

JRMP 2026_Group 09

Highlights

Metric	JRMP 2026_Group 09 Achievements
 Lectures	25+ Core & Guest Lectures
 Workshops	4+ Practical AIoT Sessions
 Assignments	12+ Continuous Assessments
 Conferences	1 International Conference (with Nobel Laureate)
 Competitions	1 National Competition
 Publications	2 Research Publications





2026

Project Code: JRMP2026_09
Research Topic: Smart School Rooms: Using Sensors and MQTT to Track Classroom Environment and Student Movement

Supporting Unit(s):  [Department of Computing](#)
 電子計算學系

Research Team Member(s):
 Dr MOHAMMED Aquil Mirza, Senior Lecturer

Short Description of the Research Project:
 This project aims to explore how sensors—such as motion detectors, temperature sensors, and light sensors—can be used to create a "smart" classroom environment. As a mentee, you will use AIoT (Artificial Intelligence of Things) and MQTT (a messaging protocol) to collect data on classroom conditions and student movement. You will then analyze this data to make the classroom more comfortable, energy-efficient, and safe.

This project is particularly interesting and accessible because:

1. It connects to real-world applications in schools and buildings.
2. It involves simple sensor data collection and basic programming.
3. It introduces key concepts such as IoT communication protocols (MQTT).
4. It is hands-on and can be completed using beginner-friendly tools like Arduino or Raspberry Pi.

Through this project, you will gain practical experience in sensor technology, data analysis, and IoT systems, while contributing to the development of smarter and more sustainable learning environments.

Lecture Topics

JRMP2026_09 Lectures Attendance Records				
Week	Lecture	Day	Date	Topic
1	1	Wed	Apr-01	Prootocls
	2	Fri	Apr-03	Functional Programming
	3	Sat	Apr-04	Python Programming
2	4	Tue	Apr-07	Sensors & MQTT
	5	Wed	Apr-08	Demos of Sensors & MQTT
	6	Th	Apr-09	Data Types and Assignment 1 Discussion
	7	Fri	Apr-10	Embedded Systems Environment Setup
3	8	Mon	Apr-13	How to do research?
	9	Tue	Apr-14	Programming using Microcontrollers
	10	Fri	Apr-17	Assignment 2 Discussion
4	11	Mon	Apr-21	Propositional Logic I
	12	Tue	Apr-22	Microcontroller Programming I
5	13	Mon	Apr-28	Microcontroller Programming II
	14	Tue	Apr-29	Propositional Logic II
	15	Sat	May-02	Type Theory I & Assignment 5 Explain
6	16	Mon	May-05	Dashboard I
	17	Tue	May-06	Programming - Logic
7	18	Mon	May-18	Dashboard II & Assignment 6
	19	Tue	May-19	Programming - Natural Numbers
8	20	Mon	May-25	Setting up Rust Environment
	21	Tue	May-26	Programming - Tuples & Recursion
9	22	Wed	Jun-03	Introduction to Hotaru Framework Research
	22	Sun	Jun-07	Trouble shoot for Dashboard & Rust



