



Coimisiún na Scrúduithe Stáit
State Examinations Commission

Leaving Certificate Examination
Sample Paper

Computer Science

Sections A & B

Higher Level

Time: 1 hour, 30 minutes

130 marks

Examination number					

Centre stamp

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	60 marks	12 questions
Section B	Long Questions	70 marks	3 questions
Section C	Programming	80 marks	1 question

Answer all questions.

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.

This examination booklet will be scanned and your work will be presented to an examiner on screen. Anything that you write outside of the answer areas may not be seen by the examiner.

Answer all twelve questions.

Question 1

For each of the JavaScript Boolean expressions listed below state whether they evaluate to **true** or **false**.

Boolean Expression	Result
<code>7 > 5</code>	
<code>(7 > 5) && false</code>	
<code>(7 == 5) true</code>	
<code>(7 != 5) false</code>	
<code>(!false) && (!true)</code>	

Question 2

(a) Add the following two binary numbers:

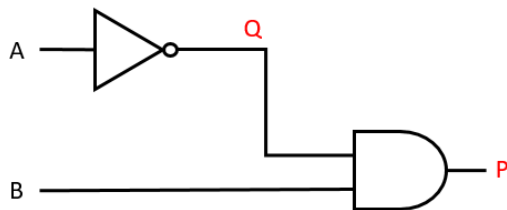
<div>01001010 10011001 <hr/></div>
--

(b) Convert your answer to decimal (base 10).

--

Question 3

Complete the truth table below from the logic diagram shown. The first row has already been completed.



A	B	Q = NOT A	P = Q AND B
0	0	1	0
0	1		
1	0		
1	1		

Question 4

Abstraction is a key pillar of computational thinking and programming.

- (a) Define what is meant by the term **abstraction** in the context of computational thinking.

- (b) Explain how functions can be used by programmers to achieve abstraction.

Question 5

This JavaScript code shows the definition of a recursive function called **mystery**. The final line of code displays the output returned when **mystery** is called with an argument of 4.

```
1 function mystery(n) {  
2  
3     if (n == 0)  
4         return 1;  
5  
6     return n * mystery(n-1);  
7 }  
8  
9 alert(mystery(4));
```

(a) Which of the following values is displayed when the program is run? (Tick **one** option.)

4 ☐

1 ☐

24 ☐

12 ☐

(b) What makes the function **mystery** a recursive function?

Question 6

The JavaScript code below shows the definition of a function called **maxNum**. The function returns the largest of three integers **x**, **y** and **z**.

The final line of code displays the output returned when **maxNum** is called using the three arguments 2, 4 and 1. The call results in 4 being displayed in an alert pop-up box.

```
1 function maxNum(x, y, z) {  
2     if ((x>y) && (x>z))  
3         return x;  
4     else if ((y>x) && (y>z))  
5         return y;  
6     else  
7         return z;  
8 }  
9  
10 alert(maxNum(2,4,1));
```

- (a) The three arguments 2, 4 and 1 constitute a test case designed specifically to trigger the statement, **return y**. Suggest **two** further test cases that could be used to test **maxNum** as follows:

(i) A test case designed to trigger return x
(ii) A test case designed to trigger return z

