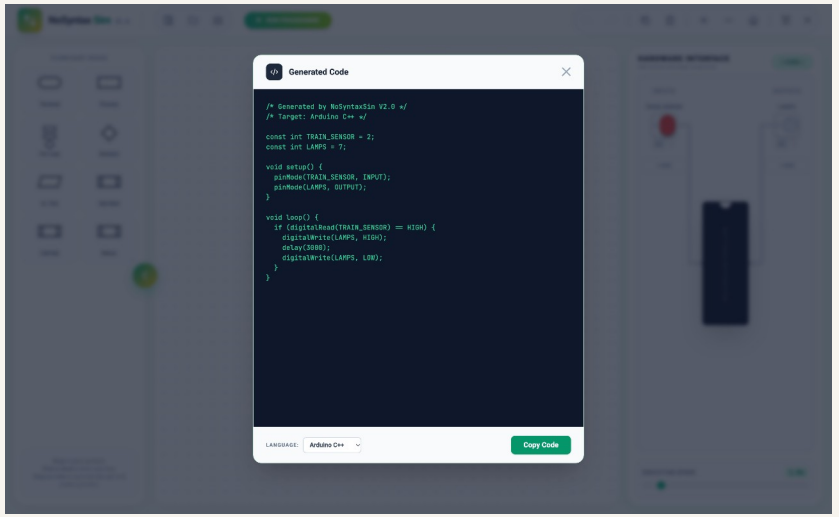
Daily Review



1	**State** how long **pause 2000** waits.	2 s
2	**State** what a **goto** above the **count reset** causes.	A never-ending loop
3	**State** the **two parts** of an **explain** answer.	Cause and effect
4	**State** what **switch on 3** does.	Switches pin 3 on
5	**State** what **x3?** becomes in a **program**.	Three counter lines

Writing Arduino C++

The Uno cannot read a flowchart. It runs Arduino C++.



National 5 Engineering Science

Which line of this code turns the lamps on?

Generated Code

```
/* Generated by NoSyntaxSim V2.0 */
/* Target: Arduino C++ */

const int TRAIN_SENSOR = 2;
const int LAMPS = 7;

void setup() {
  pinMode(TRAIN_SENSOR, INPUT);
  pinMode(LAMPS, OUTPUT);
}

void loop() {
  if (digitalRead(TRAIN_SENSOR) == HIGH) {
    digitalWrite(LAMPS, HIGH);
    delay(3000);
    digitalWrite(LAMPS, LOW);
  }
}
```

PART 1

Pseudocode to Arduino C++

Write it so the Uno can run it.

By the end of this part…



1. I can **lay out** an Arduino program: **setup()** and **loop()**
2. I can **write pseudocode** as **Arduino C++**, line by line
3. I can **find** and **fix** common Arduino **mistakes**

One Program, Two Ways

PSEUDOCODE

Flowchart

```
1 main:   switch on 5
2         pause 500
3         switch off 5
4         pause 1000
5         goto main
```

ARDUINO C++

```
1 void setup() {
2   pinMode(5, OUTPUT);
3 }
4 void loop() {
5   digitalWrite(5, HIGH);
6   delay(500);
7   digitalWrite(5, LOW);
8   delay(1000);
9 }
```

The Standard Layout



setup()

runs once: set each pin as input or output

loop()

runs for ever: the control program

```
void setup() {
  pinMode(5, OUTPUT);
}

void loop() {
  // the program
}
```

Pseudocode to Arduino C++



PSEUDOCODE

```
switch on 4

switch off 4

pause 250

if input2 is off then main

let count = 0 … if count = 4

goto main

end
```

ARDUINO C++

```
digitalWrite(4, HIGH);

digitalWrite(4, LOW);

delay(250);

while (digitalRead(2) == LOW) { }

for (int count = 0; count < 4; count++) { … }

the last } of loop()

while (true) { }
```

The Grammar Rules



;

ends every command

{ }

every { needs a }

Capitals

digitalWrite, HIGH

ms

delay(1000) is 1 s

```
1   void setup() {
2     pinMode(7, OUTPUT);
3   }
4   void loop() {
5     digitalWrite(7, HIGH);
6     delay(1000);
7     digitalWrite(7, LOW);
8     delay(1000);
9   }
```

Coming From Python?

PYTHON

```
if x == 1:
the indent marks the block
no ; at the end
# a comment
count = 0
for i in range(4):
time.sleep(1)    (seconds)
True, False
```

ARDUINO C++

```
if (x == 1) { … }
{ } mark the block
; ends every command
// a comment
int count = 0;
for (int i = 0; i < 4; i++) { … }
delay(1000);    (ms)
true, false    (pins: HIGH, LOW)
```