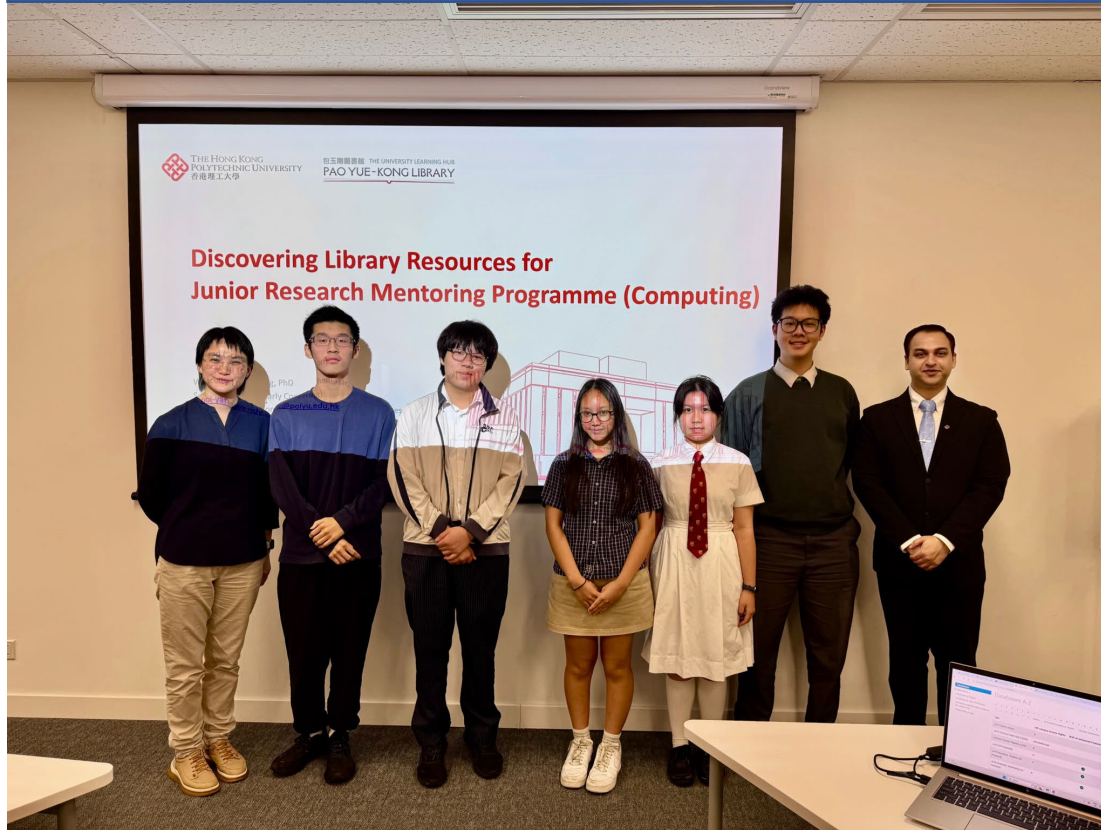
JRMP2026_09 Orientation Day with Associate Director Mr. Frank Chow



ISBDAI Conference with Nobel Laureate Prof. Ferenc Krausz, Germany



PolyU Library Tour and Resources Hands-on-Workshop by Ms. Wendy Hoi Yan (Senior AL)



