P 5 1

Pre-Leaving Certificate Examination, 2023

Computer Science

Sections A & B

Higher Level

Time: 1 hour, 30 minutes

210 marks

Name:
School:
Address:
Class:
Teacher:

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	45 marks
Section B	Long Questions	Attempt any two questions All questions carry equal marks	78 marks
Section C	Programming	Answer all question parts	87 marks

Calculators may **not** be used during this section.

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.

Section A Short Answer Questions 45 marks

Answer any nine questions.

Question 1

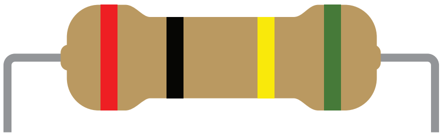
Ohm's Law states that a current flowing through a conductor between two points is directly proportional to the voltage applied across the two points.

In relation to electronics, answer the following questions:

(a) In terms of an electrical circuit, what is meant by the term voltage?

(b) In terms of an electrical circuit, what is meant by the term current?

(c) Identify the electronic device in the image below:



Question 2

HTTP and URL are key mechanisms used by web browsers when using the world wide web.

Explain the underlined terms in the statement above.

(a) HTTP:

(b) URL:

Question 3

(a) Convert the following binary number into hexadecimal.

$\mathbf{10110101_2}$

Please show all your workings.

(b) Convert the following decimal number into hexadecimal.

$\mathbf{79_{10}}$

Please show all your workings.

Question 4

Some Python code is shown below. Examine the code carefully and answer the question that follows.

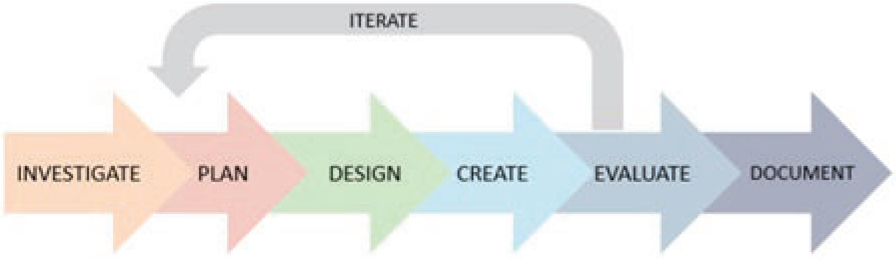
```
import random
import datetime

def func():
      message = "Welcome to my program"
      print (message)
      print("This program will show you the basics of Python!")
      x = datetime.datetime (2020,6,13)
      print (x)
      varl = True
      while varl:
            print("Hello")
            varl = False
      numl = random.randint (2,6)
      num2 = 4.2
      print("This is how you print a variable: ", num2)
      num3 = 2 + 3j
      print (num3)
func()
```

Identify **variables** for the following datatypes and fill in the table below with the appropriate variable names.

Datatype	Variable name
String	
Int	
Boolean	
Date	
Float	

Question 5



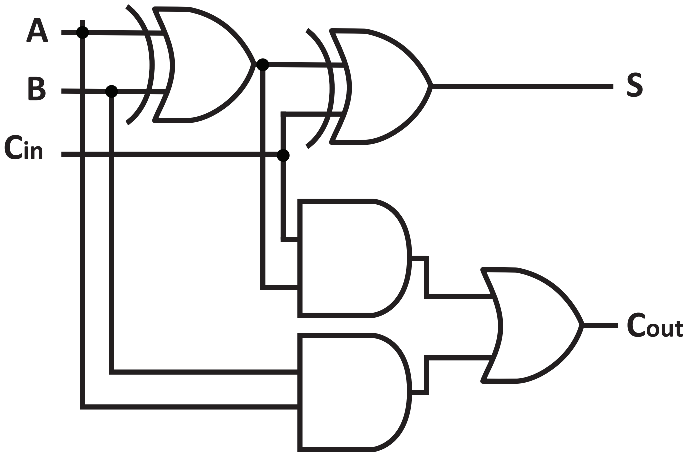
The diagram above identifies some of the main stages of a **software development design process.** Describe briefly what happens at the following stages of the design process:

(a) Plan stage

(b) Iterate stage

Question 6

The diagram below shows a **Full-Adder**. This is an important component of computer circuitry. It is composed of several logic gates combined.