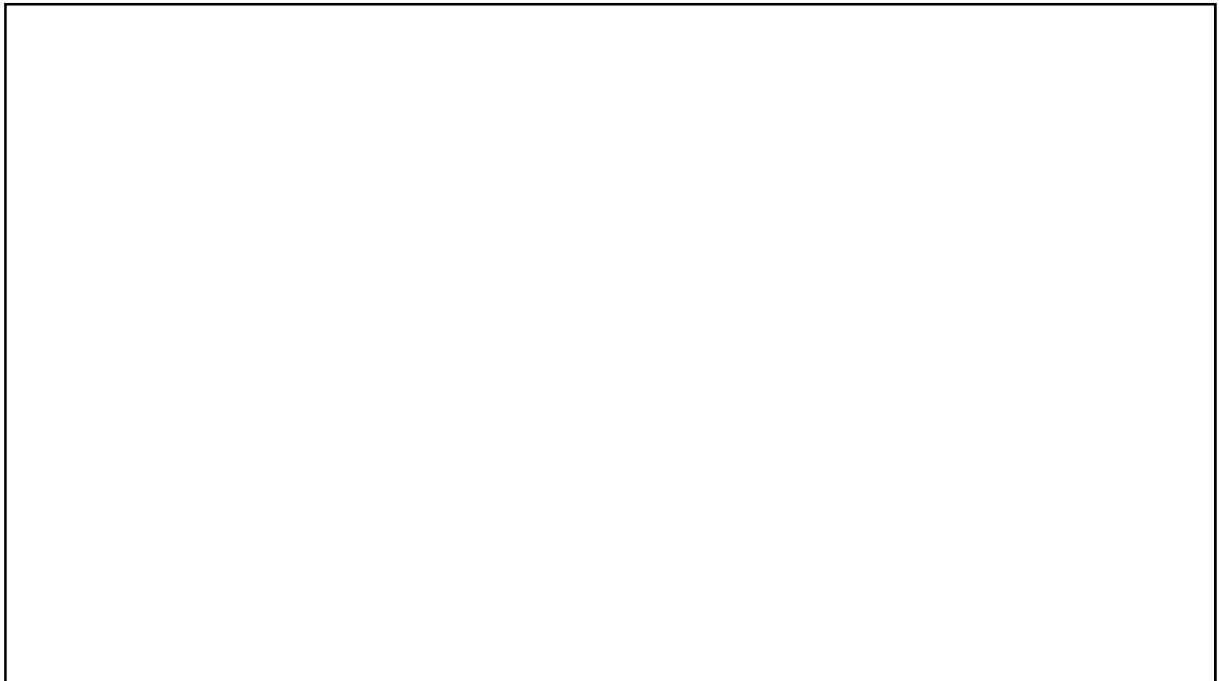
- (b) Explain why this solution is not suitable if we wish to find the maximum of a larger number of arguments.

Question 7

The intention of the HTML shown below is to gather information from an end-user.

```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <h1>Survey</h1>
5     <form action="">
6       <label>Name:</label>
7       <input type="text" value="Enter name">
8       <fieldset>
9         <legend>Do you wear glasses?</legend>
10        <input type="radio">Yes
11        <input type="radio" checked>No
12      </fieldset>
13      <input type="submit" value="Back">
14      <input type="submit" value="Next">
15    </form>
16    <p>Thank you for taking part.</p>
17  </body>
18 </html>
```

Sketch a picture of how the HTML form might look in a typical web browser.



Question 8

Fit Don't Quit is the name of a fitness club that uses a computer system to store information about its members.

- (a) State the most suitable data type for each of the following **three** items:

(i) Member Identification (e.g. 1754):	
(ii) Member Name (e.g. Mary Murphy):	
(iii) Amount Paid (e.g. 45.50):	

- (b) Suggest the name and purpose of another useful variable for the Fit Don't Quit computer system to capture information about its members. It should be a Boolean variable.

Name:
Purpose:

Question 9

- (a) Define what is meant by the term **relational database**.

- (b) Explain the purpose of a **primary key** in a relational database.

Question 10

On 15 May 2019, robotic engineers from Trinity College Dublin unveiled Stevie II, Ireland's first socially assistive robot.



- (a) Identify **two** ways in which robots such as Stevie II could have a positive impact on the lives of elderly people.

1.
2.

- (b) Outline **one** ethical concern that relates to socially assistive robots such as Stevie II.

Question 11

Algorithms, software and smart technologies have a growing presence in cities around the world. Artificial intelligence (AI), agent-based modelling, the internet of things and machine learning can be found practically everywhere now.

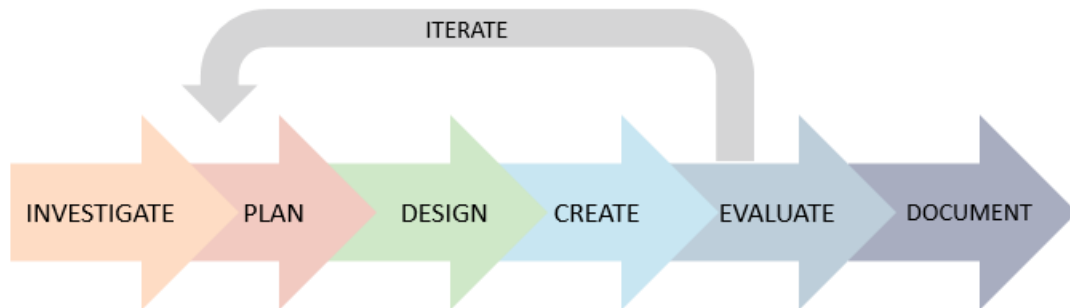
www.govtech.com, Silvio Carta, July 2019

- (a) State **one** example of agent-based modelling.