Robotics & AIoT Plenary Workshop by Sinovated Technology Limited, Hong Kong

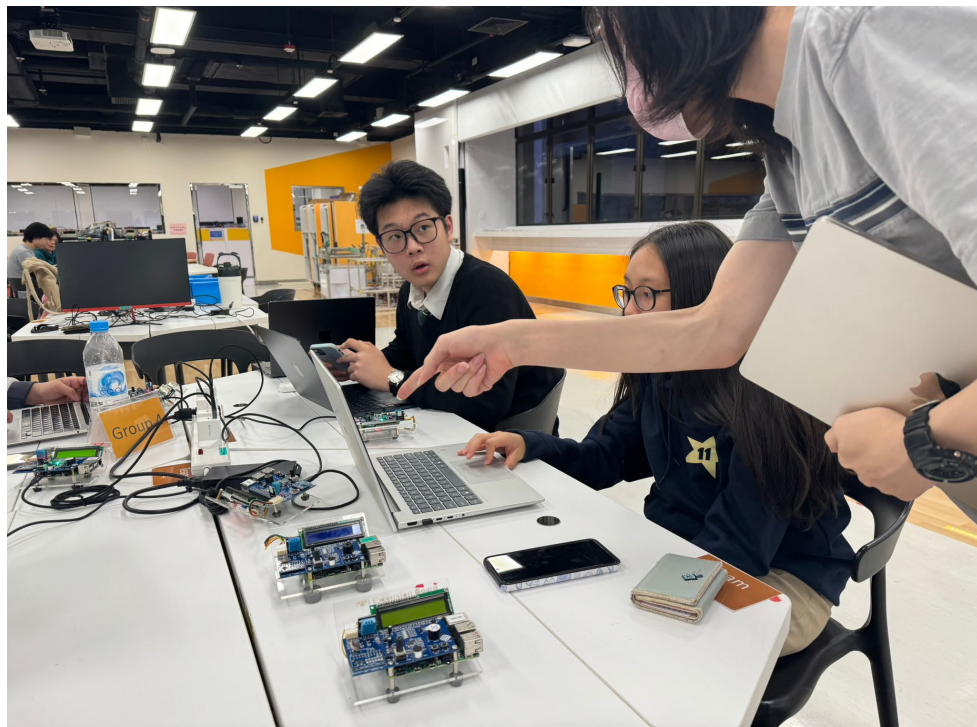


Mentee Luncheon with JRMP_09 PolyU Student Communal Canteen

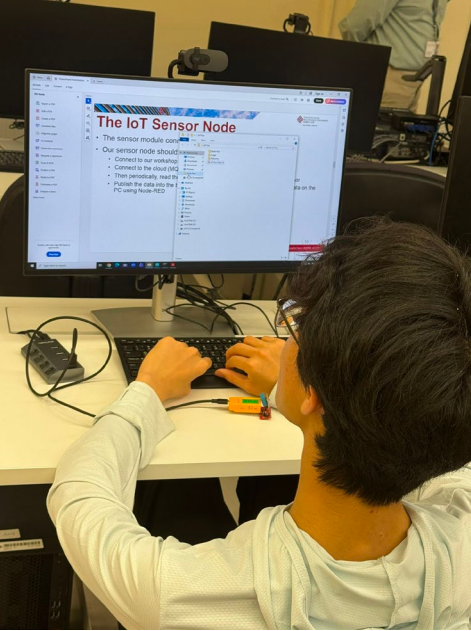


First AIoT workshop at Industrial Centre with Mr. Henry Chan, Engineering Manager





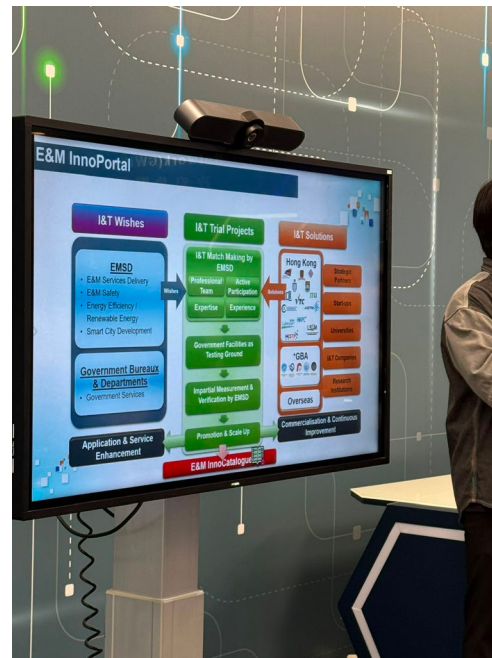
AIoT Lectures



AIoT Workshop



Briefing & EMSD Tour



Student Assignment Submissions

JRMP2026 Worksheet 1: Introduction to Programming Languages

Mentor - Prof. Dr. Mohammed Aquil Mirza, COMP

Student Mentors - Mr. Ruan Haozhe, Mr. Jiang Suyu

The Hong Kong Polytechnic University (PolyU), Hong Kong

Name: LI Wan Chak School: FRCSS

1. Write your understanding of the following terms in layman terms:

- Sensor: A device that detects infos from environment like temperature.
- Actuator: A device that take action base on the commands
- Processor: It process the data from sensors and send action command to actuator

2. Search on the web, tell the difference between HTTP and MQTT protocols. Which one is more suitable for IoT applications? Why?

HTTP is a client-server protocol, only client can send a request to server and server can response

MQTT is more suitable because server can send a data package to the client actively

3. Translate the following lambda expressions into Python code and evaluate them:

- $(\lambda x.x + 1)5$ 5+1=6
- $(\lambda x.\lambda y.x + y)3, 4$ 3+4=7
- $(\lambda x.\lambda y.xy)(\lambda z.z)4$ 4

4. Explain, in layman language, what is extensional and intensional definition of functions.

Extensional function describes a function by listing input-output results

Intensional function describes a function by giving the formula to compute the output by input

5. Why in normal maths circumstance we use extensional definition of functions, however in programming languages we use intensional definition of functions?