(a) What is the function of the Full-Adder?

(b) Using an appropriate label, label an OR logic gate on the above diagram.

(c) Complete the truth table for an OR logic gate.

Input A	Input B	Output
0	0	
1	0	
0	1	
1	1	

Question 7

A coloured pixel is shown below with its associated hexadecimal and binary code:

Hexadecimal: EC8A33_{16}
Binary: $111011001000101000110011_2$

Suggest **one** reason why it is advantageous to store information about pixel colour in a binary form.

Question 8

(a) Many types of sorting algorithms exist. Each type has their own advantages and disadvantages. One such sorting algorithm is **Bubble Sort**. Perform a Bubble Sort on the following array of numbers, showing what the array looks like after each iteration:
(You do not need to use all of the given rows.)

9	**4**	**1**	**8**	**6**

(b) What is the time complexity for the **worst-case** scenario for the above algorithm?

Question 9

(a) In relation to computers and webpages, what is your understanding of "Universal Design"?

(b) Suppose you were tasked to build a website for a government department. Name **two** ways that you could make it accessible to someone with little experience of computers and/or the World Wide Web.

1.

2.

Question 10

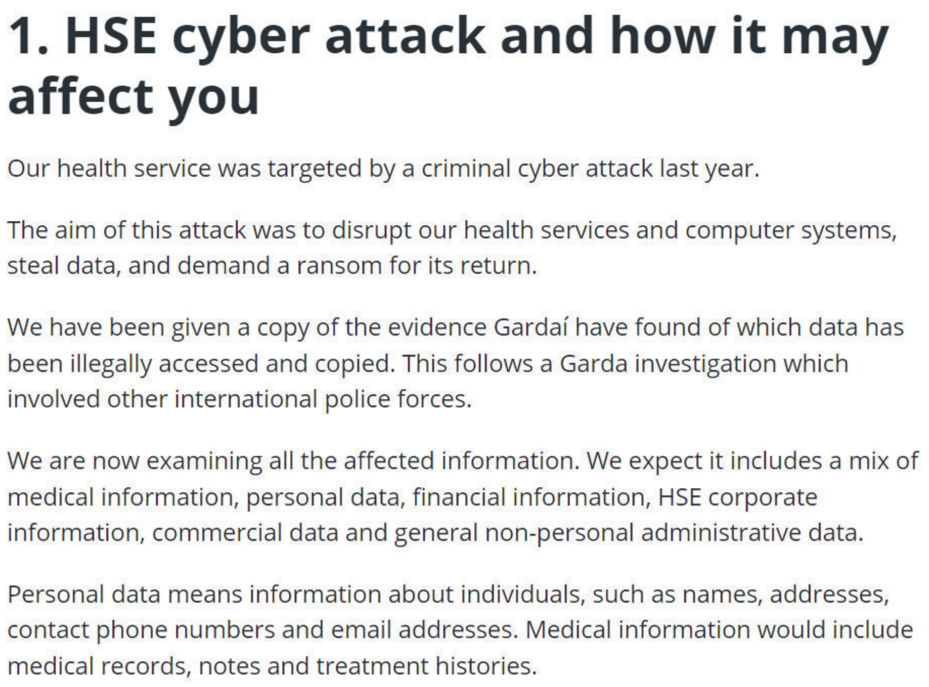
In 1958, Jack Kilby and Robert Noyce independently invented the first Integrated Circuit, or "Microchip".

MARSHALL ISLANDS
1950• DECADE OF PERIL AND PROGRESS •1959
THE 20TH CENTURY
60
MICROCHIP PRESAGES COMPUTER REVOLUTION

Pick **one** other significant historical event in the history of computing and briefly describe it. You should include roughly when it was invented and the impact it had on computing both at the time and subsequent computing technologies.

Question 11

On the 14th of May 2021, The Health Service Executive (HSE) was subjected to a cyber attack. Read the extract below and answer the questions that follow.



