Name:

Total Marks: 80

Mohammed Mahin Ibnay Mamun

Theory:

1. What is Computational Thinking? (1 Mark)

Computational thinking works by breaking down complex problems into small, manageable parts. It is split into 3: decomposion, abstraction and algorithmic thinking.

1. What is decomposition? Give an example. (2 mark)

Taking large problems and breaking them down into smaller, more manageable parts gives us a better sense of control and makes problems seem easier. For example when making a game you need to focus on creating the game first you don’t need to focus on small details like objects.

1. What is Abstraction? Give an example. (2mark)

**Finding out how a car works by looking in detail at the different parts that make up the car**

Abstraction is removing and cutting out unnecessary data. We use abstraction as computational thinking to make the problem easier. For example if you are to drive a vehicle we think about starting stopping and turning, we don’t need to worry about anything else.

1. What is Pattern Recognition? (1 mark)

Pattern recognition simply meant authentication and recognition.

1. Which of these is an example of decomposition? (Tick the correct answer) (1 Mark)

**Finding out how a car works by looking in detail at the different parts that make up the car**

1. What is a Variable? (1 mark)

Variables are used in code to store data and information. Variables are like a box which we can store items in. we can store any of the following data types in a variable: String, integer, character and Boolean

1. Why is it important to choose suitable Variable names? (1 mark)

**We use suitable variable name so it is clear which variable is being used and we know what we stored in it**

1. Choose an appropriate Variable name (Tick all the correct answers) (2 mark)

Party

Var1

name

These three

1. What is meant by Syntax error? (1 mark)

Syntax error is given for the name of an error. For example incorrect operators or numbers

1. What is meant by indentation? (1 mark)

An indentation is used in programming, it is simply an increase of spaces to the right. For example when create an if statement we use indentation to show it belongs to the if statement.

1. What is meant by Casting? Give an example. (2 mark)

Casting is also use in programming, Programmers use casting if they need to change a data type. For example if I made a program asking for a number and the input was asking for a string, I will need to change it to an integer.

1. What is meant by Concatenation? Give an example. (2 mark)

We use Concatenation when we want to attach two string in code together, an example would be printing out someone’s full name. You would make 3 variables: First\_Name, Second\_Name and Full\_Name. The user will input the first name and second as a string. You then will write

Full\_Name = (First\_Name + Second\_Name). as you can see I used a + symbol or you can use a , to concatenate.

1. Why is it important to use sequencing in programming? Give an example of sequencing. (2 mark)

It is important to use sequencing when programming because if you don’t then the code will simply not work. An example would be calculating a sum. If we say the last step before the first two it won’t work. If the first line is not at the start and the last isn’t at the end then the code won’t be efficient and may not even run.

1. Each of these SELECTION PYTHON code snippets, has ONE syntax error.   
   Can you identify the error and correct the code?

You could try running the code in Python to check if you are right!   
If you do, you will need to add an input statement for some of them as the code will not run otherwise.

a. if response = "bleep" :

Print ("cycle finished") [1]

The code is incorrect because in line one, it needs a double = not just one you must add 2 to show equals to. Also variable response is not defined so you need to add that to the start of the code.

b. mark = int(input("enter mark"))

If mark > 50 then

Print ("pass!") [1]

Line two in the code is pseudocode and is using THEN, that is why there is a syntax error, in programing python language, we use a colon (: ) .

c. if paper == "n":

print("load paper before continuing") [1]

The is no indent in the second line. It needs an indent, which is an increase of spaces. The code needs to use an indent because it has an if statement.

d. qty = int(input("how many items in stock? "))

if qty < 0 :

print("that can't be right")

else if qty > 30:

print("that's plenty, no need to reorder")

else:

print("order some more") [1]

This code will not run correctly because they have used two else functions in one if statement. This means the computer won’t know which output to print if the if statement is false. Each if statement has one if and one else. The programmer should change the first else which is on line 4 of the code to elif and then the code will run.

e. This one has invalid indentation, can you correct it?

