

A



**GCSE**

**COMPUTER SCIENCE**

**Paper 1 Computational thinking and programming  
skills – Python**

**8525/1B**

**INSERT**

**[Turn over]**

**FIGURE 1**

```
1  i ← USERINPUT
2  IF i MOD 2 = 0 THEN
3      OUTPUT i * i
4  ELSE
5      OUTPUT i
6  ENDIF
```

**FIGURE 2**

```
1  i = int(input("Enter a number: "))
2  if i % 2 == 0:
3      print(i * i)
4  else:
5      print(i)
```

**BLANK PAGE**

**[Turn over]**

### FIGURE 3

```
orderTotal ← USERINPUT
deliveryDistance ← USERINPUT
deliveryCost ← 0.0
messageOne ← "Minimum spend not met"
messageTwo ← "Delivery not possible"
IF deliveryDistance ≤ 5 AND orderTotal > 0.0 THEN
    IF orderTotal > 50.0 THEN
        deliveryCost ← 1.5
        OUTPUT deliveryCost
    ELSE IF orderTotal > 25.0 THEN
        deliveryCost ← (orderTotal / 10) * 2
```

```
    OUTPUT deliveryCost
```

```
ELSE
```

```
    OUTPUT messageOne
```

```
ENDIF
```

```
ELSE
```

```
    OUTPUT messageTwo
```

```
ENDIF
```

5

**[Turn over]**

## FIGURE 4

```
1    charge = 0
2    carReg = input("Enter your car registration: ")
3    while len(carReg) > 8:
4        displayMessage = " is not valid"
5        carReg = input(displayMessage)
6    hours = int(input("Enter your stay in hours: "))
7    if hours < 2:
8        charge = 0
9    else:
10        charge = hours * 2
11    print(charge)
```

**BLANK PAGE**

**[Turn over]**

**FIGURE 5**

**PROGRAM A**

```
print("Enter a number: ")
num = int(input())
total = 0
for i in range(1, num + 1):
    total = total + i
print(total)
```



## PROGRAM B

```
print("Enter a number: ")  
num1 = int(input())  
num2 = num1 + 1  
num2 = num1 * num2  
num2 = num2 // 2  
print(num2)
```

**[Turn over]**

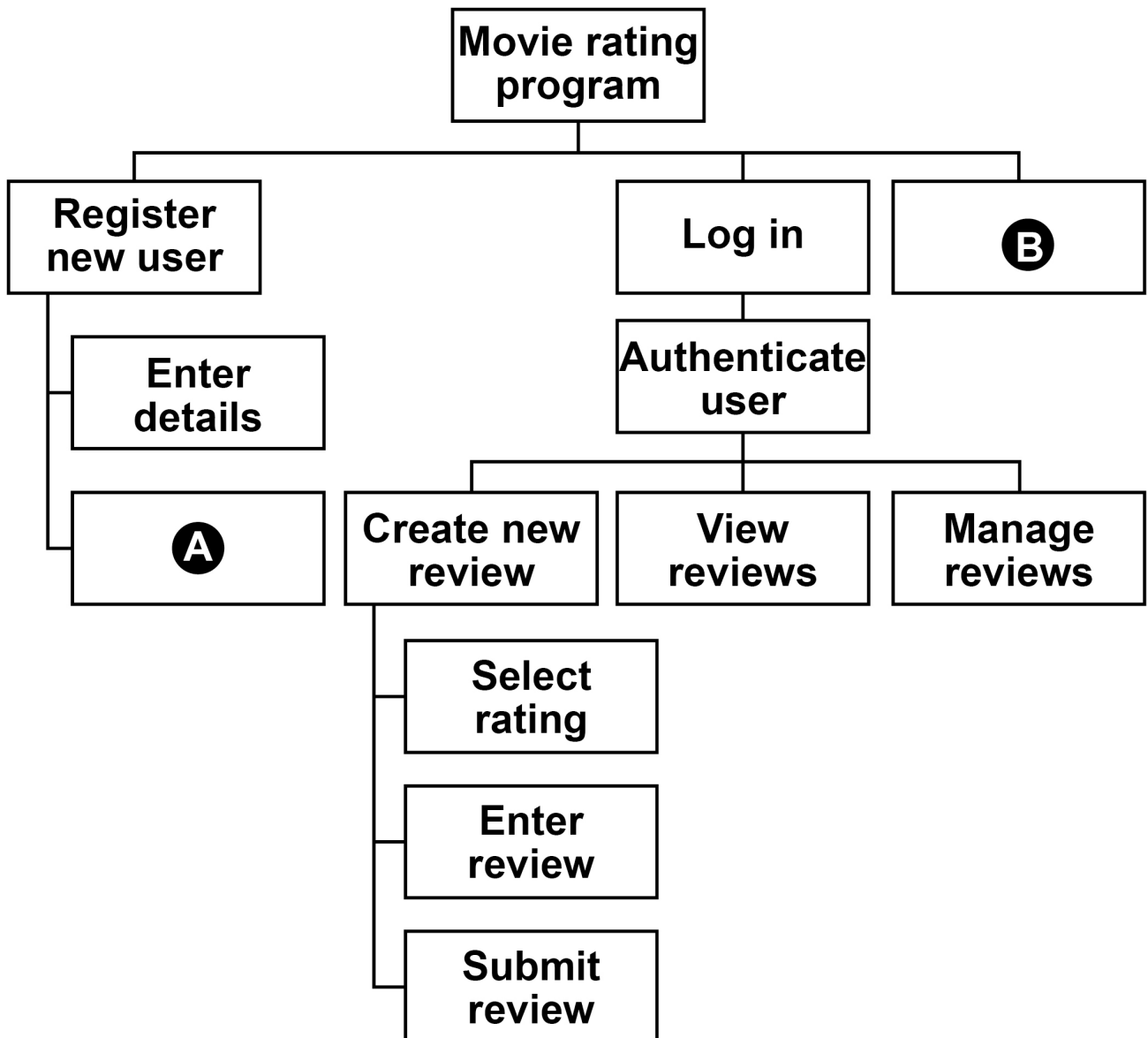
## FIGURE 6

```
1  import random
2  numbers = [ 11, 14, 56, 4, 12, 6, 42, 2 ]
3  count = 0
4  while count < 10:
5      count = count + 1
6      number = random.randrange(0, 8
7      print(numbers[count])
```