In math, it often deals with small sets of data, so listing results is practical

In programming, it usually deals with a large or infinite dataset, so we use formula to compute the result

6. In our Lamp Example before, the type for each message is Tuple (Boolean/Tuple)

7. Define Attribute Error. It occurs when access a attribute or function that doesn't exist.

8. How type signature can help us to avoid Attribute Error?

Type signature specifies what type of data a function expects and returns, so it can help us

to avoid calling attributes that doesn't exists

JRMP2026 Worksheet 2: MQTT, Sensors, Microcontrollers, and MicroPython

Mentor - Prof. Dr. Mohammed Aquil Mirza, COMP

Student Mentors - Mr. Ruan Haozhe, Mr. Jiang Suyu

The Hong Kong Polytechnic University (PolyU), Hong Kong

Name: Lu Zhengxuan School: Shun Tak Fraternal Association

1. Write your understanding of the following terms in layman terms: Seaward Woc College

- Broker: A central message hub for receiving and send messages
- Publisher: A sensor that sends messages
- Subscriber: A device that get messages from the broker
- Topic: A shared type of a specific group of data

2. Why is MQTT better than direct point-to-point communication when many devices are connected?

It is a specialist for receiving from different publishers and disturbing to different subscribers, so it will not cause the disorder of transition.

3. Search on the web, then compare HTTP and MQTT. Which one is more suitable for IoT applications? Why?

MQTT. It is more lightweight, energy-efficient, stable for those situations that need a long-time connection. Besides, HTTP is for websites.

4. In the phone and air-conditioner example from class, fill in the blanks:

- The phone publishes commands to the broker
- The air conditioner publishes status updates to the broker
- The phone is a publisher and also a subscriber
- The air conditioner is a publisher and also a subscriber

5. What is sampling? Why do computers need sampling to understand the physical world?

To capture numerous snapshots or records of an object in the real world. Because they are limited to the their data from the sensors.

6. Match each sensor with its main signal type and what it measures:

- LDR: analog signal, measures the brightness
- DHT11: digital signal, measures the temperature and the humidity
- HC-SR04: digital based reading, measures the distance

7. Explain the jobs of these wires:

- VCC: to provide electricity as power
- GND: to complete the return path of the current
- Data / Signal: to send out the data or signals

JRMP2026-09 Assignment 5: Proof Methods

Mentor - Prof. Dr. Mohammed Aquil Mirza, COMP

Student Mentors - Mr. Ruan Haozhe, Mr. Jiang Suyu

The Hong Kong Polytechnic University (PolyU), Hong Kong

Deadline: 29:59, 3 May 2026 (5:59 AM, 4 May 2026)

Name: Lu Wan Chak School: SWC

1. Explain the following proof methods in layman language, and give an example for each method.

- Direct proof: give direct intuitive conclusion based on the logic
- Proof by contradiction: prove the opposite of what you want to prove, then show it's false
- Proof by induction: produce general principle from a set of facts

2. State the definition of inference rule. State what the Modus Ponens inference rule is. Explain what is deduction theorem.

$$\vdash (A \rightarrow B) \rightarrow B$$

If $A \rightarrow B$ is true, and also A is true, B is true. $\{A, A \rightarrow B\} \vdash B$ or $\frac{A, A \rightarrow B}{B}$

If $A \rightarrow B$ and $B \rightarrow A$, and also B is true, A is true.

