



Pre-Leaving Certificate Examination, 2025

Computer Science

Section C

Higher Level

Time: 1 hour

80 marks

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 work will **not** be reviewed by an examiner.

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

Answer all question parts.

Question 16

- (a) Open the program called **Question16_A_HL.py** from your device. The source code is shown below.

Enter your name in the space provided on **line 2** in your Python file. The program initialises a variable called "s" and assigns it a value of 0 in Line 4 of the code. The user is asked to enter a string input to be stored in the variable "sentence".

```
1 #Question_16_A_HL
2 #Enter your name here:
3
4 s = 0
5
6 sentence = input("Please enter a sentence: ")
```

Make the following changes to the program:

- (i) Modify the code to count the number of times the character "S" or "s" appears in the sentence entered by the user. When the code is run, it may look as follows:

```
Please enter a sentence: She sells sea shells by the sea shore
The number of s's was: 8
```

- (ii) Modify the code to count the number of vowels in the sentence entered by the user. In this part, the sentence should be converted to lowercase. When the code is run, it may look as follows:

```
Please enter a sentence: She sells sea shells by the sea shore
Your sentence was: She sells sea shells by the sea shore
Your lowercase sentence is: she sells sea shells by the sea shore
The number of vowels was: 10
```

- (iii) Modify the code so that the program will count the number of letters and digits that appear in the user's sentence. When the code is run, it may look as follows:

```
Please enter a sentence: Hello World 123
The number of letters was: 10
The number of digits was: 3
```

- (iv) Modify the code so that the program will count the number of words in a sentence. When the program is run, it may look as follows:

```
Please enter a sentence: Computer Science is the best subject
The number of words was: 6
```

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

- (b) In many supermarkets, customers have the opportunity to “Scan as you shop”. This involves a customer using a handheld device (as seen in the image) to scan the items as they go around the shop.

The price for each item appears on the screen, the total cost and the number of items already scanned are displayed on the scanner.



Open the program called **Question16_B_HL.py** from your device. The source code is shown below. A function called `shopping()` is defined, a variable called `total` is initialised to 0 and two empty lists `item_list` and `item_prices` are also created.

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

```
1 #Question_16_B_HL
2 #Enter your name here:
3
4 import random
5 print("Welcome to my shop")
6 total = 0
7 item_list = []
8 item_prices = []
```

- (i) Modify the code that will ask the user to enter a shopping item as a string. It should store the item names in `item_list`. It should also ask the user to enter the price of the item as a float. The price should be stored in the list `item_prices`.

The “running total” should be displayed, that is, the current total cost of the shopping while you shop.

If the user enters “stop” in the item list the full list of items should be printed to the screen and the grand total should also be displayed as you are now telling the program you are ready to checkout your shopping at the till. Sample output is shown below:

```
Welcome to my shop
Please enter the item: Apple
Please enter the price of the item: 1.5
The current total is €1.5
Please enter the item: Pear
Please enter the price of the item: 1.3
The current total is €2.8
Please enter the item: Orange
Please enter the price of the item: 2
The current total is €4.8
Please enter the item: Kiwi
Please enter the price of the item: 0.9
The current total is €5.7
Please enter the item: Stop
Your items are: ['Apple', 'Pear', 'Orange', 'Kiwi']
The prices are: [1.5, 1.3, 2.0, 0.9]
Grand total € 5.7
```

- (ii) When the customer goes to the scan as you shop checkout (after they have entered “stop”), they may be asked to call a member of staff to check random items in their trolley to make sure the customer scanned everything correctly. Modify your code so that it will now print 1 random item from that items list for the checkout assistant to check.

Sample output is shown below:

```
Welcome to my shop
Please enter the item: Apple
Please enter the price of the item: 1.5
The current total is €1.5
Please enter the item: Pear
Please enter the price of the item: 1.3
The current total is €2.8
Please enter the item: Orange
Please enter the price of the item: 2
The current total is €4.8
Please enter the item: Kiwi
Please enter the price of the item: 0.9
The current total is €5.7
Please enter the item: Stop
Your items are: ['Apple', 'Pear', 'Orange', 'Kiwi']
The prices are: [1.5, 1.3, 2.0, 0.9]
Grand total € 5.7
Your random item to be checked is: Apple
```

- (iii) From the shopping list above, modify your code so that it will find the most expensive and cheapest item in the item list. Sample output is shown below:

```
Welcome to my shop
Please enter the item: Apple
Please enter the price of the item: 1.5
The current total is €1.5
Please enter the item: Pear
Please enter the price of the item: 1.3
The current total is €2.8
Please enter the item: Orange
Please enter the price of the item: 2
The current total is €4.8
Please enter the item: Kiwi
Please enter the price of the item: 0.9
The current total is €5.7
Please enter the item: Stop
Your items are: ['Apple', 'Pear', 'Orange', 'Kiwi']
The prices are: [1.5, 1.3, 2.0, 0.9]
Grand total € 5.7
Your random item to be checked is: Kiwi
The most expensive item is: Orange
The cheapest item is: Kiwi
```

Save your file.

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

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.



P 5 2

Blank Page