1. HSE cyber attack and how it may affect you

Our health service was targeted by a criminal cyber attack last year.

The aim of this attack was to disrupt our health services and computer systems, steal data, and demand a ransom for its return.

We have been given a copy of the evidence Gardaí have found of which data has been illegally accessed and copied. This follows a Garda investigation which involved other international police forces.

We are now examining all the affected information. We expect it includes a mix of medical information, personal data, financial information, HSE corporate information, commercial data and general non-personal administrative data.

Personal data means information about individuals, such as names, addresses, contact phone numbers and email addresses. Medical information would include medical records, notes and treatment histories.

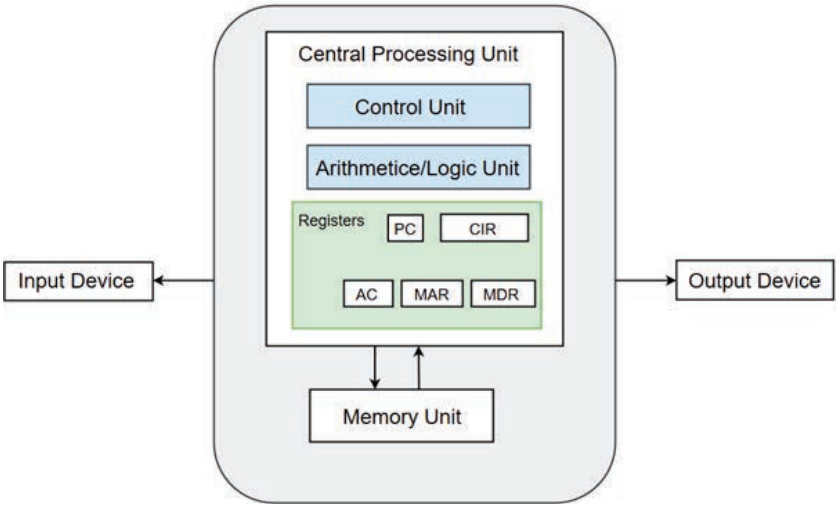
Suggest **two** ways in which the unauthorised release of the stolen data could impact the individuals involved.

1.
2

Question 12

The Von Neumann Architecture and an associated memory unit is shown below:

Von-Neumann Basic Structure:



(a) What is the function of the Arithmetic Logic Unit (ALU)?

(b) Briefly describe the Fetch – Decode – Execute cycle.

Section B Long Questions 78 marks

Answer any **two** of the three questions.

Question 13

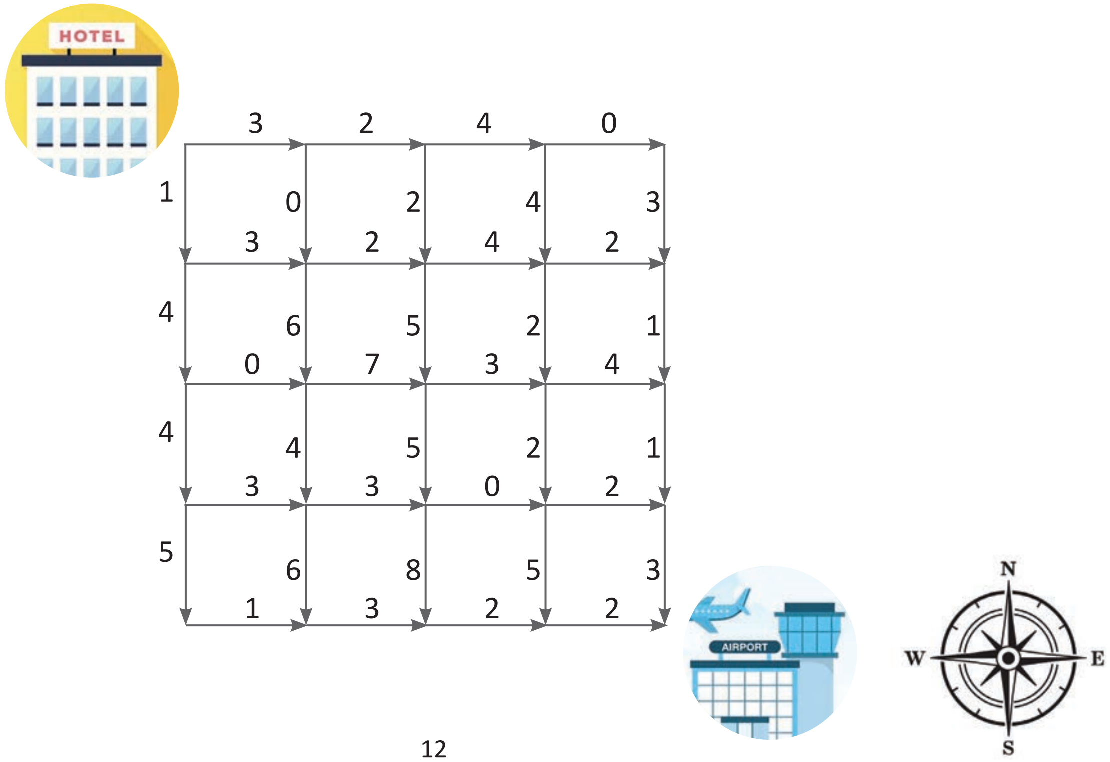
(a) Decomposition and Algorithms are two important aspects of Computational Thinking. Explain the underlined terms.