light = input("What colour is the traffic light, R, A R/A or G?")

if light == "R":

print("stop the car")

elif light == "A":

print("get ready to go")

elif light == "R/A":

print("prepare to stop")

elif light == "G":

print("keep going")

else :

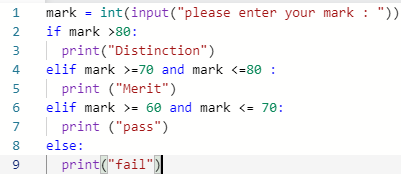
print("invalid input") [1]

This code does not work because it has too many incorrect indents which are not needed. I made this code work by deleting all the indents first so there are none and then I added indents to lines 3,5,7,9,11 because they need to be indented as they represent the if statement and the elifs in the line directly above .

1. Write a block of code which gives students a grade. If the grade is over 80 marks, display the message “distinction”. If the grade is between 70-80, display the message “merit”. If the grade is between 60-70, display the message “pass”. Otherwise, it should display “fail”.

(5 marks)

mark = int(input("please enter your mark : "))

if mark >80:

print("Distinction")

elif mark >=70 and mark <=80 :

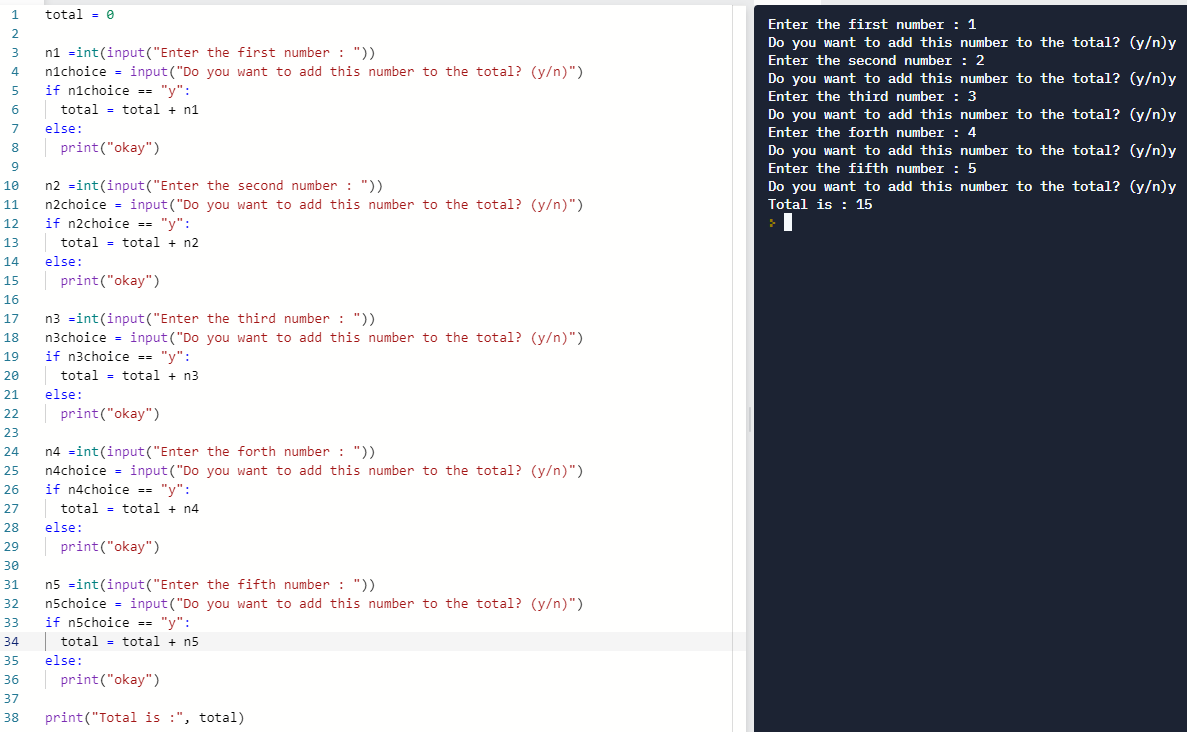
print ("Merit")

elif mark >= 60 and mark <= 70:

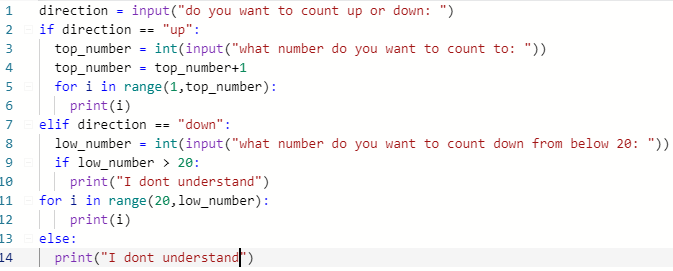
print ("pass")

else:

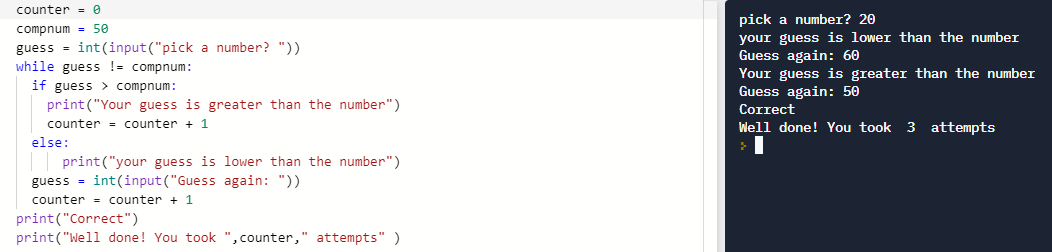
print("fail")

1. (a) :Set a variable called total to 0. Ask the user to enter five numbers and after each input ask them if they want that number included. If they do then add the number to the total. If they do not want it included, don’t add it to the total. After they have entered all five numbers, display the total.

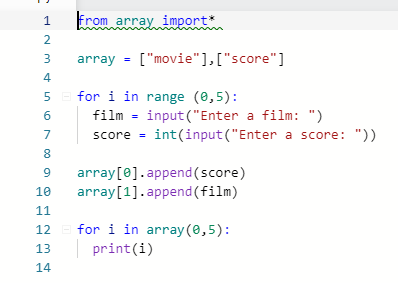
(b): Ask which direction the user wants to count (up or down). If they select up, then ask them for the top number and then count from 1 to that number. If they select down, ask them to enter a number below 20 and then count down from 20 to that number. If they entered something other than up or down, display the message “I don’t understand”. (10 marks)



1. Create a variable called compnum and set the value to 50. Ask the user to enter a number. While their guess is not the same as the compnum value, tell them if their guess is too low or too high and ask them to have another guess. If they enter the same value as compnum display the message “Well done, you took [count] attempts”. (5 marks)

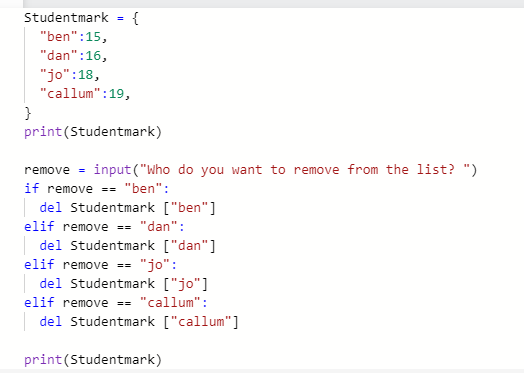


18. Create a two dimensional array that stores your top five favourite films with the individual scores out of ten you would give it:

(5 marks)

19. Numeric arrays- Create an array of five numbers between 10 and 100 which each have two decimal places. Ask the user to enter a whole number between 2 and 5. If they enter something outside of that range, display a suitable error message and ask them to try again until they enter a valid amount. Divide each of the numbers in the array by the number the user entered and display the answers shows to two decimal places. (5 marks)