Worked Example: Write It in Arduino C++

PSEUDOCODE

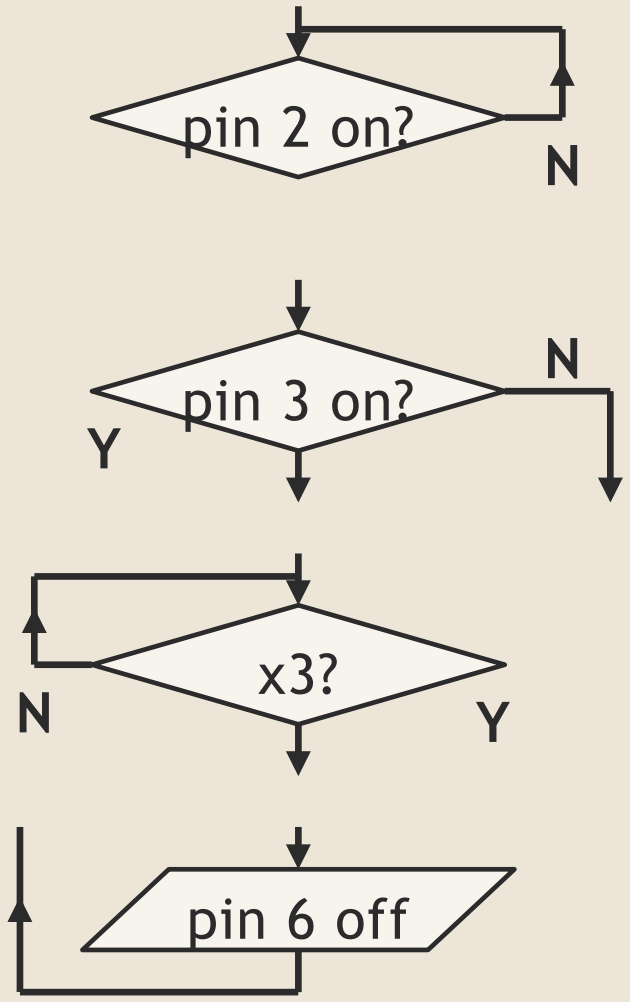
Flowchart

```
1 main:   if input2 is off then main
2         switch on 5
3         pause 3000
4         switch off 5
5         goto main
```

ARDUINO C++

```
1  void setup() {
2    pinMode(2, INPUT);
3    pinMode(5, OUTPUT);
4  }
5  void loop() {
6    while (digitalRead(2) == LOW) { }
7    digitalWrite(5, HIGH);
8    delay(3000);
9    digitalWrite(5, LOW);
10 }
```

Flowchart Shapes in Arduino C++



while (…) { }

wait here

if (…) { } else { }

two ways out

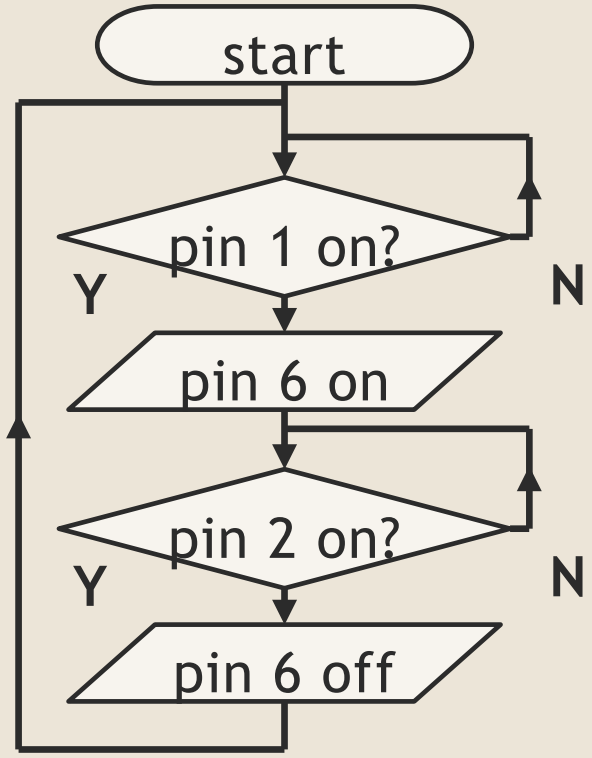
for (…) { }

count

the last } of loop()

back to the start

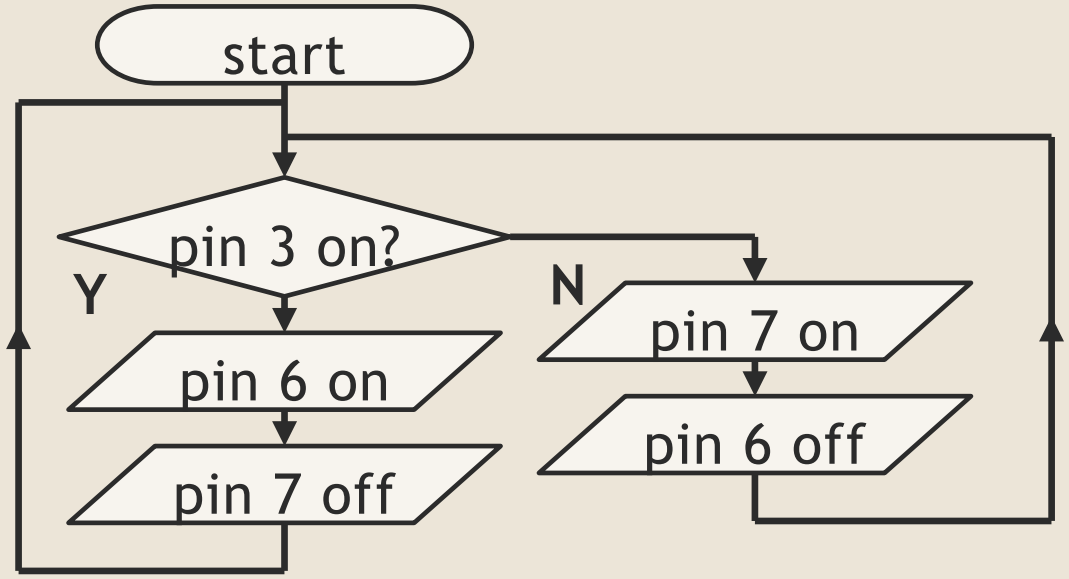
Waiting Partway Through: A Garage Door



ARDUINO C++

```
1  void setup() {
2    pinMode(1, INPUT);
3    pinMode(2, INPUT);
4    pinMode(6, OUTPUT);
5  }
6  void loop() {
7    while (digitalRead(1) == LOW) { }
8    digitalWrite(6, HIGH);
9    while (digitalRead(2) == LOW) { }
10   digitalWrite(6, LOW);
11 }
```