Decomposition
Algorithms

(b) The Manhattan Tourist Problem is a famous computer science problem.
You are visiting Manhattan in New York City, but you have left all your sightseeing until the trip to the airport!
The object of the challenge is to see as many attractions as possible from your hotel to the airport while travelling south and east only. The numbers along each arrow show the number of attractions on each street.

Complete the given Manhattan Tourist Problem below. Please show your workings.

One (non-optimal) solution to the problem would be: South 1+South 4+South 4+ South 5+ East 1+East 3+East 2+East 2 = 22 attractions visited.



(c) Relational Databases are very useful for large organisations to keep track of their customer information.

(i) Explain what is meant by the underlined term in the above statement.

(ii) Name a **primary key** a small Manhattan tourism company could use in their relational database.

Question 14

(a) In relation to computer algorithms, what is meant by a **heuristic algorithm**?

(b) A famous problem that can be solved with Heuristics is the Knapsack Problem. A simple Knapsack Problem can be solved by hand. In the Knapsack Problem, you need to pack a set of items into a bag. Each item has a weight (in kg) and value (in euro).

The problem is that you can only carry a maximum of 13kg.
You need to choose a combination of the items to maximise the total value (in euro) that you can carry. You can only choose one of each item. You do not necessarily need to reach the limit of 13kg to find the optimal value.

The items, weights and values are given below:
Gold Bar (weight 5kg; value €10)
Diamond (weight 3kg, value €20)
Ruby (weight 8kg; value €25)
Coin (weight 4kg, value €8)

Solve the Knapsack Problem with the given bag and items below. You must show that the answer you obtain is the optimal solution for this set of constraints.

Weight = 5 Kg
Value = €10

Weight = 3 Kg
Value = €20

Weight = 8 Kg
Value = €25

Weight = 4 Kg
Value = €8

Maximum Weight = 13 kg

(c) Give **two** examples of how heuristic algorithms can be used in real-life situations.

Question 15

(a) Many simple electronic components can be combined to create complex circuitry.
State the function of **each** of the following electronic components:

Resistor:

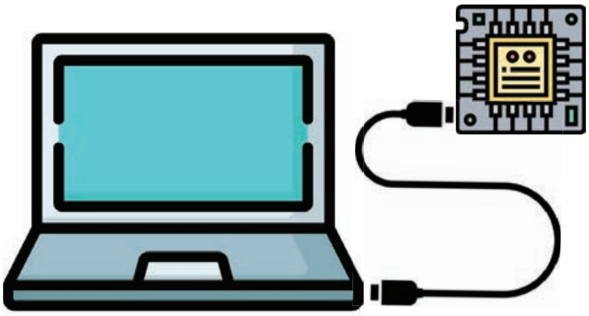
Capacitor:

Transistor:

(b) Examples of microcontrollers/microcomputers used to make Embedded Systems that you may have used during your Computer Science Applied Learning Tasks (ALTs) are the Raspberry Pi, Arduino and the Micro: Bit.

