MODULE K ALGORITHM DESIGN · K8

Trace tables and dry runs

Original practice questions · 21 marks · ~25 minutes · OCR J277 specification 2.1.2

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. All code is written in OCR Exam Reference Language. In each trace table, start a new row each time a value changes; you may not need to use all the rows. Leave the Output cell blank when nothing is output. The marks for each part are shown in brackets [ ]. The total for this worksheet is **21 marks**.

1

Total: 2 marks

(a) State what a trace table is used for. **[1]**

(b) A programmer traces an algorithm. The algorithm runs to the end, but the trace table shows an output that is not what the programmer expected. **[1]**
State the type of error that the algorithm contains.

2

Total: 4 marks

A warehouse robot builds a stack of boxes. It keeps adding boxes until the stack is at least 100 cm tall. The height of each box, in cm, is entered.

```
01 height = 0
02 boxes = 0
03 while height < 100
04     boxSize = int(input("Enter box height"))
05     height = height + boxSize
06     boxes = boxes + 1
07 endwhile
08 print(boxes)
```

(a) Complete the trace table when the inputs are **30**, **45**, **20**, **15** and **40**, in that order. You may not need to use all the inputs. **[4]**

height	boxes	boxSize	Output
0	0		

3 **Total: 6 marks**

A weather station stores five daily temperatures, in °C, in the array `temps`. The array is indexed from 0.

```
01 array temps = [12, 17, 9, 21, 15]
02 highest = temps[0]
03 day = 0
04 for i = 1 to 4
05     if temps[i] > highest then
06         highest = temps[i]
07         day = i
08     endif
09 next i
10 print("Day " + str(day) + ": " + str(highest))
```

(a) Complete the trace table for this algorithm. The first row has been done for you. **[4]**

i	highest	day	Output
	12	0	

(b) Describe the purpose of this algorithm. **[2]**

4 **Total: 5 marks**

A pump engineer takes four pressure readings and needs the mean (average) of the four readings. This algorithm should output the mean, but it contains a logic error.

```
01 total = 0
02 for count = 1 to 4
03     reading = int(input("Enter reading"))
04     total = reading
05 next count
06 average = total / 4
07 print(average)
```

(a) Complete the trace table when the inputs are **8**, **6**, **10** and **4**, in that order. **[3]**

count	reading	total	average	Output
		0		

(b) Use your trace table to identify the line number that contains the logic error. Refine the line so that the algorithm outputs the mean of the four readings. **[2]**

5 **Total: 4 marks**

A print shop's cutting machine uses this algorithm. A whole number is entered. The algorithm keeps halving it, using integer division, until it reaches 1, and counts how many times it halves the number.

```
01 n = int(input("Enter width"))
02 steps = 0
03 do
04     n = n DIV 2
05     steps = steps + 1
06 until n == 1
07 print(steps)
```

(a) Create a trace table for this algorithm when the input is **13**. Write your own column headings in the top row. **[4]**

Total for this worksheet: 21 marks