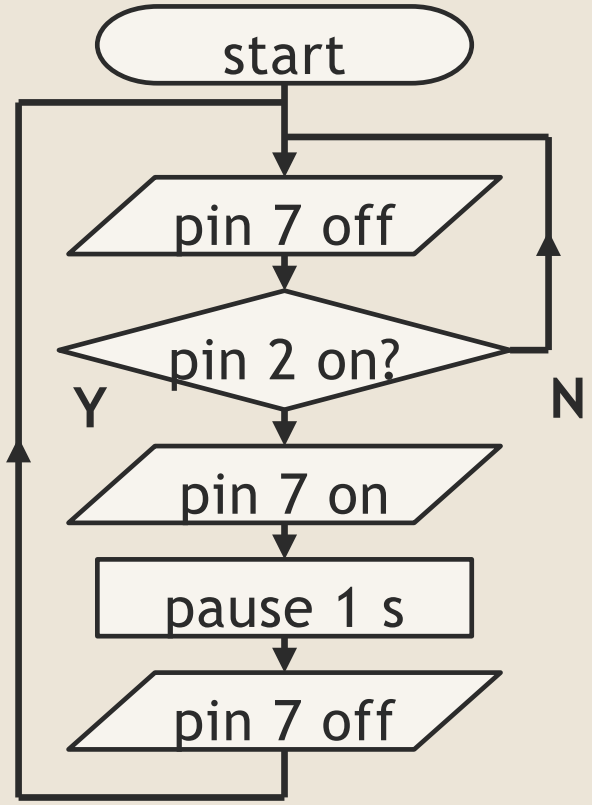
Two Ways Out: if … else



ARDUINO C++

```
1  void setup() {
2    pinMode(3, INPUT);
3    pinMode(6, OUTPUT);
4    pinMode(7, OUTPUT);
5  }
6  void loop() {
7    if (digitalRead(3) == HIGH) {
8      digitalWrite(6, HIGH);
9      digitalWrite(7, LOW);
10   } else {
11     digitalWrite(7, HIGH);
12     digitalWrite(6, LOW);
13   }
14 }
```

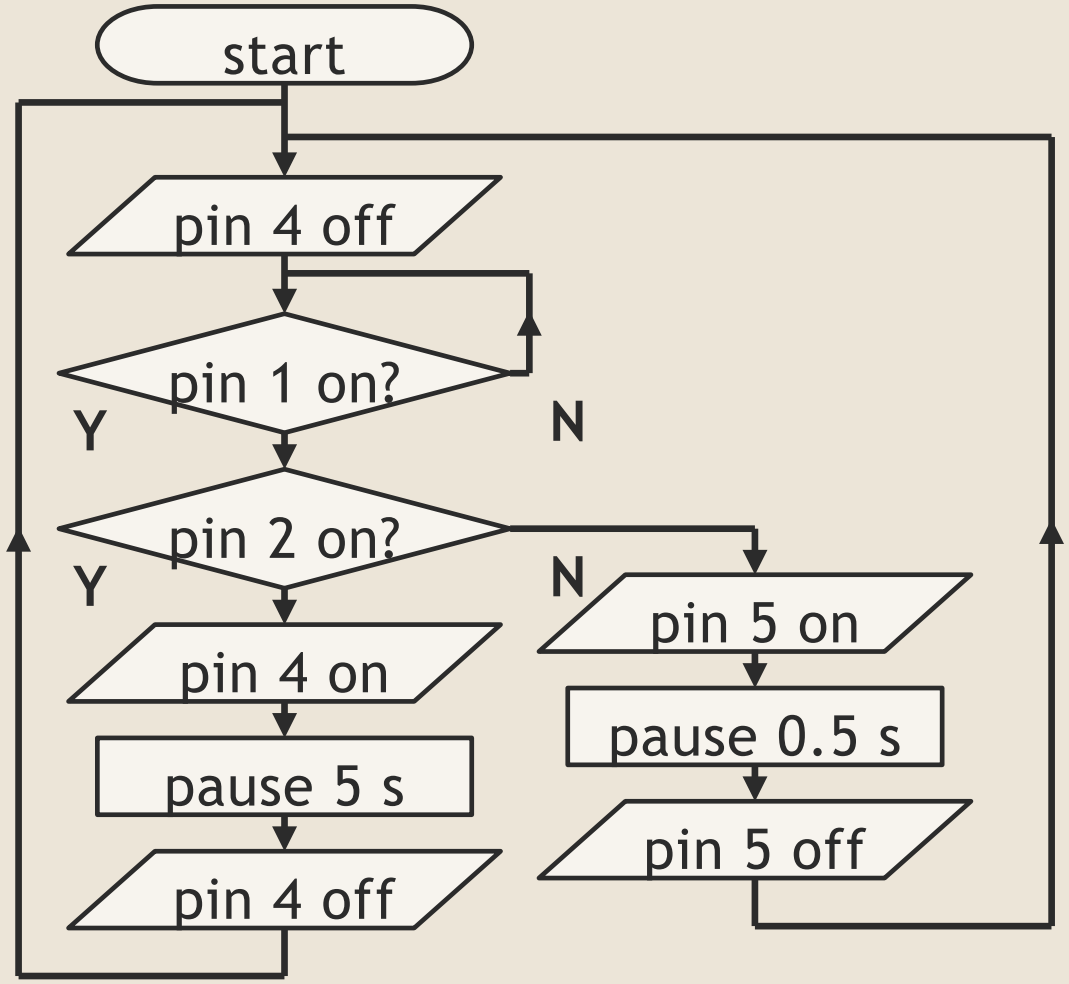
One Way Back: if With No else



ARDUINO C++

```
1  void setup() {
2    pinMode(2, INPUT);
3    pinMode(7, OUTPUT);
4  }
5  void loop() {
6    digitalWrite(7, LOW);
7    if (digitalRead(2) == HIGH) {
8      digitalWrite(7, HIGH);
9      delay(1000);
10     digitalWrite(7, LOW);
11   }
12 }
```

