LCCS 2026 Coursework – Modelling Wildfire Risk

1. What Are We Modelling?

- We live in an environment made up of many connected systems, such as the **seas**, **glaciers**, **mountains**, **air**, and **forests**.
- For this project, we are focusing on **forests** and the **dangers they may face.** We refer to these dangers as **environmental risks**.

Examples of risks to forests include:
- wildfire
- Flooding
- Landslides
- Pests
- Disease (Diseases that damage or kill trees).

Note: Rather than modelling the event itself *(we are* ***not*** *modelling a forest fire)*, we are modelling the **risk** of that event occurring.

Basic Requirements: Microbit {Collecting Data}

Advanced Requirements: Model

- In this project, we create a **computer model** to estimate how **risky** the **conditions** are for a chosen threat, such as wildfire.

Note: '*risky*' means the **chance or likelihood** that an event *could* occur under certain conditions. It does **not** mean that the event will definitely happen.

- The **purpose** of the model is to **support decision-making**, for example by helping identify when action might be needed to reduce or prevent the risk.

Because we cannot predict real-world events with certainty, the model does not produce definite outcomes. Instead, a **model** is used to explore **what could happen** under different conditions.
How? By using **"what-if" scenarios (simulations)**

e.g., *what if the temperature increased by 3 °C and the humidity decreased by 5 %?*

The model then estimates how the **risk level** might change, which helps us think about **possible actions to reduce or manage the risk** {mitigation}.

Basic Requirements

2.a Choosing Input Data (Variables) BR1

Task: To model anything, e.g., wildfire risk, we **collect** environmental **data that influences how fires can start or spread.** These measurements are inputs to the model.

Use **at least one analogue input** e.g., temperature, humidity, soil moisture

Use **at least one digital input** e.g., a button to start or calibrate the system)

Produce **at least one output** e.g., LEDs, display, sound, or message

Operate **automatically** once started

The environmental measurements collected by the system are the **inputs to the model**.

Typical **inputs** include:

- Temperature (°C)
- Humidity (%)
- Wind speed (km/h)
- Soil moisture (%)

Note: Segment 1 of Video {Approx. 45 Seconds}: Video Check List

Purpose of the System

Show on video: The complete embedded system powered on, with the micro:bit, **sensors**, and output visible.

Say: Briefly state (approx. 2 sentences) what **data** your embedded system is **designed to gather {name sensors}**

Digital Input

Show on video: The **button** or switch clearly, and **activate** it to start or calibrate the system.

Say: Identify the digital input used (e.g. button or switch) and explain that it is used to start or calibrate the system.

Calibrate: Using a button or switch to set the system's starting conditions or baseline before it runs automatically e.g., maybe a **reset** button of some kind.

Automatic Operation

Show on video: Show the system continuing to run with no user interaction after button is pressed.

Say: Explain that after starting, the system runs automatically without further user input.

Analogue Input

Show on video: The analogue sensor e.g., **temp** and a visible change to its input e.g., **warming the sensor**

Say: Identify the analogue sensor used (e.g. temperature, light, soil moisture) and state what **range of values** it produces. {an examiner might not know the range of values of a soil moisture sensor}.

Primary Output

Show on video: The primary output clearly active (LED, sounder, or display).

Say: Identify the main output used (digital <u>or</u> analogue) and explain **what the output is showing** when the system is running. This will primarily be the values from the sensor. You can add an alert if it goes 'above' a certain level that you define as 'dangerous'

MakeCode Program {or any other code used here}

Show on video: The MakeCode editor with the program visible, focusing on the main blocks (start, input reading, rules, output).

Say: Explain {your code} briefly - that the program uses simple rules to read inputs and control the output.

2.b Raw Data Collection BR2

Your embedded system is used to **collect and store environmental data**.
This data is **raw numerical data**.
Example of daily environmental readings:

Note: You only need to collect **one** of these but you may collect more.