- (b) Explain how agent-based modelling can be used to demonstrate emergent behaviours.

Question 12

The diagram below illustrates the typical stages of a software development design process. Outline briefly the main purpose of the (i) **design** and (ii) **create** stages.



Design:
Create:

Answer all three questions.

Question 13

Not to be confused with the internet, which had been evolving since the 1960s, the World Wide Web celebrated its 30th anniversary on 12 March, 2019.

Since 1991 external web servers have been up and running and today it is estimated that there are over 2 billion websites online.

While recognising the power of the web to transform governments, businesses, and societies for the better, inventor of the web, Sir Tim Berners-Lee used the occasion to highlight the abuse of his creation.

“All kinds of things have gone wrong. We have fake news, we have problems with privacy, we have people being profiled and manipulated.” He called for a ‘contract’ to reclaim the web as a force for the common good.



(Adapted from Irish Independent Editorial, May 16th 2019)

(a) Distinguish between the World Wide Web and the Internet.

(b) Explain each of the following terms:

- (i) HTML
- (ii) Web browser
- (iii) URL
- (iv) HTTP




HTML:

Web browser:

URL:

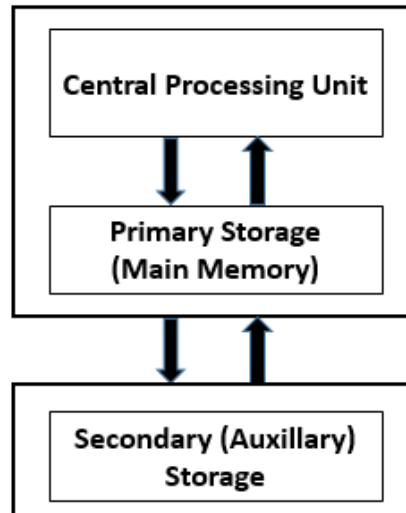
HTTP:

- "The web has evolved into an engine of inequity and division"
- Sir Tim Berners-Lee
- 
- A close-up portrait of Sir Tim Berners-Lee, looking directly at the camera with a serious expression. He has short, light-colored hair and is wearing a dark jacket over a red shirt. The background is dark and out of focus.

[illegible]

Question 14

- (a) This diagram depicts a high-level view of some of the main components of a computer system.



- (i) What is the purpose of the Central Processing Unit?

- (ii) List **two** types of primary storage.

1.
2.

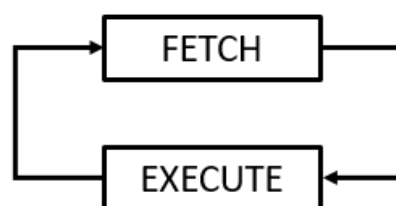
- (iii) List **two** types of secondary storage.

1.
2.

(iv) State **two** key differences between primary and secondary storage.

1.
2.

(b) Explain the cycle of operations that are carried out as instructions are fetched and executed from a computer's memory.