Two Decisions: A Hand Dryer



ARDUINO C++

```
1  void setup() {
2    pinMode(1, INPUT);
3    pinMode(2, INPUT);
4    pinMode(4, OUTPUT);
5    pinMode(5, OUTPUT);
6  }
7  void loop() {
8    digitalWrite(4, LOW);
9    while (digitalRead(1) == LOW) { }
10   if (digitalRead(2) == HIGH) {
11     digitalWrite(4, HIGH);
12     delay(5000);
13     digitalWrite(4, LOW);
14   } else {
15     digitalWrite(5, HIGH);
16     delay(500);
17     digitalWrite(5, LOW);
18   }
19 }
```

Times in Arduino C++

0.5 s

delay(500);

1 min

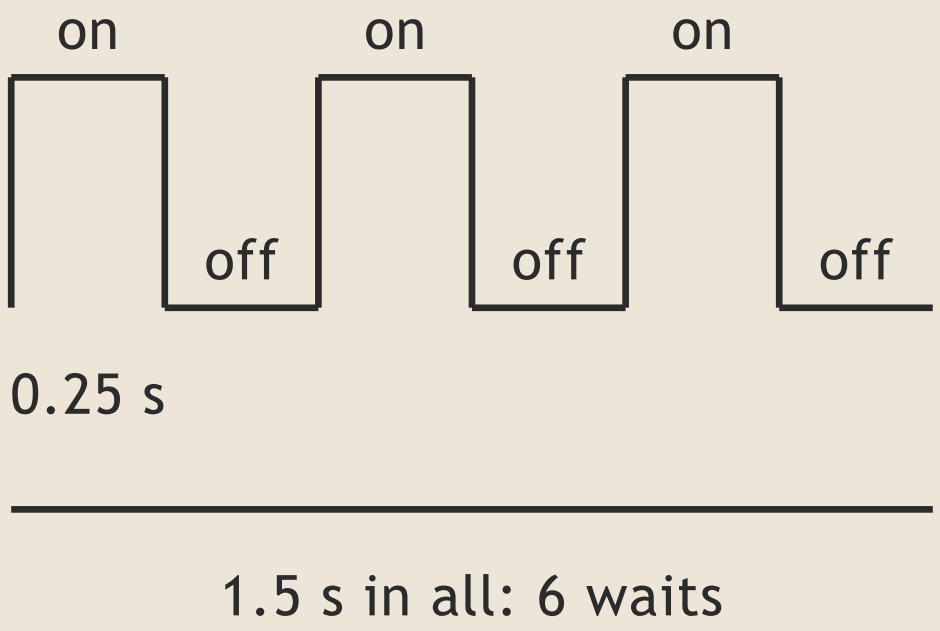
delay(60000);

5 min

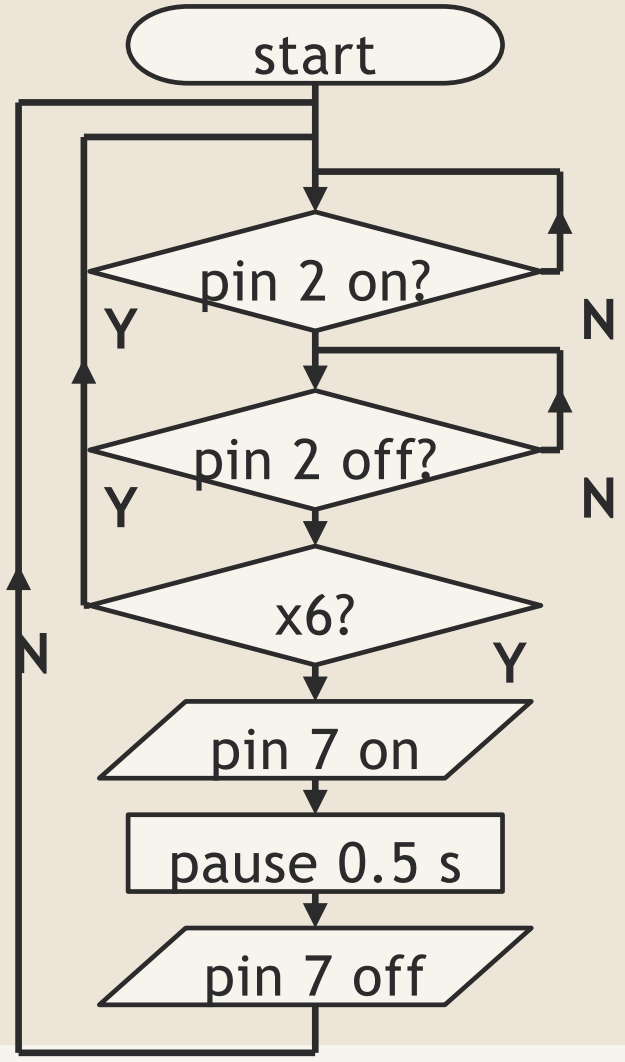
delay(300000);

3 times in 1.5 s

6 waits of 0.25 s



Counting Things Going Past



ARDUINO C++

```
1  void setup() {
2    pinMode(2, INPUT);
3    pinMode(7, OUTPUT);
4  }
5  void loop() {
6    for (int count = 0; count < 6; count++) {
7      while (digitalRead(2) == LOW) { }
8      while (digitalRead(2) == HIGH) { }
9    }
10   digitalWrite(7, HIGH);
11   delay(500);
12   digitalWrite(7, LOW);
13 }
```

Worked Example: The Alarm Bell (10 marks)