These embedded systems can be connected to computers to store data in a database.

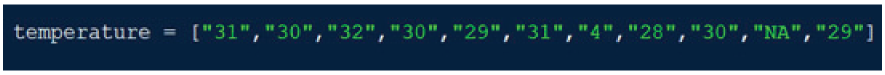
The diagram below shows a Micro: Bit microcontroller connected to a laptop via a USB cable. The Micro: Bit is using its built-in thermometer to measure the temperature of the air every 5 seconds for 1 minute. These values are being recorded by a Python file.



(i) State **one** difference between digital and analogue input.

(ii) Is temperature an example of continuous or discrete data? Give a reason for your answer.

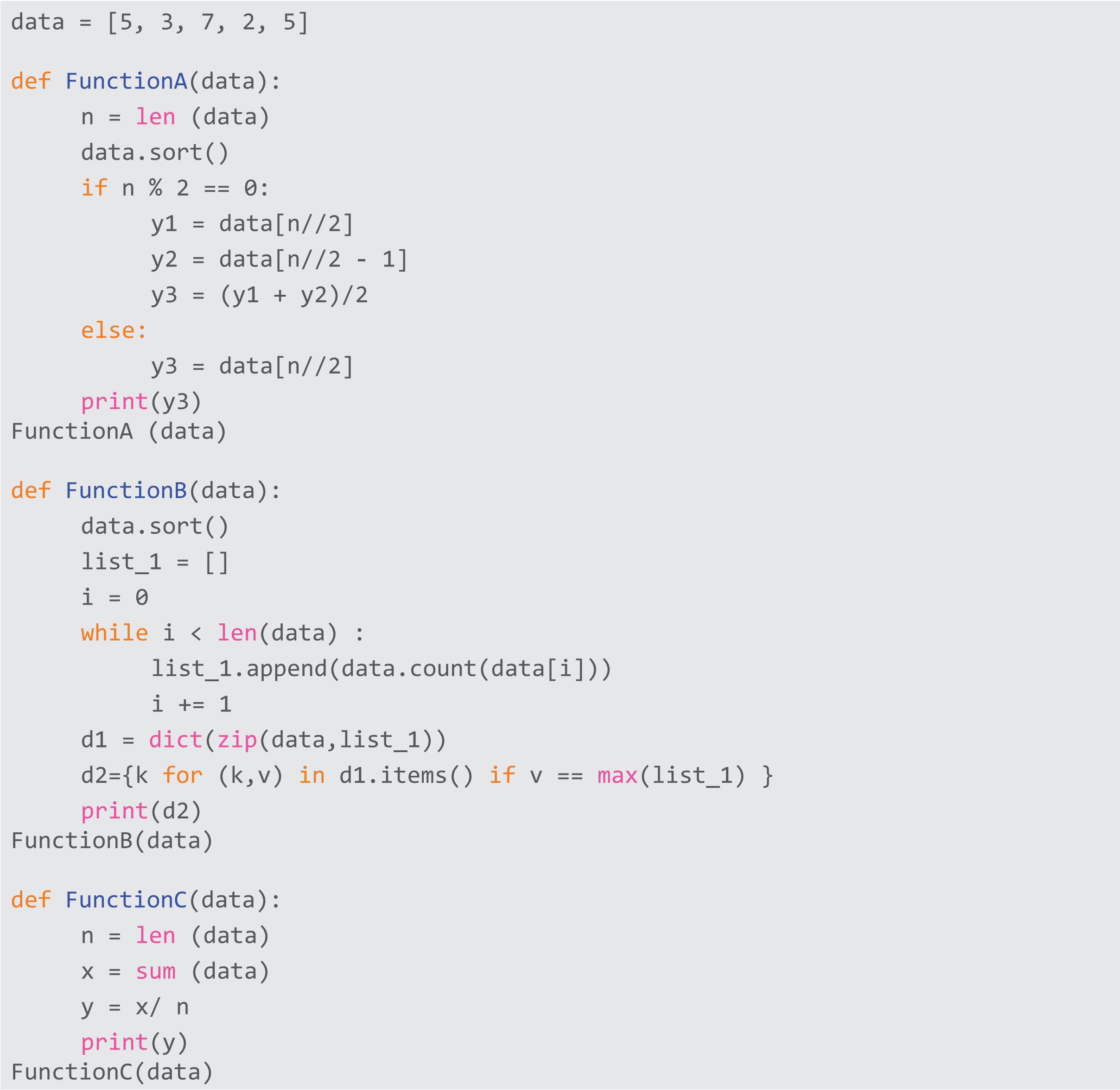
(iii) The raw data collected from the Micro: Bit is shown in the image below:



```
temperature = ["31","30","32","30","29","31","4","28","30","NA","29"]
```

