1. (a) Explain what is meant by an **algorithm**. [2]

An algorithm is a set of instructions that a computer follows. Pre defined

Here is an algorithm:

x = input(“Please enter a positive integer: ”)

answer = True

for count = 2 to (x – 1)

remainder = x mod count

if remainder == 0 then

answer = False

endif

next count

print(answer)

(b) Complete the two trace tables below for inputs of x = 7 and x = 6. [4]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **x** | **answer** | **count** | **remainder** | **Output** |
| 7 | True | 2 | 1 |  |
|  |  | 3 | 1 |  |
|  |  | 4 | 3 |  |
|  |  | 5 | 2 |  |
|  |  | 6 | 1 | False |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **x** | **answer** | **count** | **remainder** | **Output** |
| 6 | False | 2 | 0 |  |
|  |  |  |  |  |
|  | False |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

(c) What is the purpose of the algorithm? [1]

To work out how many iterations until there is no remained after taking 1 away

(d) Explain why this is not a “good” algorithm. Which of the properties of a   
good algorithm does it not have? [2]

It is not simple, it is not in a programming language, there are a few logic errors, not defined as an int.

(e) Alter the program to make it more efficient. [3]

x = int(input(“Please enter a positive integer: ”))

answer = True

for count = 2 to (x – 1)

remainder = x mod count

if remainder == 0 then

answer = False

endif

next count

print(answer)

2. A factory uses barcodes which represent 6 digits, with the last digit being a check digit.

For example, 253217 is a valid barcode.

The check digit is calculated as follows:

* assign weights of 1, 3, 1, 3, 1 to the first five digits
* multiply each digit by its weight
* add the weighted numbers together
* divide the total by 10
* the remainder is the check digit

(a) One of the following barcodes is invalid. Which one is invalid, and what should the check digit be?

(i) 212557

2,3,2,15,5 -> 27- 2.7 –

7

(ii) 365125

INVALID – BECAUSE THE OTHER IS VALID

[2]

(b) Write a pseudocode algorithm which

* inputs 100 barcodes, each input as six individual random digits
* recalculates the check digit for each one and compares it with the one input
* outputs the number of barcodes which were input correctly [6]

[Total 20 marks]