**BLANK PAGE**

**[Turn over]**

FIGURE 7



**FIGURE 8**

```
1  names ← ['Lily', 'Thomas']
2  name1 ← 'Sarah'
3  name2 ← 'Freddie'
4  OUTPUT name1[0]
5  OUTPUT LEN(names)
6  var ← SUBSTRING(0, 3, name1)
7  OUTPUT var
```

**[Turn over]**

**FIGURE 9**

SUBROUTINE calculate(n)

$a \leftarrow n$

$b \leftarrow 0$

  REPEAT

$a \leftarrow a \text{ DIV } 2$

$b \leftarrow b + 1$

  UNTIL  $a \leq 1$

  OUTPUT b

ENDSUBROUTINE

**FIGURE 10**

```
SUBROUTINE calculate(n)
  a ← n
  b ← 0
  WHILE a > 1
    a ← a DIV 2
    b ← b + 1
  ENDWHILE
  OUTPUT b
ENDSUBROUTINE
```

**[Turn over]**

**FIGURE 11**

bit	byte	getSize	OUTPUT
rate	res	RETURN	sampRate
seconds	size	size + 8	size * 8
size / 8	size MOD 8	SUBROUTINE	USERINPUT



**FIGURE 13**

```
1  arr[0] ← 'c'
2  arr[1] ← 'b'
3  arr[2] ← 'a'
4  FOR i ← 0 TO 1
5      FOR j ← 0 TO 1
6          IF arr[j + 1] < arr[j] THEN
7              temp ← arr[j]
8              arr[j] ← arr[j + 1]
9              arr[j + 1] ← temp
10         ENDIF
11     ENDFOR
12 ENDFOR
```

**[Turn over]**

**FIGURE 15**

<b>CPU</b>	<b>ALU</b>	<b>Pixel</b>
<b>NOT gate</b>	<b>Binary</b>	<b>LAN</b>
<b>Register</b>	<b>Cache</b>	<b>Protocol</b>

**FIGURE 17**

	0	1	2
0	CPU	ALU	*
1	*	*	LAN
2	Register	Cache	*

**END OF INSERT**

**BLANK PAGE****Copyright information**

For confidentiality purposes, all acknowledgements of third-party copyright material are published in a separate booklet. This booklet is published after each live examination series and is available for free download from [www.aqa.org.uk](http://www.aqa.org.uk).

Permission to reproduce all copyright material has been applied for. In some cases, efforts to contact copyright-holders may have been unsuccessful and AQA will be happy to rectify any omissions of acknowledgements. If you have any queries please contact the Copyright Team.

Copyright © 2022 AQA and its licensors. All rights reserved.

**IB/M/CD/Jun22/8525/1B/E3**

2 2 6 G 8 5 2 5 / 1 B