Sample:

Day 1: Temp 18°C, Humidity 65%, Wind 10 km/h, Soil Moisture 42%

Task: Collect

Validate

Clean

Store

Note: Segment 2 of Video {Approx. 45 Seconds}

Purpose of Data Collection

Show on video: The embedded system powered on with the micro:bit and the data-collection sensor visible.

Say: Briefly state **what data your sensor is collecting**

Collect Raw Data

Show on video: Live sensor readings being taken (LED display, serial monitor, or data output).

Say: Explain that the system collects raw environmental data directly from the sensor.

Validate Data

Show on video: The data validation that you are performing.

Say: Explain what type of data validation you are performing e.g., - is data in **int** format.
- range of acceptable data

Clean Data

Show on video: How you cleaned the data

Say: Explain how you cleaned the data.

Store Data

Show on video: Evidence of data being saved (CSV file} or any other database.

Say: Briefly describe the data being **stored** and the data 'type', e.g., int, float, string

**** Identify which input data you will be using in your model**

2.c Process Simulation Using Rules BR3

Task: You must use your embedded system to simulate a real-world process related to wildfire risk by showing how the system changes behaviour when environmental conditions change.

This simulation is **not** a rules-based risk model. {We address this in Advanced Requirements}

Example: Simulating Wildfire Conditions

- The embedded system measures **temperature** using a sensor.
- A **hair dryer** (set to cool or low heat, held at a distance) is used to gently increase the air temperature near the sensor.
-

Digital Output: As the temperature rises, **the system automatically changes its output**:

Cooler air → 1 LED on
Warmer air → 2 LEDs on
Hotter air → 3 LEDs on

or

Analog Output: A gradually increasing sound volume is used to show the temperature **changing smoothly over time**:

Students observe and record how the output changes as conditions change.

This simulates hotter environmental conditions associated with wildfire risk, without modelling or predicting a fire.

Note: Observe and **record** how the output changes as conditions change.

Note: Segment 3 of Video {Approx. 45 Seconds}

Simulation

Show on video: What real-world process you are simulating e.g., soil moisture, temp, etc

Say: Explain the real-world process you're simulating and its relevance to your solution {project}.

Test 1

Show on video: The system running while environmental input values are set to safe conditions, and the output displaying a low or safe status.

Note: **You will need to artificially simulate conditions e.g., a hair dryer….**

Say:

- What conditions they are using (safe or normal conditions)
- What the output shows
- Explain how the output makes sense for those conditions {name} being simulated

Display: **One** LED is turned on to indicate a safe display. **{Digital}**

Display: The needle/servo pointer moves to the low end of a scale (e.g. 0–2 on a 0–10 dial) to show a safe/low status. **{Analog}**

Test 2

Show on video: The system running while environmental input values are changed to **moderate** conditions, and the output displaying a medium or caution status.

Note: You will need to artificially simulate conditions e.g., a hair dryer….

Say:
- What conditions they are using (safe or normal conditions)
- What the output shows
- Explain how the output makes sense for those conditions {name} being simulated

Display: **Two** LEDs are turned on to indicate a moderate or caution display. **{Digital}**

Display: The needle/servo pointer moves to the middle of the scale to indicate a medium or caution status. **{Analogue}**

Test 3

Show on video: The system running while environmental input values are changed to more **extreme** conditions, and the output displaying a high or warning status.

Note: You will need to artificially simulate conditions e.g., a hair dryer….

Say:
- What conditions they are using (safe or normal conditions)
- What the output shows
- Explain how the output makes sense for those conditions {name} being simulated

Display: **Three** LEDs are turned on to indicate a high or warning display. **{Digital}**

Display: The needle/servo pointer moves to the middle of the scale to indicate a high or dangerous status. **{Analogue}**

Advanced requirement 1

4. Risk Scores

Approach 1: Rules- Based Method: This involves making **justified assumptions BASED** on **real-world research**.