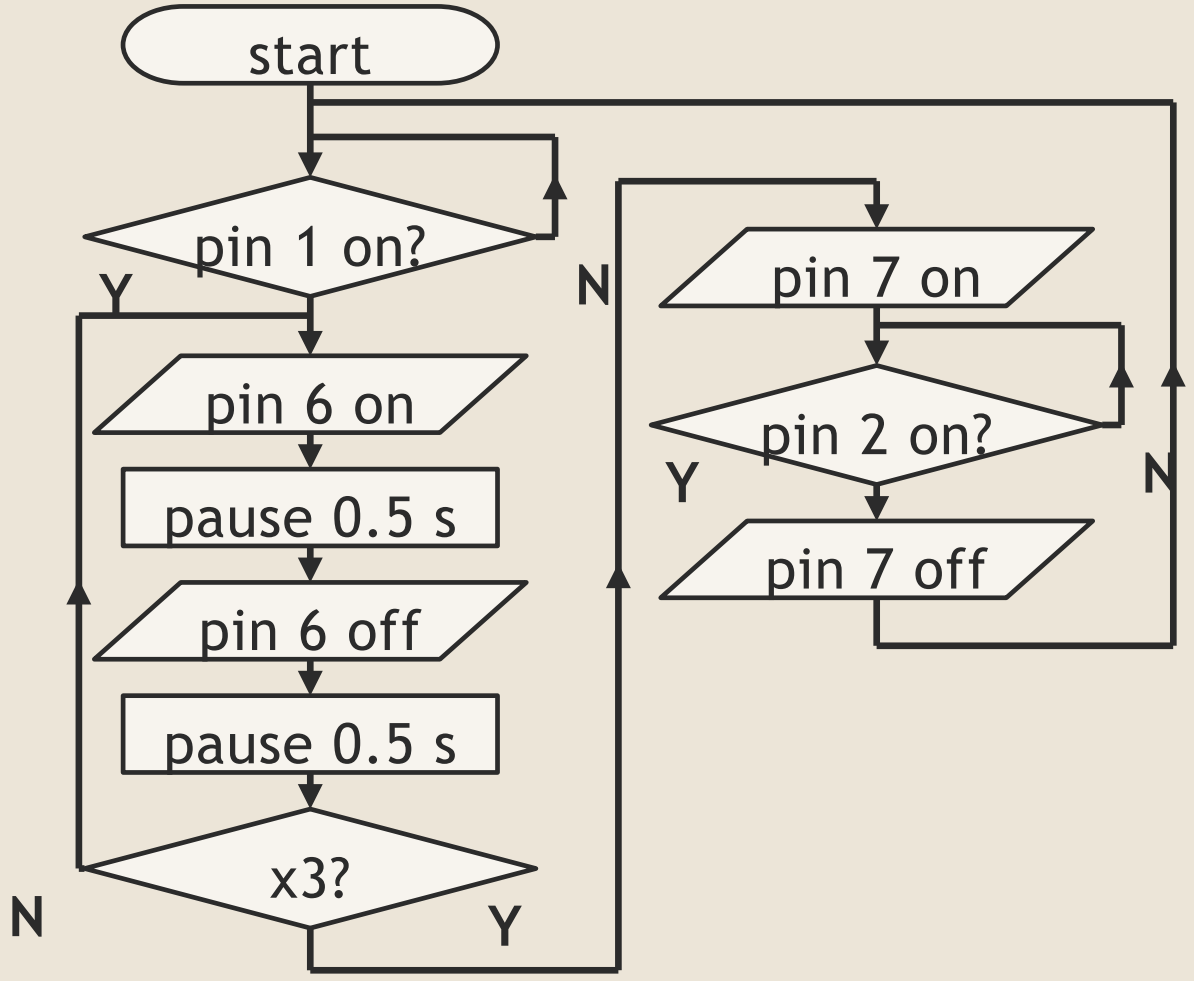
Input connection	Pin	Output connection
	7	lamp
	6	bell
reset switch	2	
start switch	1	

Complete the **flowchart**.
Include all pin numbers and **delay units**. (10)
Then **write** it in **Arduino C++**.

It operates using this sequence:

- When the start switch is activated, the bell rings three times, turning on for 0.5 seconds and off for 0.5 seconds each time.
- The lamp then turns on until the reset switch is activated.
- The lamp then turns off.
- The sequence returns to the start.

Worked Example: Flowchart and Code



ARDUINO C++

```
1  void setup() {
2    pinMode(1, INPUT);
3    pinMode(2, INPUT);
4    pinMode(6, OUTPUT);
5    pinMode(7, OUTPUT);
6  }
7  void loop() {
8    while (digitalRead(1) == LOW) { }
9    for (int count = 0; count < 3; count++) {
10     digitalWrite(6, HIGH);
11     delay(500);
12     digitalWrite(6, LOW);
13     delay(500);
14   }
15   digitalWrite(7, HIGH);
16   while (digitalRead(2) == LOW) { }
17   digitalWrite(7, LOW);
18 }
```

Your Turn: The Greenhouse (8 marks)