3. Prove the following propositions using the listed axioms.

Let $\vdash (A \rightarrow B) \rightarrow (B \rightarrow A)$ be $\odot$ $\vdash (\neg A \rightarrow \neg B) \rightarrow (A \rightarrow B)$
false, $\vdash (A \rightarrow B) \rightarrow B$ is true, $\odot$ $\vdash (A \rightarrow B, B \rightarrow C) \rightarrow A \rightarrow C$
 $(A \rightarrow B)$ is false, Axioms: $\odot$ $\vdash (\neg A \rightarrow \neg B) \rightarrow (B \rightarrow A)$
hence A is true $\odot$ $\vdash (A \rightarrow B) \rightarrow (A \rightarrow C)$
 B is false from $\odot$ $\vdash (A \rightarrow B) \rightarrow ((A \rightarrow B) \rightarrow A)$
 $\odot$ $\vdash (A \rightarrow B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow A)$
put in 1 $\odot$ $\vdash (\neg A \rightarrow \neg B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow B)$ 4 $A \rightarrow C$
(7A) is false $\odot$ $\vdash (A \rightarrow B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow (A \rightarrow B))$
 $\rightarrow B$ is true $\odot$ $\vdash (A \rightarrow B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow (A \rightarrow B))$
 $(\neg A \rightarrow \neg B) \rightarrow B$ is true $\odot$ $\vdash (\neg A \rightarrow \neg B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow (A \rightarrow B))$
there is no contradiction. $\odot$ $\vdash (\neg A \rightarrow \neg B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow (A \rightarrow B))$
proposition $\odot$ is false. $\odot$ $\vdash (\neg A \rightarrow \neg B) \rightarrow (B \rightarrow A)$

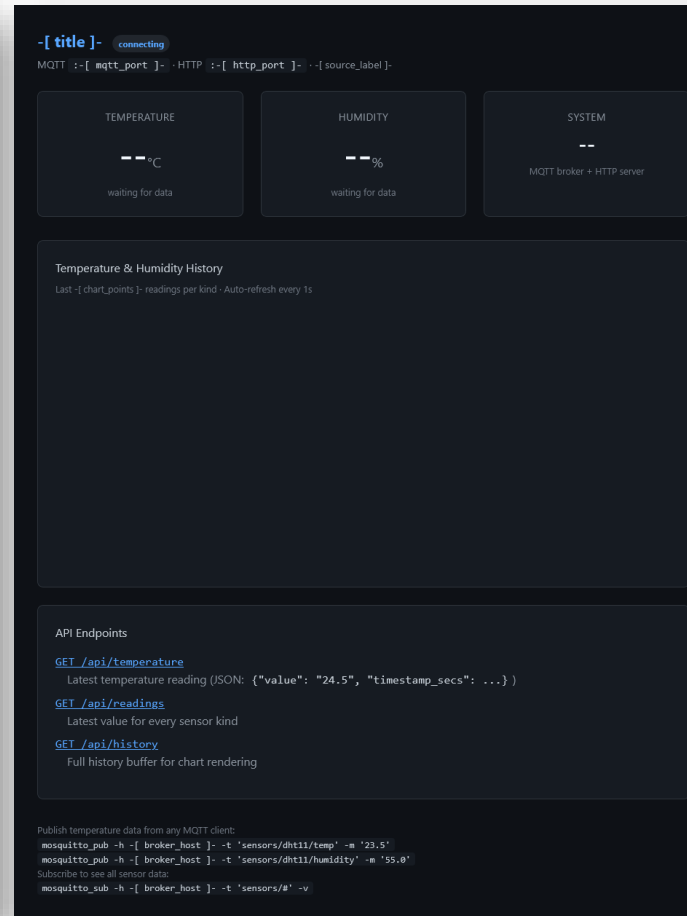
Then, fill in proposition A, B and C with some concrete propositions. Use $\{A \rightarrow B, B \rightarrow C\} \vdash A \rightarrow C$ to prove something in real life.

If tomorrow is a weekday, then I have to go to school.
If I go to school, then I will meet my head teacher.
Therefore, if tomorrow is a weekday, then I will meet my head teacher.