Note: Reference this research also.

How Risk Scores Are Decided

Risk scores are calculated using simple **rules based on research**.
For each set of input values, the model applies these rules to produce a **single risk level**.

Data Types: - Numerical Data e.g., Temperature 18(°C)
- Ordinal Data e.g., Wet, Dry etc.

Step 1: **Data is ready {includes Microbit Data AND data downloaded/generated}**

** **Reminder:** **What data will you collect using the Microbit?**

Step 2: **Set a 'Risk Scale'. Here is an example:**

Key
1 is Low Risk
2 is Medium Risk
3 is Height Risk

Step 3: **Our data does NOT come with a risk level**

Reading	Temperature (°C)	Humidity (%)	Soil Moisture	Wind Speed	Risk Level
A	18	65	Wet	Low	
B	22	55	Damp	Low	
C	25	45	Dry	Med	
D	28	35	Dry	Med	
E	32	25	Very Dry	High	

Risk Levels: **How DO we arrive at these actual Risk Levels?**

Risk levels are **NOT** given in the data. They are **calculated** by **applying scoring rules** {**rule or assumptions based on real world-research**} to each set of readings and converting the total score into a risk level.

Risk Scoring Rules and Risk Level Scale

Scoring Rules: **Based on real-world research and justified assumptions**
The Higher the Score = The Greater the Perceived Risk

Risk Level: **The final value calculated by applying the scoring rules to the data**

Step 3a: Scoring Rules {Justified assumptions BASED on real-world research}			
Temperature: 18°C ≤ 25°C Score: +0 26-30°C Score: +1 >30°C Score: +2	**Humidity: 65%** ≥ 50% Score: +0 30-49% Score: +1 < 30% Score: +2	**Soil Moisture: Wet** Wet Score: +0 Damp Score: +1 Dry/Vr. Dry Score: +2	**Wind Speed: Low** Low: Score +0 Med: Score +1 High: Score +2

Step 3b: Add up Total Score:	Step 3c: Convert Total Points to Risk Level
Row A: Temp 18, Humidity 65, Soil Wet, Wind Low Temperature contributes +0 Humidity contributes +0 Soil contributes +0 Wind contributes +0 **Total Score = 0 points**	0-2 points → Risk Level 1 (Lo) 3-5 points → Risk Level 2 (Med) 6-8 points → Risk Level 3 (Hi) **Final Risk Level: 1**

4: Research Justification [WHY we chose these specific {scoring} rules] **For Video**

Temperature rule

Because research shows that wildfire risk increases significantly above 25 °C, the model assigns higher risk scores as temperature rises beyond this value.

Humidity rule

Research indicates that lower humidity leads to drier vegetation, increasing fire risk. For this reason, the model increases the risk score when humidity drops below 50%.

Soil moisture rule

Dry and very dry soil conditions reduce moisture available in vegetation. Based on this, the model assigns higher risk scores to dry soil conditions.

Wind rule

Wind increases the spread of fire. Therefore, higher wind speeds are assigned higher risk scores in the model.

Video Evidence: **ALL of this is PRIORITY EVIDENCE FOR YOU VIDEO**

Approach 2: Data-Driven Method

Note: In Approach 1 (method shown above in Steps 3a-3c), we defined **scoring rules** FIRST based on research, **then** calculated risk levels. However, there is an alternative approach:

If you have historical data with known risk levels already assigned (e.g., from past wildfire records or expert assessments), you could:

This method MUST come with Risk Levels ALREADY assigned, examples of where this might be the case:

- Historical wildfire records (actual fires that occurred)
- Expert assessments (fire professionals who evaluated conditions)
- Previous models/systems that already classified the risk

1. Start with data that already includes risk levels:

Reading	Temperature (°C)	Humidity (%)	Soil Moisture	Wind Speed	Risk Level
A	18	65	Wet	Low	1
B	22	55	Damp	Low	1
C	25	45	Dry	Med	2
D	28	35	Dry	Med	2
E	32	25	Very Dry	High	3

2. Use **correlations, scatter graphs, and statistical analysis** to discover patterns
3. **Derive scoring rules** from those patterns
4. Build your model based on what the data reveals

Build Model

Advanced requirement 1: Now build your model in Python. Use these scoring rules {in Approach 1} to:
- calculate total points
- determine risk level.