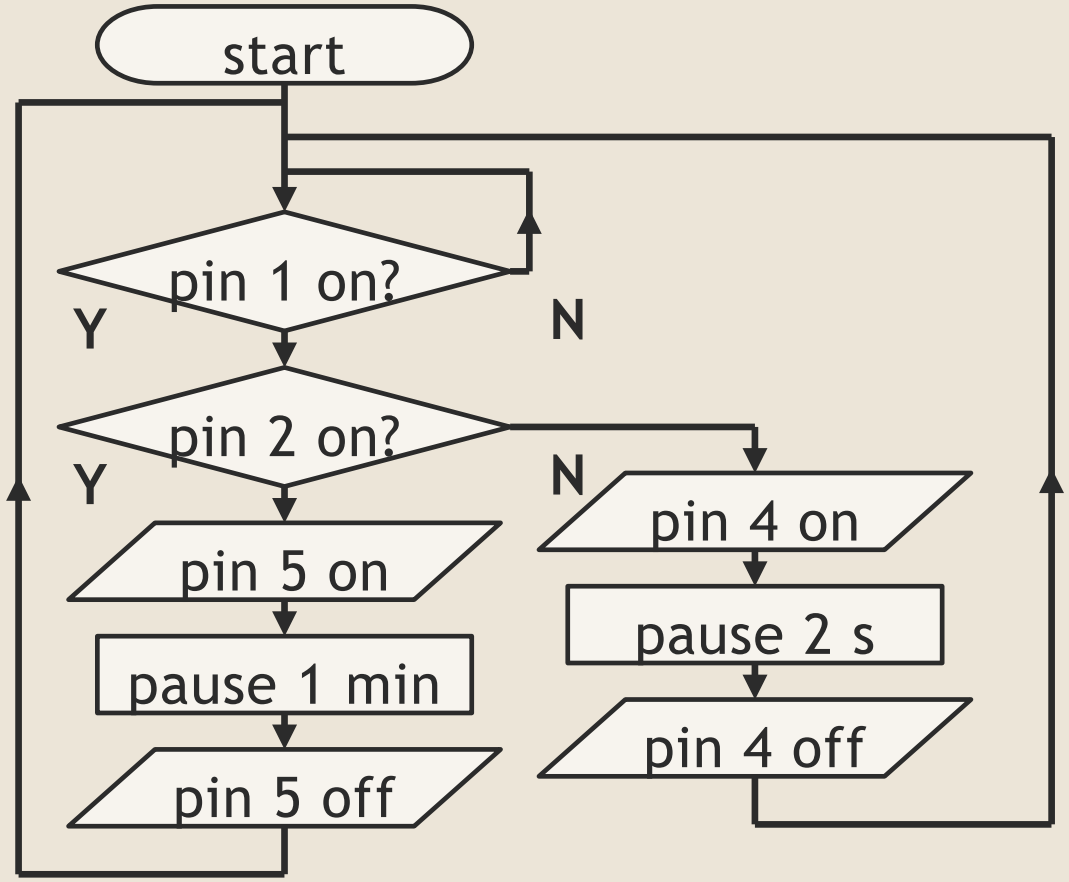
Input connection	Pin	Output connection
	5	pump
	4	green LED
soil sensor	2	
start switch	1	

Complete the **flowchart**.
Include all pin numbers and **delay units**. (8)
Then **write** it in **Arduino C++**.

It operates using this sequence:

- When the start switch is activated, the soil sensor is checked.
- If the soil is dry, the pump turns on for 1 minute and then turns off.
- If the soil is not dry, the green LED turns on for 2 seconds and then turns off.
- The sequence returns to the start.

Your Turn: The Answer



ARDUINO C++

```
1  void setup() {
2    pinMode(1, INPUT);
3    pinMode(2, INPUT);
4    pinMode(4, OUTPUT);
5    pinMode(5, OUTPUT);
6  }
7  void loop() {
8    while (digitalRead(1) == LOW) { }
9    if (digitalRead(2) == HIGH) {
10     digitalWrite(5, HIGH);
11     delay(60000);
12     digitalWrite(5, LOW);
13   } else {
14     digitalWrite(4, HIGH);
15     delay(2000);
16     digitalWrite(4, LOW);
17   }
18 }
```

Mistake 1: The Missing ;

THE MISTAKE

- 5 digitalWrite(7, HIGH)
- 6 delay(1000);
- Verify: expected ';' before 'delay'

THE CORRECT ANSWER

- 5 digitalWrite(7, HIGH);
- Every command ends with ;
- Verify names line 6: look up.

Mistake 2: The Missing }

THE MISTAKE

- 8 delay(1000);
- 9 (nothing here)
- Verify: expected '}' at end of input

THE CORRECT ANSWER

- 9 }
- Every { needs its }.
- Count them: they must match.

Mistake 3: Capitals Matter

THE MISTAKE

- digitalwrite(7, HIGH);
- digitalWrite(7, high);
- Verify: 'digitalwrite' was not declared

THE CORRECT ANSWER

- digitalWrite: capital W
- HIGH, LOW, INPUT, OUTPUT
- Copy the table exactly.

Mistake 4: = Is Not ==

THE MISTAKE

- if (digitalRead(2) = HIGH)
- = sets a value.
- Verify: lvalue required as left operand

THE CORRECT ANSWER

- if (digitalRead(2) == HIGH)
- == compares two values.
- Read == as 'is equal to'.

Mistake 5: delay(1) Is Not 1 Second

THE MISTAKE

- delay(1);
- 1 ms: too fast to see.
- Verify passes it.

THE CORRECT ANSWER

- delay(1000);
- delay() is in milliseconds.
- 1 s = 1000 ms.

Mistake 6: No pinMode

THE MISTAKE

- void setup() { }
- The pin is never an output.
- Verify passes it.

THE CORRECT ANSWER

- pinMode(7, OUTPUT);
- One pinMode for every pin.
- It goes in setup().

Spot the Mistakes

```
1    void setup() {
2      pinMode(6, OUTPUT)
3    }
4    void loop() {
5      digitalWrite(6, high);
6      delay(2);
7      digitalWrite(6, LOW);
8      delay(2000);
9    }
```

The program should **flash pin 6**: 2 s on, 2 s off.

Find the **three mistakes**.

State which mistake **Verify** would **not** find.

