Exercise walk-through

5.2
Flowcharts, tracing and test data

International Baccalaureate · IB Diploma · Computer Science · HL

Computing Algorithms · executable trace

- Trace the state changes and boundary guard.
- Run the boundary case and compare the printed output.

```
def first_index(values, target):
    for index, value in enumerate(values):
        if value == target:
            return index
    return -1
print(first_index([4, 7, 4], 4))
print(first_index([4, 7, 4], 9))
print(first_index([], 4))
```

Exercises 2.1·2.2

2.1 Name the flowchart symbol for: (i) `OUTPUT total`; (ii) `IF age >= 18`; (iii) `count ← count + 1`. [3]

2.2 Trace the flowchart above for the inputs 30, 75, 75, and state the output. [3]

Exercises 2.1·2.2

2.1 Name the flowchart symbol for: (i) `OUTPUT total`; (ii) `IF age >= 18`; (iii) `count ← count + 1`. [3]

2.1

- (i) **parallelogram**
- (ii) **diamond**
- (iii) **rectangle**

2.2 Trace the flowchart above for the inputs 30, 75, 75, and state the output. [3]

2.2

- $30 > 0$: `max` 30
- $75 > 30$: `max` 75; the second 75 is not greater, so `max` stays 75
- output **75**

Exercises 2.3·3.1

2.3 A password must be 8 to 16 characters long. Give **one** normal, **two** boundary and **one** invalid length. [2]

3.1 Which construct does a diamond whose “yes” arrow loops back to an earlier step show? [1]

- ☐ **A** sequence
- **B** selection only
- **C** iteration
- **D** abstraction

Exercises 2.3·3.1

2.3 A password must be 8 to 16 characters long. Give **one** normal, **two** boundary and **one** invalid length. [2]

2.3

- normal: e.g. 12
- boundary: 8 and 16 (or 7 and 17 just outside); invalid: e.g. 3 or 40

3.1 Which construct does a diamond whose "yes" arrow loops back to an earlier step show? [1]

- ☐ **A** sequence
- ☐ **B** selection only
- ☐ **C** iteration
- ☐ **D** abstraction

3.1 C

- a loop back is iteration

Exercise 3.2

3.2 What is the output of this pseudocode? [1]

```
x ← 1
loop while x < 30
    x ← x * 2
end loop
output x
```

- **A** 16
- **B** 30
- **C** 32
- **D** 64

Exercise 3.2

3.2 What is the output of this pseudocode? [1]

```
x ← 1
loop while x < 30
    x ← x * 2
end loop
output x
```

- ☐ **A** 16
- **B** 30
- **C** 32
- **D** 64

3.2 C

- 1, 2, 4, 8, 16, 32: the loop stops at 32

Exercise 3.3

3.3 Which value is boundary data for an age from 11 to 18 inclusive? [1]

- **A** 14
- **B** 18
- **C** 25
- **D** “eleven”

Exercise 3.3

3.3 Which value is boundary data for an age from 11 to 18 inclusive? [1]

- **A** 14
- **B** 18
- **C** 25
- **D** “eleven”

3.3 B

- 18 is the top edge of the range

Setup for exercise 3.4

3.4 An algorithm counts how many of the numbers entered are even. It stops when -1 is entered.

```
count ← 0
input N
loop while N ≠ -1
    if N mod 2 = 0 then
        count ← count + 1
    end if
    input N
end loop
output count
```

Exercise 3.4 (a)

(a) Complete a trace table for the inputs 4, 7, 10, 0, -1, with columns `N`, `N mod 2 = 0`? and `count`. State the output. [4]

Exercise 3.4 (a)

(a) Complete a trace table for the inputs 4, 7, 10, 0, -1, with columns `N`, `N mod 2 = 0`? and `count`. State the output. [4]

(a)

`N`	`N mod 2 = 0`?	`count`
		0
4	yes	1
7	no	1
10	yes	2
0	yes	3
-1		

- 4 and 7 correct
- 10 and 0 correct: 0 is even
- loop ends at -1 without testing it
- output **3**

Exercise 3.4 (b)

(b) Identify **one** example of selection and **one** of iteration in the algorithm. [2]

Exercise 3.4 (b)

(b) Identify **one** example of selection and **one** of iteration in the algorithm. [2]

(b)

- selection: `if N mod 2 = 0`
- iteration: `loop while N` $\neq$ `-1`

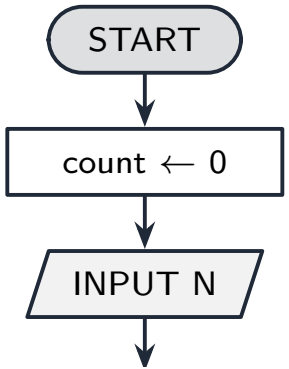
Exercise 3.4 (c)

(c) Draw a flowchart for the algorithm. [4]

Exercise 3.4 (c)

(c) Draw a flowchart for the algorithm. [4]

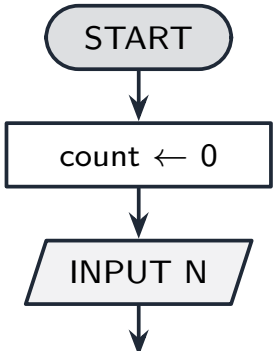
(c)



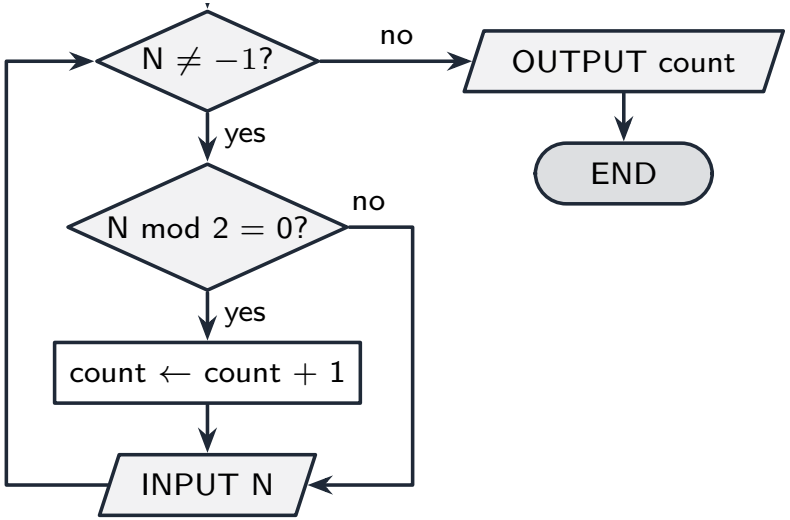
Full solution on next slide

Full solution 3.4 (c) (1/2)

(c)



Full solution 3.4 (c) (2/2)



- START, `count ← 0` and INPUT `N` in sequence
- decision `N ≠ -1?`, with no leading to OUTPUT `count` and END
- decision `N mod 2 = 0?`, yes leading to `count ← count + 1`
- INPUT `N` then an arrow back to the first decision

Exercise 3.4 (d)

(d) State the output if the first number entered is -1, and whether this is correct. [2]

Exercise 3.4 (d)

(d) State the output if the first number entered is -1, and whether this is correct. [2]

(d)

- the loop never runs: output **0**
- correct: no numbers were entered, so none are even