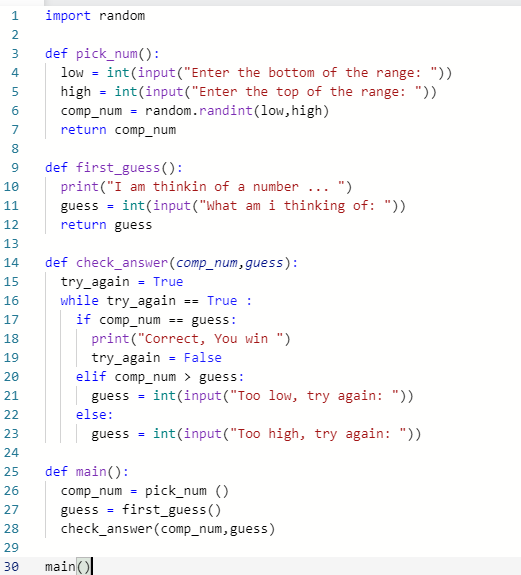
20. 2D lists and dictionaries- Create four names, ages and shoe sizes, ask the user to enter the name of the person they want to remove from the list. Delete this row from the data and display the other rows on separate lines. (5 marks)



21. Subprograms- Define a subprogram that will ask the user to pick a low and a high number, and then generate a random number between those two values and store it in a variable called “comp\_num”

Define another subprogram that will give the instruction “I am thinking of a number…” and then ask the user to guess the number they are thinking off.

Define a third subprogram that will check to see if the comp\_num is the same as the user’s guess. If it is, it should display the message “Correct, you win”, otherwise it should keep looping, telling the user if they are too low or too high and asking them to guess again until they guess correctly.

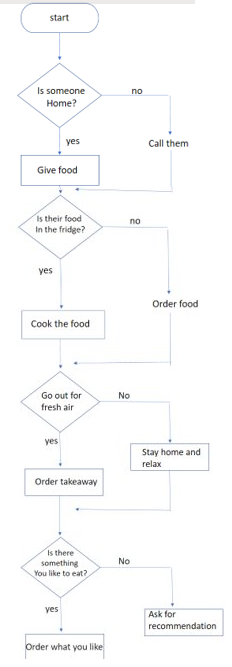


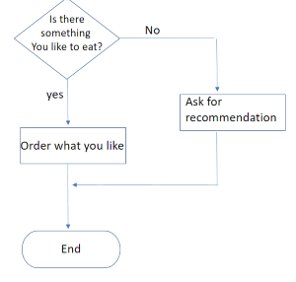
(10 marks)

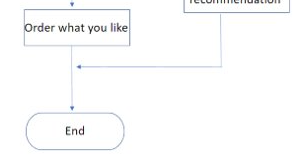
22. Create a flowchart- (5 marks)

In your flow chart you need to include as all the points below:

* Is someone at home to give food. Have a YES and No option
* Look in the fridge. Is there something to eat in the fridge? Have a YES and a NO option
* Have an option for going out for fresh air and get takeout.
* Have another option where you feel lazy to go out, so you order take outs on the phone
* Have an option for if there is something you would like to eat from the takeaway menu.
* Link everything back to one of the two options: “Stay hungry” “Enjoy your lunch”







23. Pseudocode-

Here is some sample pseudocode. Use this as a guide to write pseudocode algorithms for the descriptions in the questions below.

input PIN

if *correct PIN entered* then

unlock phone

else

print "Try again"

endif

mark = int(input(“Enter your mark?”))

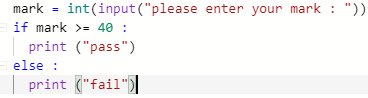
if mark >= 40 THEN

Print (“pass)

ELSE THEN

Print(“Fail”)

Read a student’s mark and print ‘pass’ or ‘fail’ depending on their mark. The pass mark is 40 or more. (3 marks)



To print the correct hat size based on the circumference of your head:

* Less than 57cm = Small
* Greater than 60cm = Large
* Anything in between = Medium

(3 marks)