Spot the Mistakes: Answers



```
1    void setup() {
2      pinMode(6, OUTPUT)
3    }
4    void loop() {
5      digitalWrite(6, high);
6      delay(2);
7      digitalWrite(6, LOW);
8      delay(2000);
9    }
```

Line 2: the missing ;

```
pinMode(6, OUTPUT);
```

Line 5: the **capitals**

```
digitalWrite(6, HIGH);
```

Line 6: the **delay**

```
delay(2000);
```

Correct Your Booklet



BOOKLET (OLD)

```
high lamp
low lamp
pause 1 second
loop … end loop
for count = 1 to 4
next count



end
```

EXAM PSEUDOCODE

```
switch on 4     (the lamp's pin)
switch off 4
pause 1000
main: … goto main
let count = 0, then label_1:
let count = count + 1
if count = 4 then label_2
goto label_1
end
```

Let the Simulator Write It

Flowchart in

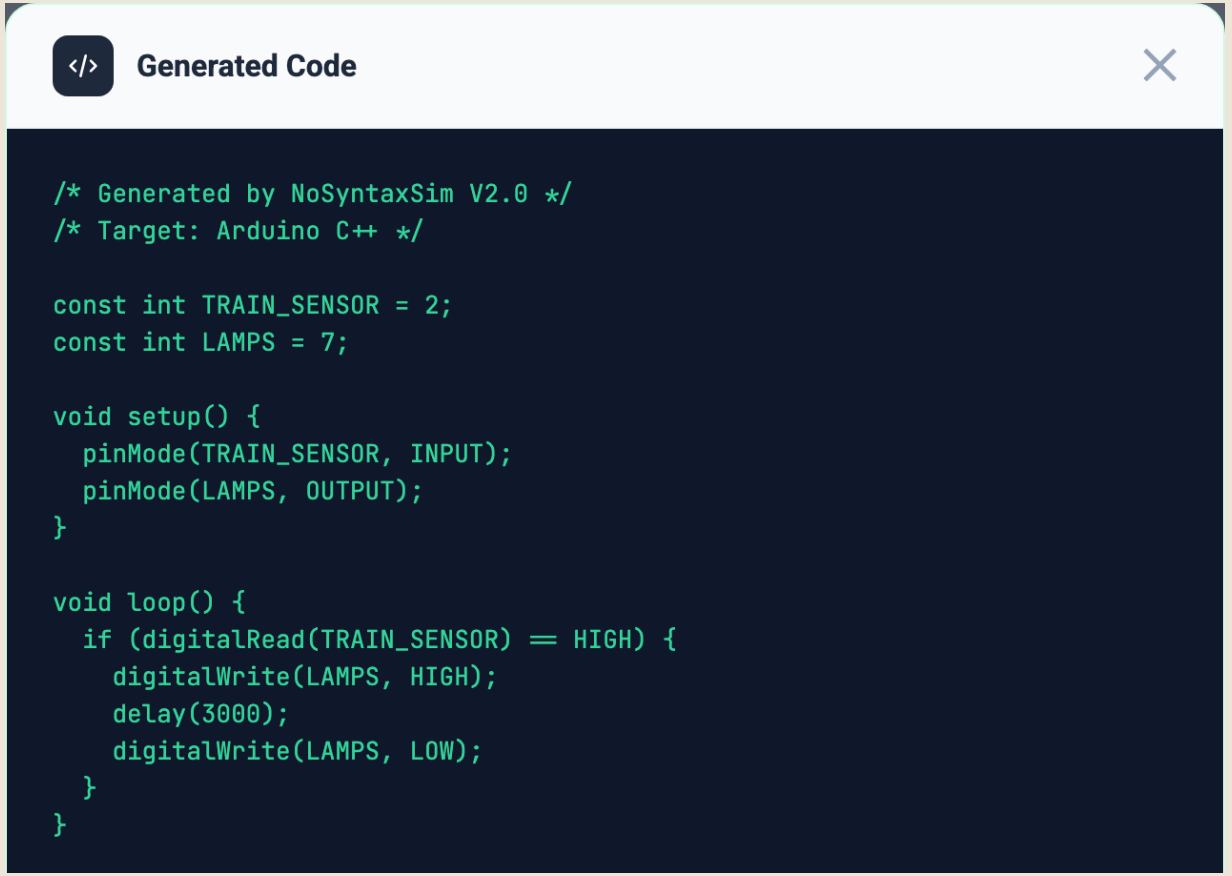
Arduino C++ out

The decision

if (digitalRead(...) == HIGH)

The wait

delay(3000)



What Does It Do?



1. **State** what **pinMode(3, INPUT);** does.
2. **State** what **digitalWrite(7, HIGH);** does.
3. **State** how long **delay(250);** waits. (**value and unit**)

It sets pin 3 as an input

It switches pin 7 on

0.25 s

What Does It Do?



4 **State** what **digitalWrite(7, LOW);** does.

It switches pin 7 off

5 **State** what **digitalRead(2)** does.

It reads the input on pin 2

PRACTICE: Programming in Real Life

Q1-6 (B4b-p22-23)

Answer Q1 to Q6 **on your own.**
Leave Q7 and Q8 for **Lesson 4.**

1. State what the setup() function is for in an Arduino C program. (1)

2. State what the loop() function is for in an Arduino C program. (1)

3. Write the Arduino C line that switches ON an LED on pin 13. (1)

4. Write the Arduino C line that sets pin 4 as an input. (1)

5. State how long delay(2500) waits for. (1)

6. Write pseudocode that flashes an LED 4 times, on 0.5 s and off 0.5 s, then stops. (4)

Answers: Programming in Real Life (B4b-p22)



Question	Answer
1 State: setup()	It runs once at the start and sets each pin as an input or an output.
2 State: loop()	It runs for ever after setup(): the main program.
3 Write: LED on, pin 13	digitalWrite(13, HIGH);
4 Write: pin 4 as an input	pinMode(4, INPUT);
5 State: delay(2500)	2.5 s

Answers: Q6 (4 marks) (B4b-p23)



```
1   main:       let count = 0
2   label_1:    switch on 4
3               pause 500
4               switch off 4
5               pause 500
6               let count = count + 1
7               if count = 4 then label_2
8               goto label_1
9   label_2:    end
```

Write pseudocode that **flashes** an LED **4 times**, on 0.5 s and off 0.5 s, then **stops**.