The data as presented above is not suitable for further **numerical analysis**. State **three** ways that this data could be prepared or formatted to allow for numerical analysis.

(c) Examples of numerical analysis that could be performed using Python are the mean, median and mode calculations. The image below shows code for each of these calculations but without any meaningful variable names or information. Match each function (`FunctionA`, `FunctionB` and `FunctionC`) with the correct type of numerical analysis in the table below, giving a reason for each choice.



```
data = [5, 3, 7, 2, 5]

def FunctionA(data):
     n = len (data)
     data.sort()
     if n % 2 == 0:
          y1 = data[n//2]
          y2 = data[n//2 - 1]
          y3 = (y1 + y2)/2
     else:
          y3 = data[n//2]
     print(y3)
FunctionA (data)

def FunctionB(data):
     data.sort()
     list_1 = []
     i = 0
     while i < len(data) :
          list_1.append(data.count(data[i]))
          i += 1
     d1 = dict(zip(data,list_1))
     d2={k for (k,v) in d1.items() if v == max(list_1) }
     print(d2)
FunctionB(data)

def FunctionC(data):
     n = len (data)
     x = sum (data)
     y = x/ n
     print(y)
FunctionC(data)
```

Function	Numerical Analysis	Reason
A		
B		
C		

Space for extra work.

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

Space for extra work.

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

Acknowledgements

Q1. Source: https://html-online.com/editor/

Q5. Source: SEC LCCS Sample Paper

Q10.Source: https://alchetron.com/Jack-Kilby#jack-kilby-b85092a8-3438-44d6-8848-d300961362c-resize-750.jpeg

Q11. Source: https://www2.hse.ie/services/cyber-attack/how-it-may-affect-you.html

Q12. Source: https://www.javatpoint.com/von-neumann-model

Q13. resources from Flaticon.com

Q14. resources from Flaticon.com

Q15. resources from Flaticon.com

P 5 2

Pre-Leaving Certificate Examination, 2023

Computer Science

Section C

Higher Level

Time: 1 hour

210 marks

Instructions

There is one section in this paper.

Section C	Programming	One question Answer all question parts.	87 marks

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.

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

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 responsibilty to ensure that you have saved all of your files onto your external media.

Section C Programming 87 marks

Answer all question parts.

Question 16

(a) Many organisations can use programming languages to help speed up processes that would take much longer by hand. An example would be a government department responsible for calculating how much tax people pay.

Open the program called **Question16_A.py** from your device.
Enter your name on **line 2.**

```
1   #Question 16(a)
2   #Write your name here:
3
4   wages = int (input("Please enter your annual wages: "))
5
6   cutoff = 36800
7
8   def income_tax(wages):
9       print("Welcome to my income tax calculator")
10
11
12
13  income_tax(wages)
```

This program is designed to calculate how much income tax a person will pay, and the percentage of their wages lost to tax.

It will ask the user to enter their gross yearly wages; it will check if the gross wages meet the cut-off value (€36800) for income tax. If it does not meet the cut-off then no income tax will be paid. If yearly wages do meet the cut-off point, then total income tax will be applied in the following manner:

- Wages will be taxed at 20% for the first €36800 earned.
- The remaining balance of wages after €36800 will be taxed at 40%.