Smart Classroom Implementation

```
sftp.json state.rs X
C: > Users > ammoham > AppData > Local > Temp > f683f68c-241e-4fd3-84ef-9eec5fb8a574_smart_classroom_Eric.zip.574 > s
1  /// Shared in-memory state – ring buffer of readings + latest-per-kind map.
2  ///
3  /// Every sensor publish funnels through `record_reading`, which writes to
4  /// both HISTORY (for the chart) and LATEST (for real-time display).
5
6  use std::collections::{HashMap, VecDeque};
7  use std::sync::Arc;
8  use std::time::{SystemTime, UNIX_EPOCH};
9
10 use hotaru::Value;
11 use hotaru::object;
12 use once_cell::sync::Lazy;
13 use tokio::sync::Mutex;
14
15 use crate::config::HISTORY_CAP;
16
17 /// Ring buffer of all historical readings (newest at back), capped at HISTORY_CAP.
18 pub static HISTORY: Lazy<Arc<Mutex<VecDeque<Value>>>> =
19     Lazy::new(|| Arc::new(Mutex::new(VecDeque::with_capacity(HISTORY_CAP))));
20
21 /// Latest value per kind – avoids scanning HISTORY for real-time display.
22 pub static LATEST: Lazy<Arc<Mutex<HashMap<String, Value>>>> =
23     Lazy::new(|| Arc::new(Mutex::new(HashMap::new())));
24
25 /// Unix epoch seconds (best-effort; 0 on clock failure).
26 pub fn now_secs() -> u64 {
27     SystemTime::now()
28         .duration_since(UNIX_EPOCH)
29         .map(|d| d.as_secs())
30         .unwrap_or(0)
31 }
32
33 /// Build a reading entry as a `Value::Dict`.
34 pub fn make_entry(kind: &str, value: &str) -> Value {
35     let ts = now_secs();
36     object!({
```



```
File Edit Selection View Go Run Terminal Help
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
classroom_state.json readings.json mains X Release Notes: 1.126.0
C: > Users > ammoham > AppData > Local > Temp > 5921ce81-4b96-4e3a-8fb2-649d06df0bd9_smart_classroom_Eric.zip.bd9 > smart_classroom_Eric > src
1  /// Smart Classroom one binary hosting MQTT broker, HTTP dashboard,
2  /// DHT11 sensor loop, and LCD driver.
3  ///
4  /// This is the answer submission for the three questions:
5  ///
6  /// Q1: LCD display data flow (in/out) from the broker see `lcd.rs`
7  /// Q2: Endpoint/output for temperature data see `handlers.rs`
8  /// Q3: Dashboard to display temperature data see `handlers.rs` + `templates/dashboard.html`
9  ///
10 /// Architecture:
11 ///
12 /// sensors.rs publish BROKER MQTT_APP endpoint! handlers.rs (state)
13 ///
14 /// LCD_CLIENT endpoint! lcd.rs (hardware)
15 ///
16 /// External MQTT subscribers (port 1883)
17 ///
18 /// HTTP clients (browser) HTTP_APP endpoint! handlers.rs LATEST/HISTORY
19
20 classroom_state.json readings.json mains X Release Notes: 1.126.0
C: > Users > ammoham > AppData > Local > Temp > 5921ce81-4b96-4e3a-8fb2-649d06df0bd9_smart_classroom_Eric.zip.bd9 > smart_classroom_Eric > src
42 async fn main() {
43     persistence::load_from_disk().await;
44
45     // 2. Print banner.
46     println!("{}", Smart Classroom (Answer Submission));
47     println!("{}", MQTT broker: {}, MQTT_BIND);
48     println!("{}", HTTP dashboard: http://{}/, HTTP_BIND);
49     println!("{}", Temperature API: http://{}api/temperature/, HTTP_BIND);
50     println!("{}", JSON API: /api/readings /api/history");
51     println!("{}", Flush every: {}, FLUSH_INTERVAL_SECS);
52     #[cfg(feature = "rpi-hardware")]
53     println!("{}", Sensors: real DHT11 on BCM GPIO{}", crate::config::DHT11_GPIO_BCM);
54     #[cfg(not(feature = "rpi-hardware"))]
55     println!("{}", Sensor: SIMULATED (build with --features rpi-hardware for real DHT11)");
56     println!("{}", );
57
58     // 3. Start the DHT11 sensor loop (real or simulated).
59     sensors::spawn_loop();
60
61     // 4. Start background drivers.
62     tokio::spawn(lcd::start_session());
63     tokio::spawn(persistence::flush_loop());
64
65     // 5. Run both servers until Ctrl+C.
66     tokio::select! {
67         _ = MQTT_APP.clone().run() => {}
68         _ = HTTP_APP.clone().run() => {}
69         _ = tokio::signal::ctrl_c() => {
70             println!("{}", Shutting down... flushing to disk");
71             persistence::flush_to_disk().await;
72         }
73     }
74 }
75 }
```

