



Coimisiún na Scrúduithe Stáit
State Examinations Commission

Leaving Certificate Examination 2025
Computer Science

Sections A & B
Higher Level

Wednesday 21 May Morning 9:30 - 11:00
130 marks

Examination Number

Date of Birth

/

/

For example, 3rd February
2005 is entered as 03 02 05

For Examiner use only								
Section	Question	Mark	Section	Question	Mark	Section	Question	Mark
A	1		A	7		B	13	
	2			8			14	
	3			9			15	
	4			10		Section B Total:		
	5			11		P1 Total (A+B)		
	6			12		C	16	
						P2 (Sec C) Total:		
Section A Total:						Total:		

Instructions

There are **three** sections in this examination. Section A and B appear in this booklet. Section C is in a separate booklet that will be provided for the computer-based element.

Section A	Short Answer Questions	Attempt any nine questions All questions carry equal marks	54 marks
Section B	Long Questions	Attempt any two questions All questions carry equal marks	76 marks
Section C	Programming	Answer all question parts	80 marks

Calculators may **not** be used during this section of the examination.

The superintendent will give you a copy of page 78 (Logic gates) of the *Formulae and Tables* booklet on request. You are **not** allowed to bring your own copy into the examination.

Write your answers for Section A and Section B in the spaces provided in this booklet. There is space for extra work at the end of the booklet. Label any such extra work clearly with the question number and part.

Answer any **nine** questions.

Question 1

State the output of the following Python code.

```
1 a = 5
2 b = a + 2
3 b = 10
4 print(a)
5 print(b)
6 a = b * 2
7 c = a // b
8 print(a)
9 print(c)
10 a = a + 1
11 print(a % b)
12 print(b == c)
```

Output:

--

Question 2

ASCII and Unicode are both examples of character encoding systems. Compare ASCII and Unicode under the headings, memory usage and compatibility.

Memory usage:
Compatibility:

Question 3

Four girls recently competed in a coding hackathon, each creating a project with a unique theme. Each girl achieved a different ranking in the competition. Use the following clues to determine who created each project and where they came in the competition (ranking).



- Clue 1: The person who completed an artificial intelligence project took first place.
- Clue 2: Grace came third.
- Clue 3: Ada's web development project was not last.
- Clue 4: Kay beat the person who created an embedded system project.
- Clue 5: The person who completed a data analytics project beat Joy.

Write your answers in the table below. Clues 1 and 2 have already been answered. Space is provided for roughwork if you need it.

Roughwork

Ranking	Girl's Name	Project theme
1		Artificial intelligence
2		
3	Grace	
4		

Question 4

Two key functions of an operating system are resource management and file management. Explain what is meant by the terms 'resource management' and 'file management'.

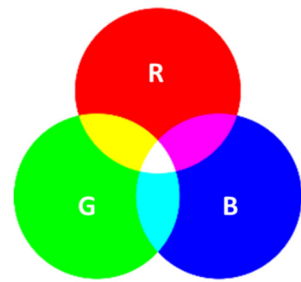
Resource management:
File management:

Question 5

In web design, colours are often represented using a hexadecimal code which defines the red, green, and blue (RGB) components of the colour.

For example, the hex colour #0A33F0 represents a shade of blue, where:

- the first two characters (0A) represent the red component.
- the next two characters (33) represent the green component.
- the last two characters (F0) represent the blue component.



Each component is represented in hexadecimal from 00 to FF (0 to 255 in decimal).

Convert the hexadecimal colour **#0A33F0** to its RGB decimal values.

Answer: Red:

Green:

Blue:

Question 6

Referring to the quote below from Professor Timothy Miller, University of Melbourne, describe **one** benefit and **one** limitation of using artificial intelligence (AI) in the area of medical diagnosis.

Will a computer take your job?

It's predicted that computers could one day become smarter than humans, but we shouldn't forget people power in the age of artificial intelligence.

Benefit:

Limitation:

Question 7

A binary question is a question that has two possible answers – yes or no. The diagrams, in **Figure 1** and **Figure 2** below, describe how binary questions can be chained together to determine the outcome of tossing a six-sided dice. Two different search algorithms are illustrated.