Tax credits are used to reduce the amount of income tax a person pays. They are subtracted from the total income tax.

Modify the program to do the following:

(i) The user currently enters their wages as an integer. Modify the program to display the following formulae:

```
Net income = Gross wages – total tax

Total tax = (€36800 x 20%) + ((Gross wages - €36800) x 40%) – Tax Credits
```

(ii) There is a hard-coded variable called "cutoff" that has a value of 36800. Create another hard coded variable called "tax_credits" directly above the cutoff variable that will have a value of 1700.

(iii) Inside the function “`income_tax`”, modify the program so that it displays to the user if they must pay income tax or not. If the user enters a wage of €36800 or more, the program displays to the user they must pay income tax; if the user enters a wage of less than €36800, then the program displays to the user they do not have to pay any tax. When the program is run, it may look as follows:

```
Please enter your annual wages: 35000
You pay no income tax
```

```
Please enter your annual wages: 40000
You will have to pay income tax
```

```
Please enter your annual wages: 36800
You will have to pay income tax
```

(iv) Currently, the program only tells the user if they do or do not pay income tax. Modify the program so it will:

- calculate the total income tax paid after tax credits have been applied. This should be rounded to 2 decimal places.
- The percentage of total wages lost to income tax (after tax credits have been applied) should also be displayed and rounded to 2 decimal places.

When the program is run, it may look as follows:

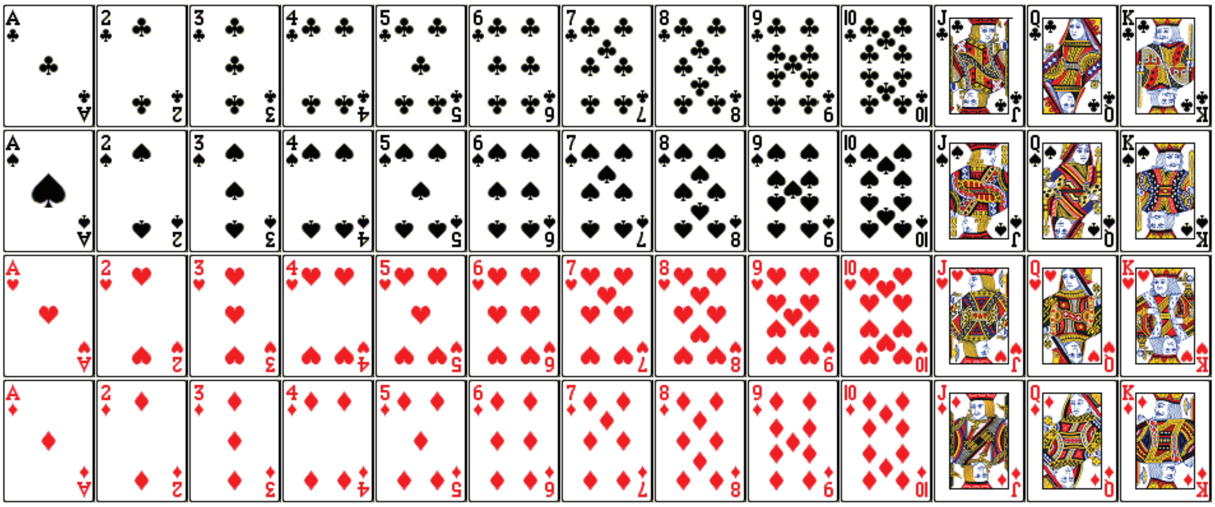
```
Please enter your annual wages: 35000
You pay no income tax
```

```
Please enter your annual wages: 40000
You will have to pay income tax
Your income tax bill is: € 6940.0
The percentage you lost to tax was: 17.35 %
```