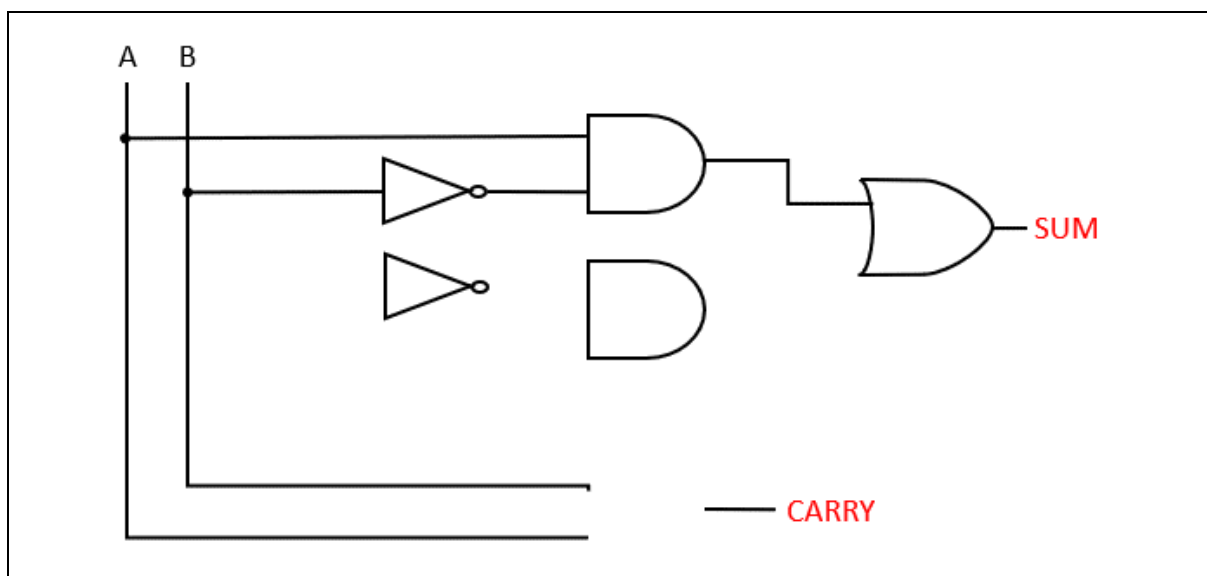
Your explanation should refer to the role of registers.

- (c) The **SUM** and **CARRY** resulting from the addition of two binary variables, A and B, are described by the following two Boolean expressions:

$$\text{SUM} = (A \text{ AND NOT } B) \text{ OR } (\text{NOT } A \text{ AND } B)$$

$$\text{CARRY} = A \text{ AND } B$$

- (i) Use the information provided to complete the drawing of the logic circuit shown.



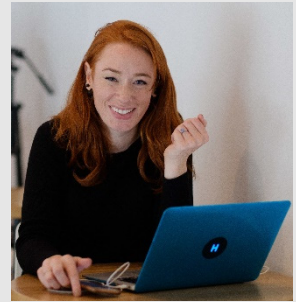
- (ii) Explain how 0+1 is computed to have a SUM of 1 and a CARRY of 0.

SUM:
CARRY:

Question 15

Algorithms have given the world everything from social media feeds to search engines, satellite navigation to music recommendation systems, and are as much a part of our modern infrastructure as bridges, buildings and factories ever were.

Adapted from: *Hello World: Being Human in the Age of Algorithms* by Hannah Fry



(a) Define the term **algorithm**.

(b) Outline briefly **one** example of how algorithms have played a role in the following areas:

(i) Travel

(ii) Retail

Travel:
Retail:

- (c) The data below depicts the steps taken by a certain sorting algorithm to sort the list of integers: 88, 46, 25, 11, 18, 11 and 22.

Study the data and answer the questions that follow.

```
INPUT (initial list):  [88, 46, 25, 11, 18, 11, 22]
Pass 1: [46, 25, 11, 18, 11, 22, 88]
Pass 2: [25, 11, 18, 11, 22, 46, 88]
Pass 3: [11, 18, 11, 22, 25, 46, 88]
Pass 4: [11, 11, 18, 22, 25, 46, 88]
Pass 5: [11, 11, 18, 22, 25, 46, 88]
Pass 6: [11, 11, 18, 22, 25, 46, 88]
Pass 7: [11, 11, 18, 22, 25, 46, 88]
OUTPUT (sorted list): [11, 11, 18, 22, 25, 46, 88]
```

- (i) State the name of the algorithm.

--

- (ii) Explain how the algorithm works.

- (iii) Referring to passes 5, 6 and 7 identify **one** limitation of the algorithm.

(iv) Suggest how this limitation could be overcome.

(d) Quicksort is an efficient sorting algorithm, developed in 1959, which is still commonly used today.

(i) Using 50 as the initial pivot perform a quicksort on the following list of integers. In your answer you should show the state of the list after each pass.

85 24 63 45 17 31 96 50

--

(ii) Explain why the worst-case time complexity for the quicksort algorithm is $O(n^2)$.