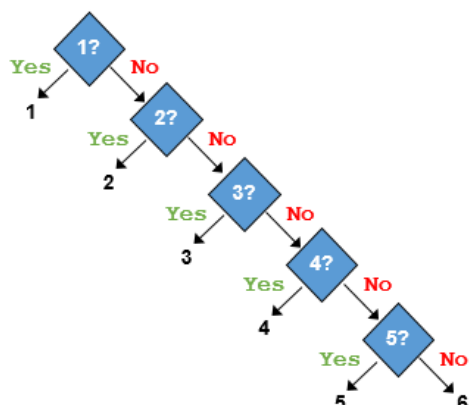


Figure 1: Algorithm 1 (tree depth = 5)

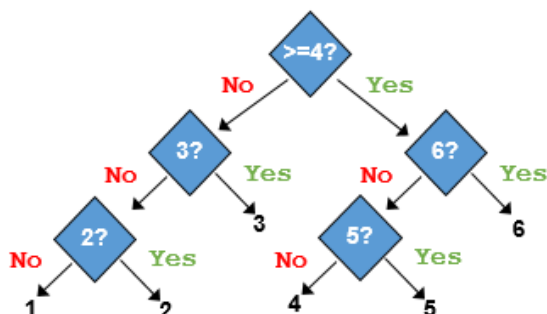


Figure 2: Algorithm 2 (tree depth = 3)

(a) State the name of each search algorithm.

Algorithm 1:
Algorithm 2:

(b) Suggest which of the two algorithms is most efficient. Justify your answer.

Most efficient:
Justify:

Question 8

Distinguish between the IP and VoIP communication protocols.

Question 9

The intention of the Python code below is to scan through all integers from 1 to 20 inclusive and:

- if the integer is evenly divisible by both 3 and 5, display the word FizzBuzz.
- if the integer is evenly divisible by 3 and not divisible by 5 display the word Fizz.
- if the integer is evenly divisible by 5 and not divisible by 3 display the word Buzz.

```
1 For i in range(1, 21):  
2     if i % 3 == 0:  
3         print("Fizz"  
4     elif i % 5 == 0:  
5         print(Buzz")  
6     elif i % 3 == 0 and i % 5 == 0:  
7         print("FizzBuzz")  
8     else  
9         print(number)
```

(a) There are a number of syntax errors in the code. Identify any **two** of these errors.

Syntax error 1:
Syntax error 2:

(b) Identify the logic error in the code and suggest a solution.

Logic error:
Suggested solution:

Question 10

List **two** examples of how computing technology is used to automate processes. Outline **one** limitation of either example.

Example 1:
Example 2:
Limitation:

Question 11

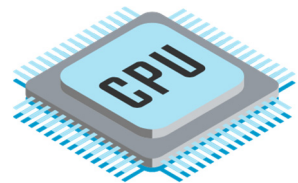
Social media platforms often use data from users to personalise content. Give **one** advantage and **one** potential risk of this approach in the context of political elections.

Advantage:
Potential risk:

Question 12

The Central Processing Unit (CPU) is often referred to as the "brain" of the computer.

Identify and state the purpose of **two** CPU components.



CPU Component 1:
Purpose:
CPU Component 2:
Purpose:

Answer any **two** questions.

Question 13

- (a) The Deposit Return Scheme (DRS) was introduced in Ireland in February 2024. It is designed to encourage recycling and reduce litter. Under this scheme, when a consumer buys a drink in a container with the Re-turn logo they are charged a small refundable deposit. The cost of the deposit depends on the size of the container.

Containers can be returned using a Reverse Vending Machine (RVM), such as that shown in **Figure 3**. The RVM issues a voucher that can be redeemed at the till against store purchases or for a cash deposit refund.



Figure 3

You are tasked with designing a software application that supports the DRS for a retail environment. The application should manage all stages of the lifecycle of drink containers.

- (i) As project manager one of your first responsibilities will be to assemble your project team. Identify **two** other roles on the project team and outline **two** responsibilities you would assign to each role.

Role 1:
Responsibility 1:
Responsibility 2:
Role 2:
Responsibility 1:
Responsibility 2:

This question continues on the next page.

- (ii) Once you have the project team in place you will need to decide on a software development process. State the name of **two** software development processes available to you.

1.
2.

- (iii) Choose the software development process that should be used for this project and justify your choice by describing **two** key features of the methodology you chose.

Choice:
Feature 1:
Feature 2:

- (b) You have identified store staff and customers as two key stakeholders in the system.

- (i) For each stakeholder outline **one** goal they may want from the system and **one** benefit they should gain from the system.

Goal (store staff):
Benefit (store staff):
Goal (customers):
Benefit (customers):

- (ii) State the name of **one** other stakeholder of the system once it goes live.