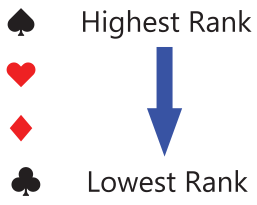
```
Please enter your annual wages: 36800
You will have to pay income tax
Your income tax bill is: € 5660.0
The percentage you lost to tax was: 15.38 %
```

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

(b) Computer programs can also be used to create simple games. One such simple game is called “High-Card Draw”. The player draws a random card from a standard deck of 52 playing cards. The computer also draws a random card from a standard deck of 52 playing cards. A standard 52-card deck can be seen below:



The player’s draw and the computer’s draw are compared in the following manner:
Whoever drew the higher face value (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K) wins. If the face values are the same, then the suits (spades ♠, hearts ♥, diamonds ♦ and clubs ♣) are used to determine the winner in the following manner:



Open the program called **Question16_B.py** from your device.
Enter your name on **line 2**.

```
# Question 16(b)
# Enter your name here:


import random

faces = [“1”,”2”,”3”,”4”,”5”,”6”,”7”,”8”,”9”,”10”, “11”,”12”,”13”]

suits = [“C”, “D”, “H”,”S”]

player_face = faces[random.randint (0,12)]
player_suit = suits[random.randint(0,3)]

computer_face = faces [random.randint(0,12)]
computer_suit = suits[random.randint(0,3)]
```

The above program contains two lists, each list is a collection of strings:

A list called “`faces`” that contains the face values of all the cards in a 52-card deck. Note that A has been replaced by 1, J has been replaced by 11, Q has been replaced by 12 and K has been replaced by 13.

It also contains a list called “suits” that contains the four suits of a 52-card deck: “C” for Clubs, “D” for Diamonds, “H” for Hearts and “S” for Spades.

The program will extract a random face and suit from those lists twice each: one for the player and one for the computer.

Modify the program to do the following:

(i) Create two new variables: one called “player_draw”. This will be the **concatenation** of “player_face” and “player_suit”. The other variable will be called “computer_draw” and will be the concatenation of “computer_face” and “computer_suit”.
Print these variables to the screen. When the program is run, it may look as follows:

```
Your draw was:  4S
The computer draw was:  8D
```

In this run of the program above, the player drew 4 of Spades and the computer drew 8 of Diamonds.

(ii) Modify the program to include a betting system. Create a new variable called “bet” that will represent an integer value from the user. The user should be prompted to enter the bet after the user draw has been made, but before the computer draw.

When the program is run, it may look as follows:

```
Your draw was:  3H
Please enter your bet: 6
The computer draw was:  8C
```

(iii) Using int() or otherwise, convert the “player_face” string variable to an integer and store it in a variable called “player_face_num”. Do the same for the “computer_face” variable and call the new variable “computer_face_num”.

(iv) Using the “player_face_num” and “computer_face_num” variables, create a game where the program will check if the player wins, computer wins or if the game ends in a draw based on the card face they drew. If the player wins, it should multiply their bet by 3 and return how much they won.

When the program is run, it may look as follows:

```
Your draw was:  4H
Please enter your bet: 6
The computer draw was:  6C
You lose!
```

```
Your draw was:  8C
Please enter your bet: 6
The computer draw was:  3C
Well done
You win: € 18
```

```
Your draw was:  2C
Please enter your bet: 6
The computer draw was:  2S
Draw
```

(v) Currently, the game can end in a draw if both the player and computer draw the same face. However, in the real game of High-Card Draw, suits are used to act as a tiebreaker in such scenarios. Modify the program so that the suits will be checked after the faces to determine a winner or if it is indeed a draw. Use the suit ranking image at the start of the question for the ranks. The betting system should be used as above. Hint: a Nested-IF-ELIF-ELSE statement may be useful.

When the program is run, it may look as follows:

```
Your draw was:  6H
Please enter your bet: 6
The computer draw was:  6H
Draw
Draw
```

```
Your draw was:  4H
Please enter your bet: 6
The computer draw was:  4D
Well done
You win: € 18
```

```
Your draw was:  4C
Please enter your bet: 6
The computer draw was:  4H
You lose
```

This is the end of 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.

Blank Page

Blank Page

P 5 2

Blank Page