program executes correctly with no syntax or runtime errors • Program meets more than half of the requirements • Program design is well explained with comments (4 marks)	• Program executes correctly with no syntax or runtime errors • Program meets any of the requirements • Program design is well explained with comments (3 marks)

Coursework (90 marks in total)	
The report	***Marks***
• Quality of report website structure and layout. • Evidence of adherence to the principles of good user interface design when creating the website. • Adherence to the word count and video length (penalties may apply).	**5**
1. Meeting the brief	
• Meeting the basic requirements of the brief. • Meeting the advanced requirements of the brief.	**27**
2. Investigation	
• Research on the brief, including research on existing solutions, systems or ideas that are aligned to the brief. • Consideration of how your research informs decisions about your project.	**10**
3. Plan and design	
• A clear, detailed description of the design objectives of the project. • Consideration of different options and justification of design choices. • Consideration of stakeholders and end user's needs. • The technologies that will be used. • System architecture/high-level flowchart showing components.	**18**
4. Create	
• Describe the key milestones of the development process. • Describe the testing that took place during the development process. • Explain a problem that was encountered in the development of the project and how it was overcome. • Describe the model or the embedded system algorithm that was programmed as part of the project.	**20**
5. Evaluation	
• An evaluation of your final artefact. • Suggest how you would further improve/iterate this project.	**10**
References and summary word count	
• You must also include references and/or a bibliography. • Include a summary of the word count of the report, including the total word count.	**0**

Higher grade	Ordinary grade	Reference Mark	Higher Mark	Ordinary Mark
1		81 – 90	81 – 90	90
2		72 – 80	72 – 80	90
3		63 – 71	63 – 71	90
4		54 – 62	54 – 62	90
5	1	45 – 53	45 – 53	81 – 90
6	2	36 – 44	36 – 44	72 – 80
7	3	27 – 35	27 – 35	63 – 71
8	4	23 – 26	23 – 26	54 – 62
	5	18 – 22	18 – 22	45 – 53
	6	14 – 17	14 – 17	36 – 44
	7	9 – 13	9 – 13	27 – 35
	8	0 – 8	0 – 8	0 - 26

COURSEWORK – conversion from reference mark to Ordinary-level mark

For Ordinary level candidates, the final mark is found from the reference mark as follows:

- If the reference mark is 54 or more the final mark is 90.
- If the reference mark is at least 27 but less than 54, then add 36 to the reference mark to get the final mark.
- If the reference is at least 1 but less than 27, then double the reference mark and add 9 to get the final mark.
- If the reference mark is 0 the final mark is 0

Reference Mark	Conversion
54 or more	Award 90 marks
27 – 53	Add 36 marks
1 - 26	Multiply the reference mark by 2 and add 9 marks
0	0