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]

There is no examination material on this page

There is no examination material on this page

There is no examination material on this page

Acknowledgements

Images

Image on page 9: www.irishtimes.com/news/ireland/irish-news/robot-caregiver-with-cute-accent-gets-rave-reviews-1.3893612

Image on page 12: www.mikestrongphoto.com/CV_Galleries/LessonExamples/MultiMedia/TimBernersLee.htm

Image on page 14: itsfoss.com/solid-decentralized-web/

Image on page 18: en.wikipedia.org/wiki/Hannah_Fry

Texts

Document on page 10: www.govtech.com/products/I-Visualized-How-Algorithms-See-Urban-Environments-and-Build-Detailed-Profiles-of-Citizens.html

Document on page 12: www.independent.ie/opinion/editorial/editorial-we-must-reclaim-internet-as-force-for-common-good-38116862.html

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

Sample Paper

Time: 1 hours, 30 minutes



Coimisiún na Scrúduithe Stáit
State Examinations Commission

Leaving Certificate Examination
Sample Paper

Computer Science

Section C

Higher Level

Time: 1 hour

80 marks

Examination number					

Centre stamp

Instructions

There is one section of the examination paper in this booklet.

Section C	Programming	80 marks	1 question
-----------	-------------	----------	------------

Answer all parts of the question on your digital device.

Instructions are provided for each question.

Ensure that you save your work regularly and when you complete each question.

Do not change the file names or save your work under different file names.

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.

Answer all question parts.

Question 16

According to a recent report in the Irish Independent (19 August 2019) one in four children and three out of four adults in Ireland are overweight or obese. This causes significant increases in illnesses such as diabetes and heart disease. To determine whether a person is overweight or obese, a measure called the Body Mass index (BMI) can be used.

BMI is defined as body weight (in kilograms) divided by the square of the body height (in metres). If height is given instead in centimetres, then the formula for calculating BMI is:

$$BMI = \frac{weight}{height^2} \times 10000$$

- (a) Open the program called **Question16_A.py** from your device.
Enter your Examination Number in the space provided on **Line 2**.

```
1 # Question 16(a)
2 # Examination Number:
3
4 weight = int(input("Enter weight (in kilograms): "))
5 height = 180 # centimetres
6
7 bmi = weight / (height * height) * 10000
8
9 print("BMI is: ", bmi)
10
```

This program is designed to calculate and display the BMI for a person whose weight is entered by the end-user and whose height is 180 cm.

A sample run of the program is displayed below – the user enters a value of 90 (kilograms) for **weight** and the program displays the resulting BMI.

```
Enter weight (in kilograms): 90
BMI is: 27.777777777777778
```

Modify the program to do the following:

- (i) Insert a comment to say '*read weight*' in the appropriate location in the program to show where the weight is input.
- (ii) Currently in the program the value of the variable **height** is hard-coded to 180. Modify the program so that it prompts the user to enter a value for **height**. The value should be converted to an integer.

When the program is run the output may look as follows:

```
Enter weight (in kilograms): 90
Enter height (in centimetres): 180
BMI is: 27.77777777777778
```

- (iii) By using the function **round**, or otherwise, modify the program so that the value of the BMI displayed is rounded to one decimal place.

When the program is run the output may look as follows:

```
Enter weight (in kilograms): 90
Enter height (in centimetres): 180
BMI is: 27.8
```

- (iv) By using the function **pow**, or otherwise, replace the expression (**height * height**) with an alternative, but equivalent, implementation.
- (v) Incorporate the following function definition into your program and insert a line so that the function is called before the user enters any data.

```
def display_intro():
    print("Welcome to my BMI calculator!")
```

When the program is run the output may look as follows:

```
Welcome to my BMI calculator!
Enter weight (in kilograms): 90
Enter height (in centimetres): 180
BMI is: 27.8
```


- (vi) A BMI number alone might not mean much to the average person. A more meaningful output might be one of the four BMI categories: underweight, normal weight, overweight or obese.

Extend the program so that it displays the BMI category according to the BMI ranges in the table below.