--

- (iii) You have identified database management as one of the key system components. Suggest **two** specific use cases of this component.

1.
2.

- (iv) Suggest **two** ways artificial intelligence (AI) technology could be used in the DRS application.

1.
2.

- (c) You are keen to adopt a user-centred approach to the design of the system and ensure that it provides an inclusive user experience for all users.

- (i) Read the following scenario and answer the question that follows.

Scenario: A visually impaired user needs assistance in identifying containers and navigating the DRS application.

Identify and explain **two** appropriate assistive technology solutions that could be used in the DRS system to enhance accessibility for this user.

Assistive technology 1:
Explain:
Assistive technology 2:
Explain:

This question continues on the next page.

- (ii) The system will need to be thoroughly tested before going live. This will include unit testing and functional testing. Distinguish between unit testing and functional testing.

Question 14

- (a) The Greatest Common Divisor (GCD) of two integers a and b is the largest number that divides into both with no remainder. One of the oldest and most famous algorithms was first described by Greek mathematician, Euclid, in his book of Elements written around 300BC. The code snippet below is an implementation of Euclid's algorithm using subtraction to find the GCD of 45 and 18.



```
1 def gcd(a, b):
2     while a != b:
3         if a > b:
4             a = a - b
5         elif b > a:
6             b = b - a
7
8     return a
9
10 print("The GCD is", gcd(45,18))
```

- (i) State the purpose of the `def` keyword on line 1 of the code.

- (ii) Starting from line 2 of the code, complete the trace table shown below. The variables a and b have been initialised to 45 and 18 respectively.

a	b
45	18

- (iii) This implementation uses a `while` loop. Explain why a `for` loop would **not** be suitable for this algorithm.

This question continues on the next page.

(b) The code below shows a recursive implementation to find the GCD of two integers.

```
1 def recursive_gcd(a, b):  
2     if b == 0:  
3         return a  
4     return recursive_gcd(b, a%b)  
5  
6 print("The GCD is", recursive_gcd(45, 18))
```

(i) Every recursive function, including the one shown here, has a base case and a recursive case. Outline the meaning of the terms 'base case' and 'recursive case'.

Base case:
Recursive case:

(ii) Explain the use of the variable `b` and the expression `a%b` on line 4 of the code.

b:
a%b:

(iii) Given a choice of using `gcd` shown in **part (a)**, or `recursive_gcd` shown here in **part (b)**, which version would you recommend to find the greatest common divisor of very large numbers on machines with limited RAM? Justify your answer, in terms of memory utilisation.

gcd or recursive_gcd:
Justify:

- (c) Quicksort is a widely used sorting algorithm developed by British computer scientist Tony Hoare in 1960. The algorithm itself is recursive and it uses a process called partitioning around a chosen pivot to achieve its objective. Quicksort is known for its efficiency and speed, especially in handling very long lists of data.



- (i) Given the unsorted list shown below and a pivot of 13, state the contents of the two sub-lists created after one partitioning step of quicksort.

29	10	14	37	13
----	----	----	----	-----------

Left sub-list:

Right sub-list:

- (ii) Describe the next steps quicksort would take in order to achieve its objective.

- (iii) Outline **two** common strategies used when selecting a pivot for a quicksort algorithm.

1.
2.

This question continues on the next page.

- (iv) Referring to **Figure 4** below, explain the best and worst-case time complexity of the quicksort algorithm.

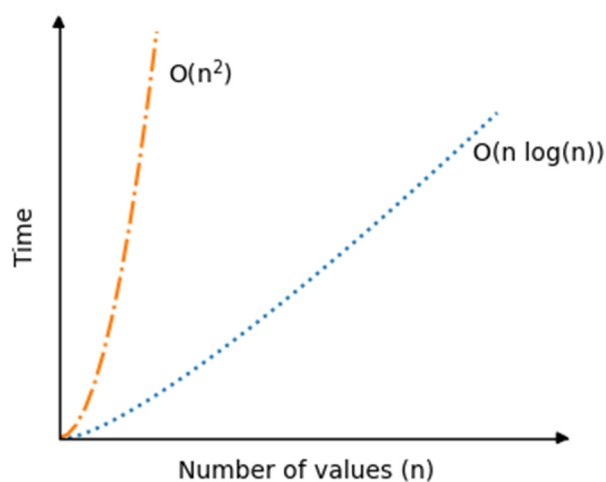


Figure 4

Best case:
Worst case:

Question 15

Modelling and simulation are essential tools for understanding and solving complex real-world problems. These approaches often involve breaking down systems, identifying key components, and examining how changes can influence outcomes.