Hint! Take Screenshots AND Video of Model Working as you go. {Splice Together and Voiceover Later}

Advanced requirement 2: Manual What-if Testing

AR2: What-if Manual Testing

Hint! Take Screenshots AND Video of Model Working as you go. {Splice Together and Voiceover Later}

The Big Idea: **AR2** = You type in data every time (manual)

AR3 = Model reads data automatically and tells you when things change (adaptive)

What Makes AR2 'What-If Testing'?

1. **Set a baseline:** (Baseline hardcoded into model to reflect normal or low risk conditions)

   In a 'what-if' test, we want to predict **what happens if conditions change**. However, you can't predict change unless you know **what the conditions were before.**

Analogy: Imagine you want to lose weight, say 2 kg in a month. But, in a month's time when you weigh yourself, **you can't tell if you gained or lost weight unless you know your starting weight (baseline).**

```
BASELINE: Low Risk Conditions
Example Baseline:  Temperature: 22°C, Humidity: 55%, Soil: damp, Wind: low
Total Points: 1 | Risk Level: 1
```

2. **YOU choose test scenarios** (e.g., What if temp rises to 35°C?)

```
WHAT-IF TEST #1
Temperature (°C): 34
Humidity (%): 12
Soil (wet/damp/dry/very dry): dry
Wind (low/moderate/high): high
```

3. **Compare to baseline** (Did risk increase, decrease, or stay the same?

```
RESULT
Total Points: 8 | Risk Level: 3
Compared to baseline: Risk INCREASED.
```

Why This Meets the Requirement

The brief says two what-if scenario simulations.'

AR2 does this by letting you test at least TWO scenarios (changing temp, humidity, soil, wind) and compare results to the baseline. Examples:

What if temperature rises to 35°C? (heatwave)

What if humidity drops to 15% AND soil becomes very dry? (drought)

What if wind becomes high during dry conditions? (fire spread risk)

Advanced requirement 3 **Add the 'feedback' mechanism.**

Hint! Take Screenshots AND Video of Model Working as you go. {Splice Together and Voiceover Later}

AR3: Making Your Model AUTOMATIC

The Big Idea: **AR2** = You type in data every time (manual)

AR3 = Model reads data automatically and tells you when things change (adaptive)

What Makes AR3 'Adaptive'?

1. **Reads new data automatically** (from CSV file)
2. **Remembers the last risk level** (so it can compare)
3. **Tells you what changed** (Risk went UP or Risk went DOWN)

```
NEW ROW OF DATA READ -> MODEL UPDATED
Time: Day 1 09:00
Temp: 22 Humidity: 55 Soil: damp Wind: low
Total Points: 1 | Risk Level: 1
Risk Comparison with Previous: Risk stayed the same ( 1 )

NEW ROW OF DATA READ -> MODEL UPDATED
Time: Day 1 12:00
Temp: 25 Humidity: 48 Soil: damp Wind: moderate
Total Points: 3 | Risk Level: 2
Risk Comparison with Previous: Risk increased from 1 to 2
```

Note: In AR3 of the brief, I **am 'adjusting predictions based on the most recent data'**.

Why This Meets the Requirement

The brief says your system must **adapt automatically to changing conditions.'**

AR3 in this model does this because: It runs **without you** typing anything

It **notices** when risk changes

It **responds differently** based on what happened

Your CSV Should Have Changes

Make sure your risk goes up and down so the model can demonstrate adaptation.

e.g. Example Progression: `Risk increased from 1 to 2`