ANSWERS

F13.2 Trace tables

Exam preparation · GCSE · about 15 min

What this lesson is about

Being the computer: a row for every change, and the checks that catch a slip.

- The method
- An example, by hand
- The same trace, printed
- Watch for these

Questions

5 marks in all

1. In a trace table, when do you write a new row? [1 mark]

- ☐ **A** Every time a variable changes
- ☐ **B** Once at the end
- ☐ **C** Only when something is printed
- ☐ **D** Once for each variable

> **Answer:** A. With no table given, a new row for every change is the safest method, and nothing is rubbed out. If the question's table has a row for each time round the loop, follow that instead.

2. What does this program print? [1 mark]

```
total = 0
n = 4
while n > 0:
    total = total + n
    n = n - 1
print(total)
```

> **Answer:**
>
> ```
> 10
> ```
>
> 4 + 3 + 2 + 1 = 10.

3. total = 0, n = 5. WHILE n > 1: total = total + n, n = n - 2. What is total at the end? [1 mark]

> **Answer:** 8. 5 + 3 = 8, then n is 1 and the loop stops.

4. A loop ends when its condition is false. What should the trace show for that check? [1 mark]

- ☐ **A** A row with the check written as false
- ☐ **B** Nothing: the loop has ended
- ☐ **C** The first row again
- ☐ **D** Only the output

> **Answer:** A. The final check is usually worth a mark.

5. What does 7 // 2 give? [1 mark]

> **Answer:** 3. Integer division throws the remainder away.

The task: print the trace

Trace the algorithm in the comment by running it, printing one line per row in the form `n=<n> total=<total> check=<true or false>`, in the order the computer reaches them. Print the check as `true` while the loop keeps going and `false` on the row that ends it. Finish with `output: <total>`.

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

# total = 0
# n = 5
# WHILE n > 1
#     total = total + n
#     n = n - 2
# ENDWHILE
# OUTPUT total
total = 0
n = 5
```

The hint students can ask for: Print the row at the top of each pass, before changing anything, with the result of the check. The row that ends the loop is printed too, with the check false, and then the output.

A solution

```
from bugbot import *
connect()
total = 0
n = 5
while n > 1:
    print(f"n={n} total={total} check=true")
    total = total + n
    n = n - 2
print(f"n={n} total={total} check=false")
print("output:", total)
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.