- (a) Consider the dataset shown in **Figure 5** below which is being used by a computer science student as part of her Applied Learning Task 3 (modelling and simulation) to model driving test outcomes.

ID	Gender	Age	Transmission	Instructor experience (years)	Test outcome
1	Male	18	Manual	8	FAIL
2	Female	22	Automatic	5	PASS
3	Male	20	Automatic	8	PASS
4	Male	19	Manual	3	FAIL
5	Female	30	Automatic	12	PASS
6	Female	25	Manual	2	FAIL
7	Female	23	Automatic	10	PASS
8	Male	17	Manual	6	FAIL
9	Male	19	Manual	5	FAIL
10	Female	21	Manual	15	PASS

Figure 5

- (i) State the name of the output variable in the model.

--

- (ii) State the names of **two** input variables.

Input variable 1:
Input variable 2:

- (iii) Describe **two** patterns you can identify from the model.

Pattern 1:
Pattern 2:

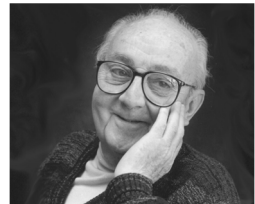
This question continues on the next page.

- (iv) Identify **two** additional variables that could be used to enhance the driving test outcome model.

Variable 1:
Variable 2:

- (b) The quote below is attributed to statistician George Box and is often cited in the context of modelling and simulation. It highlights the limitations of models while emphasising their practical value.

"All models are wrong, but some are useful"



- (i) Outline **one** limitation and **one** benefit of modelling and simulation.

Limitation:
Benefit:

- (ii) Explain why abstraction is important when developing a model.

(iii) Modelling and simulation can be used in many situations to address a wide variety of problems. Some example areas where modelling and simulation can be used to improve outcomes include:

- Education
- Mental wellbeing
- Environment
- Traffic management

Pick any **two** of the above areas and briefly describe how modelling could be used to address a problem and improve outcomes.

Area 1:
Describe:
Area 2:
Describe:

(c) A health agency wants to predict how an infectious disease might spread in a school and evaluate different containment strategies.

(i) Describe how agent-based modelling could be used to simulate disease spread in this scenario.

This question continues on the next page.

(ii) Outline **one** emergent behaviour that could be revealed by this simulation.

(iii) Suggest **one** strategy the health agency could test in the model to reduce infection rates and explain how you would measure its success.

Strategy:
Explain:

Space for extra work.

Indicate clearly the number and part of the question(s) you are answering.

[illegible]

Space for extra work.

Indicate clearly the number and part of the question(s) you are answering.

[illegible]

Space for extra work.

Indicate clearly the number and part of the question(s) you are answering.

[illegible]

Acknowledgements

Images

Image on page 4: <https://girlswhocode.com/>

Image on page 5: https://en.wikipedia.org/wiki/RGB_color_model

Image on page 8: <https://www.pdq.com/sysadmin-glossary/cpu/>

Image on page 9: <https://www.irishtimes.com/ireland/2024/04/23/what-is-your-experience-of-the-deposit-return-scheme/>

Image on page 13: <https://www.worldhistory.org/Euclid/>

Image on page 15: https://en.wikipedia.org/wiki/Tony_Hoare

Image on page 16: https://www.w3schools.com/dsa/dsa_timecomplexity_quicksort.php

Image on page 18: https://en.wikipedia.org/wiki/George_E._P._Box

Texts

Text on page 5: <https://pursuit.unimelb.edu.au/articles/will-a-computer-take-your-job>

Copyright notice

This examination paper may contain text or images for which the State Examinations Commission is not the copyright owner, and which may have been adapted, for the purpose of assessment, without the authors' prior consent. This examination paper has been prepared in accordance with Section 53(5) of the *Copyright and Related Rights Act, 2000*. Any subsequent use for a purpose other than the intended purpose is not authorised. The Commission does not accept liability for any infringement of third-party rights arising from unauthorised distribution or use of this examination paper.

Leaving Certificate – Higher Level

Computer Science – Sections A & B

Wednesday 21 May

Morning 9:30 – 11:00



Coimisiún na Scrúduithe Stáit
State Examinations Commission

Leaving Certificate Examination 2025

Computer Science

Section C

Higher Level

Wednesday 21 May Morning 11:30 – 12:30

80 marks

Do not hand this up.
This document will not be returned to the
State Examinations Commission.