BMI Category	BMI Range
Underweight	less than 18.5
Normal	from 18.5 to 24.9
Overweight	from 25 to 29.9
Obese	greater than or equal to 30

The table below shows the expected BMI values and categories for some sample weight and height values. You could use this data to test your program.

weight (<i>kgs</i>)	height (<i>cms</i>)	bmi (<i>kg/cm²</i>)	BMI Category
50	165	18.3	Underweight
70	170	24.2	Normal
90	180	27.8	Overweight
90	170	31.1	Obese

When the program is run the output may look as follows:

```
Welcome to my BMI calculator!
Enter weight (in kilograms): 78
Enter height (in centimetres): 180
BMI is: 24.1
Normal
```

Save and close your file before moving on to the next part.

- (b) Open the program called **Question16_B.py** from your device.
Enter your Examination Number in the space provided on **Line 2**.

```
1 # Question 16(b)
2 # Examination Number:
3
4 # For this question it is useful to understand ...
5 # 1. randint(a, b) returns a random integer N such that a<=N<=b.
6 # 2. s.append(x) appends the element x to the end of list s.
7
8 from random import *
9
10 heights = [] # an empty list of heights
11 weights = [] # an empty list of weights
12
13 # Loop to build up the lists with random values
14 for count in range(10):
15     # a random integer between 150 and 210
16     heights.append(randint(150, 210))
17     # a random integer between 50 and 130
18     weights.append(randint(50, 130))
19
20 # Display the lists
```

The program creates two empty lists – **heights** and **weights**.

It then uses a **for** loop to populate both lists with 10 random values as follows:

- each element in **heights** is a random integer between 150 and 210 (cm)
- each element in **weights** is a random integer between 50 and 130 (kg)

Modify the program to do the following:

- (i) Display the contents of both lists, along with the labels **Heights** and **Weights**.

When the program is run the output may look as follows:

```
Heights: [202, 180, 200, 152, 159, 165, 206, 208, 157, 154]
Weights: [87, 118, 106, 53, 96, 106, 87, 121, 129, 95]
```

- (ii) Currently the program generates 10 pairs of values every time it is run. Modify the program so that it generates a variable number of values entered by the user.

When the program is run the output may look as follows:

```
Enter the number of pairs of values you wish to generate: 7
Heights: [151, 198, 168, 197, 154, 197, 166]
Weights: [120, 95, 111, 89, 72, 87, 115]
```

- (iii) Extend the program to create and populate (using a separate loop) a third list called **bmi_values**.

Each element of **bmi_values** should be calculated using the corresponding values from **heights** and **weights** (that is **bmi_values[i]** should be calculated using **heights[i]** and **weights[i]**).

You may wish to use the code from **part (a)** to calculate the BMI value.

The BMI values should be rounded to one decimal place.

When the program is run the output may look as follows:

```
Enter the number of pairs of values you wish to generate: 7
Heights: [178, 199, 160, 205, 203, 153, 166]
Weights: [106, 119, 59, 53, 92, 109, 105]
BMI values: [33.5, 30.0, 23.0, 12.6, 22.3, 46.6, 38.1]
```

Save and close your file before moving on to the next part.

- (c) Open the program called **Question16_C.py** from your device.
Enter your Examination Number in the space provided on **Line 2**.

```
1 # Question 16(c)
2 # Examination Number:
3
4 bmi_values = [24, 19, 33, 35, 27, 18, 15, 33, 35, 23, 32, 23]
```

The program initialises a list called **bmi_values** with 12 integer values.

Write a Python program to do the following:

- (i) Determine and display the number of obese values in the list.
(Recall from **part (a)** that if the BMI is greater than or equal to 30 then it is categorised as obese.)

When the program is run the output may look as follows:

```
Obese: 5
```

- (ii) Determine and display the largest and second largest values in the list.
(Note that the list contains values that are repeated.)

When the program is run the output may look as follows:

```
Largest: 35
Second Largest: 33
```

- (iii) Implement a general function that returns the n^{th} largest value from the list.

Name the function **find_nth_largest** as shown in the function definition header:

```
def find_nth_largest(n, list_of_values):
```

The function accepts two parameters, **n** and **list_of_values**. The function body should be coded to determine and return the n^{th} largest value in the list.

An example of the function being used to determine the 3rd largest value of the list is shown.

```
print(find_nth_largest(3, bmi_values))
```

When the program is run the output may look as follows:

```
32
```

Save your file.

Ensure that you have saved and closed all files before you finish the examination.

There is no examination material on this page

There is no examination material on this page

There is no examination material on this page

Acknowledgements

None

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

Sample Paper

Time: 1 hour