Marks

4 times (1); on (1); off (1); both pauses, then end (1)

Plenary: Part 1



1. **State** how often **setup()** runs.
2. **State** how long **delay(1000)** waits. (**value and unit**)
3. **State** what every **command** ends with.

Once

1 s

A semicolon ;

Plenary: Part 1



4 **State** which sign **compares** in an **if**: = or ==.

==

5 **Write** the booklet's **pause 1 second** the exam's way.

pause 1000

Extension

1. In the 4-beep program, line 8 is changed to **goto main**. **Describe** what **happens**.

It beeps for ever

2. **State** how many times **for (int i = 0; i < 5; i++)** runs.

5 times

3. **Explain** why Arduino C++ needs no **goto main**.

loop() already repeats for ever

Look back — by the end of this part…



1. I can **lay out** an Arduino program: **setup()** and **loop()**
2. I can **write pseudocode** as **Arduino C++**, line by line
3. I can **find** and **fix** common Arduino **mistakes**

LOOKING AHEAD

Lesson 4: Build, Test and Fix

Put it on the Uno. Then find what is wrong with it.

Course Specification

Digital electronic control systems

- using suitable commands (high, low, for…next, if…then, pause, end (or their equivalents)) to design programs to solve simple control problems, involving time delays and continuous and fixed loops

Credits

Title slide (NoSyntaxSim): NoSyntaxSim V2 screenshot (Generate Code). Simulator by Peter Strain, NoStrainSims (sites.google.com/view/nostrainsim), free and open to all.

The big question (NoSyntaxSim): NoSyntaxSim V2 screenshot (Generate Code). Simulator by Peter Strain, NoStrainSims (sites.google.com/view/nostrainsim), free and open to all.

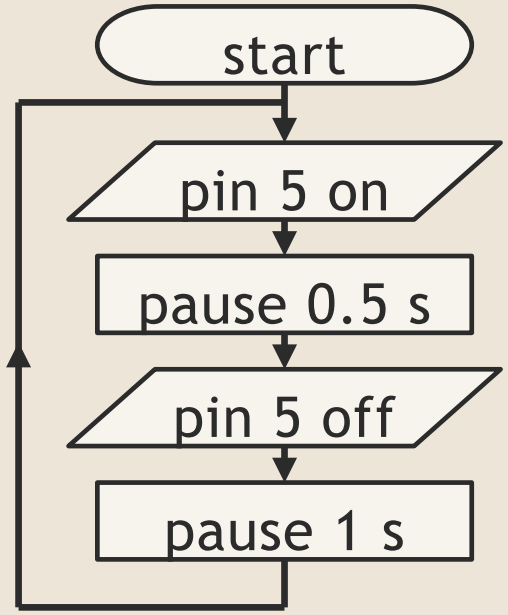
Let the Simulator Write It: NoSyntaxSim V2 screenshot (Generate Code). Simulator by Peter Strain, NoStrainSims (sites.google.com/view/nostrainsim), free and open to all.

PRACTICE: Programming in Real Life (booklet page): Booklet 4b pages by R Stewart.

Course Specification: © Qualifications Scotland (SQA). Course spec v2.0, p.5

R Stewart - October 2026

The Flowchart: One Program, Two Ways

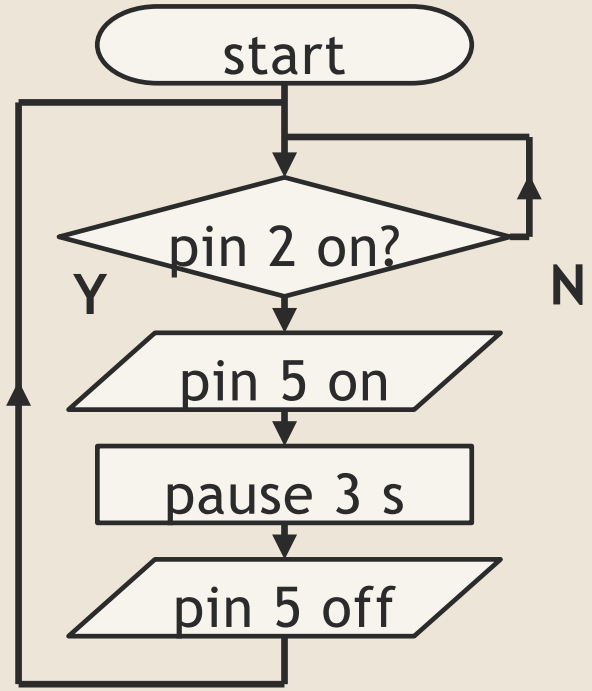


ARDUINO C++

Back

```
1 void setup() {
2   pinMode(5, OUTPUT);
3 }
4 void loop() {
5   digitalWrite(5, HIGH);
6   delay(500);
7   digitalWrite(5, LOW);
8   delay(1000);
9 }
```

The Flowchart: Worked Example



ARDUINO C++

Back

```
1  void setup() {
2    pinMode(2, INPUT);
3    pinMode(5, OUTPUT);
4  }
5  void loop() {
6    while (digitalRead(2) == LOW) { }
7    digitalWrite(5, HIGH);
8    delay(3000);
9    digitalWrite(5, LOW);
10 }
```