Instructions

There is **one** section in this paper.

Section C	Programming	One question	80 marks
Answer all question parts			

Answer **all** parts of the question on your digital device.

Calculators may be used during this section of the examination.

The *Formulae and Tables* booklet cannot be used for this section of the examination.

The superintendent will give you a copy of the *Python Reference Guide*.

Ensure that you save your work regularly.

Save your files using the naming structure described at the beginning of each question part.

If you are unable to get some code to work correctly, you can comment out the code so that you can proceed. The code that has been commented out will be reviewed by the examiner.

Rough work pages are provided at the end of this booklet. Please note that this booklet is not to be handed up and will **not** be reviewed by an examiner.

At the end of the examination it is your responsibility to ensure that you have saved your files onto your external media.

You will be provided with a brown envelope for your external media. Write your examination number on this envelope and place your external media into it before sealing. Place this envelope in the pouch at the front of the red envelope that contains your examination booklet from Section A and B.

There is no examination material on this page

Answer **all** question parts.

Question 16

- (a) Open the program called **Question16_A.py** from your device. The source code is shown below and described on the next page.

Before making any changes, you should save your working copy of the file using the format **ExaminationNumberQuestion16_A.py**. For example, you would save the file as **123456Question16_A.py** if your Examination Number was 123456.

Enter your Examination Number in the space provided on **line 2** in your Python file.

```
1 # Question 16 (a)
2 # Examination Number:
3
4 def get_grade(result):
5     grade = "Unsuccessful"
6
7     if result >= 80:
8         grade = "Distinction"
9     elif result >= 65:
10         grade = "Upper Merit"
11
12     return grade
13
14 # Calculate and display the mean of a list of results
15 results = [39,32,62,88,51,62,64,81,77] # Initialise the list
16 N = len(results) # Initialise N to the number of results
17 total = 0 # Initialise the running total to 0
18
19 # Loop N times
20 for i in range(N):
21     total = total + results[i] # Running total
22
23 # Divide by the total number of results to give the mean
24 arithmetic_mean = total/9
25
26 # Display the answer
27 print("The mean percentage mark is", arithmetic_mean)
```

Line 15 of the program initialises a list called `results` with nine values. Each value represents a percentage mark obtained by an individual student in nine class tests all in the same subject. The final percentage for the subject is calculated by averaging the nine results.

The `for` loop adds up the values in `results` and stores the answer in the variable called `total`. Line 24 of the program calculates the mean (average) of all the results by dividing `total` by 9. The mean is saved in the variable called `arithmetic_mean`.

When the program is run it displays the following message:

```
The mean percentage mark is 61.77777777777778
```

Make the following changes to the program:

- (i) Round the mean percentage to two decimal places.

When the program is run the output should now look as follows:

```
The mean percentage mark is 61.78
```

- (ii) Currently, the code divides the total by 9 to calculate the mean. Modify the code so that it divides the total by the number of elements in the results list, regardless of its size.

When the program is run the output should remain the same:

```
The mean percentage mark is 61.78
```

- (iii) Complete the function `get_grade` so that it sets the variable `grade` to the correct grade using the parameter `result` and the information provided in the table below.

Result	Grade
≥ 80	Distinction
≥ 65	Upper Merit
≥ 50	Lower Merit
≥ 40	Pass
< 40	Unsuccessful

When the program is run the output should remain the same:

```
The mean percentage mark is 61.78
```

This question continues on the next page.

- (iv) Extend the program so that it calls the function `get_grade` and displays the grade in a message such as: *The grade for the average result is [grade]*. You will need to pass `arithmetic_mean` into the function.

When the program is run the output should now look as follows:

```
The mean percentage mark is 61.78
The grade for the average result is Lower Merit
```

- (v) Add code so that the program finds the lowest and highest scores in `results`. The program should display this information in two separate messages such as:

The lowest score is [lowest score]

The highest score is [highest score]

When the program is run the output should now look as follows:

```
The mean percentage mark is 61.78
The grade for the average result is Lower Merit
The lowest score is 32
The highest score is 88
```

- (vi) Add two features to count how many scores in `results` are a) less than 40 and b) between 50 and 79 inclusive. The program should display both counts in two separate messages such as:

The number of scores below 40 is [count1]

The number of scores between 50 and 79 inclusive is [count2]

When the program is run the output should now look as follows:

```
The mean percentage mark is 61.78
The grade for the average result is Lower Merit
The lowest score is 32
The highest score is 88
The number of scores below 40 is 2
The number of scores between 50 and 79 inclusive is 5
```

- (vii) Extend the program so that it determines and displays the longest run (sequence) of consecutive result increases.

Example:

Input: `results = [39, 32, 62, 88, 51, 62, 64, 81, 77]`

Output: `[51, 62, 64, 81]`

Explanation: The sub-list `[51, 62, 64, 81]` is the longest run of result increases. The value 77 breaks the sequence because it is less than its previous value, 81.

When the program is run the output should now look as follows:

```
The mean percentage mark is 61.78
The grade for the average result is Lower Merit
The lowest score is 32
The highest score is 88
The number of scores below 40 is 2
The number of scores between 50 and 79 inclusive is 5
The longest run of result increases is [51, 62, 64, 81]
```

Save your file using the format **ExaminationNumberQuestion16_A.py**. For example, you would save the file as **123456Question16_A.py** if your Examination Number was 123456.

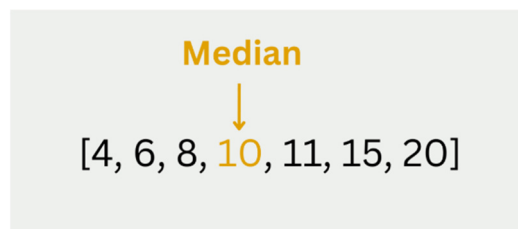
This question continues on the next page.

- (b) Open the program called **Question16_B.py** from your device. This file contains only two comments, on lines 1 and 2.

Before adding any code, you should save your working copy of the file using the format **ExaminationNumberQuestion16_B.py**.

For example, you would save the file as **123456Question16_B.py** if your Examination Number was 123456.

Enter your Examination Number in the space provided on **line 2** in your Python file.



The median of a list of values is the middle value in that list after it has been sorted in either ascending or descending order. If the number of values in the list is odd the median is the middle value. However, if the number of values in the list is even the median is found by calculating the mean of the two middle values i.e. by adding both middle values and then dividing the result by two.

Write a Python program to find the median of a list of zero or more values.

You should use comments throughout your program to explain your code. You may wish to reuse some of the code you used in **part (a)** as part of your solution.

Your program should meet the following requirements:

- Initialise a list of integers.
- Display the list.
- Sort the list.
- Display the sorted list.
- Determine the median by examining the list length.
 - **Odd:** The number of elements divided by 2 will have a remainder of 1. In this case the median is the element at the middle position of the sorted list.
 - **Even:** The number of elements divided by 2 will have no remainder. In this case the median is the mean of the two middle elements in the sorted list.
- Display the median.
- Display an error message if the list is empty.

Note:

1. You may **not** make use of the `statistics.median()`, `numpy.median()` or any other Python library median function in your solution. You must write the code to calculate the median yourself.
2. You should test that your program works with an even and odd number of values.
3. You should test that your program works with an empty list.

Example outputs are shown on the next page.

Sample output 1:

```
The initial list of values is: [27, 13, 32, 50, 16]
The sorted list of values is: [13, 16, 27, 32, 50]

The median is 27
```

Sample output 2:

```
The initial list of values is: [27, 13, 32, 50, 16, 29]
The sorted list of values is: [13, 16, 27, 29, 32, 50]

The median is 28.0
```

Sample output 3:

```
The initial list of values is: []

The list is empty. Cannot compute the median.
```

Use the format **CandidateNumberQuestion16_B.py** to save your file. For example, you would save the file as **123456Question16_B.py** if your candidate number was 123456.

Space for rough work.

This page will not be reviewed by an examiner.

Space for rough work.

This page will not be reviewed by an examiner.

Do not hand this up.

This document will not be returned to the
State Examinations Commission.

Copyright notice

This examination paper may contain text or images for which the State Examinations Commission is not the copyright owner, and which may have been adapted, for the purpose of assessment, without the authors' prior consent. This examination paper has been prepared in accordance with Section 53(5) of the *Copyright and Related Rights Act, 2000*. Any subsequent use for a purpose other than the intended purpose is not authorised. The Commission does not accept liability for any infringement of third-party rights arising from unauthorised distribution or use of this examination paper.

Leaving Certificate – Higher Level

Computer Science – Section C

Wednesday 21 May

Morning 11:30 – 12:30