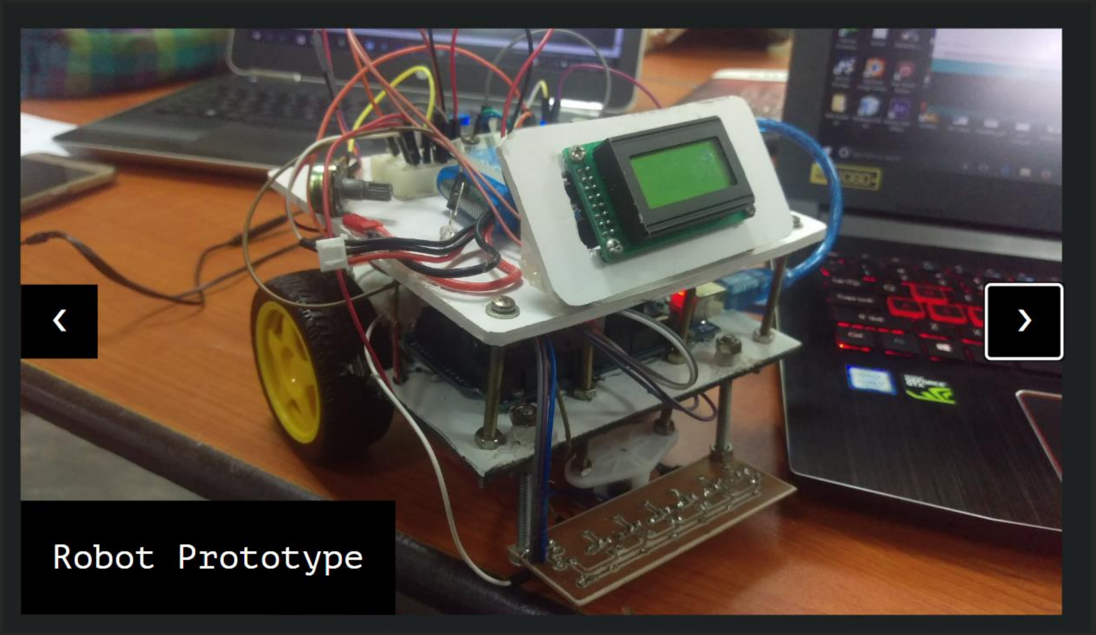
Orientation 2026

Session 01 : Introduction to the Program

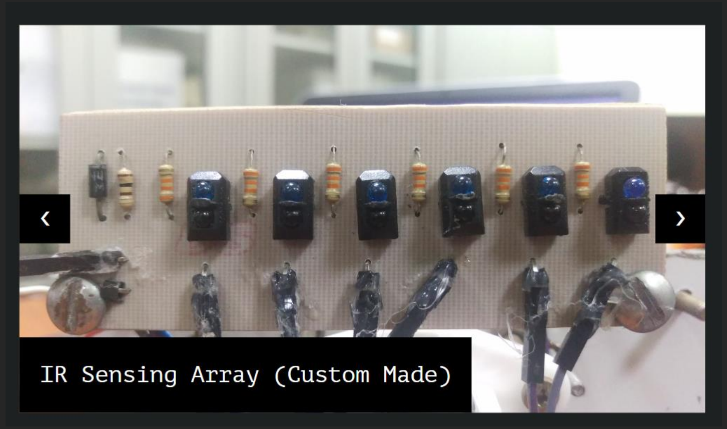
25 YEARS OF EXCELLENCE
SLIIT UNI
THE KNOWLEDGE UNIVERSITY

Amila Alexander

// About Me



Robot Prototype



IR Sensing Array (Custom Made)

- *BSc. Robotics and Mechatronics (RMIT)*
- *MSc. (Reading) Software Engineering (Kingston University, UK)*

- Passion for programming and solving tough to solve problems 🧡

Welcome to the Program

- Hands-on, creative and in class activity based
- Program is designed for students with **little or no coding experience**
- **Drag and drop** tools and **visual programming** tools prioritized

Program Structure

- 2 hrs each week
- Weekly hands-on work (practicals)
- 1–2 mini-assignments (homework)
- Final project involves creating a website (using NoCode tools)

What You’ll Learn

- Web development (“*NoCode*” based)
- UI/UX design
- Programming logic (visual first – using MIT Scratch)
- Artificial Intelligence
- Microcontrollers & IoT (optional)
- Modern tech skills for life and career

//What you’ll need

- BYOL (Bring Your Own Laptop)
- Things will involve ~~**No**~~ some Coding
- Problem solving mindset

The Problem Solving Mindset

Think about what an engineer does

- **Solves problems**
- Problem is presented
- Consider the following problem

The Problem

- Need a way of crossing
- Potential solutions?
    - Bridge?
    - Boat
    - Barge
    - Jet ski?
    - Plane
- Think of the potential benefits and downsides of the following solutions

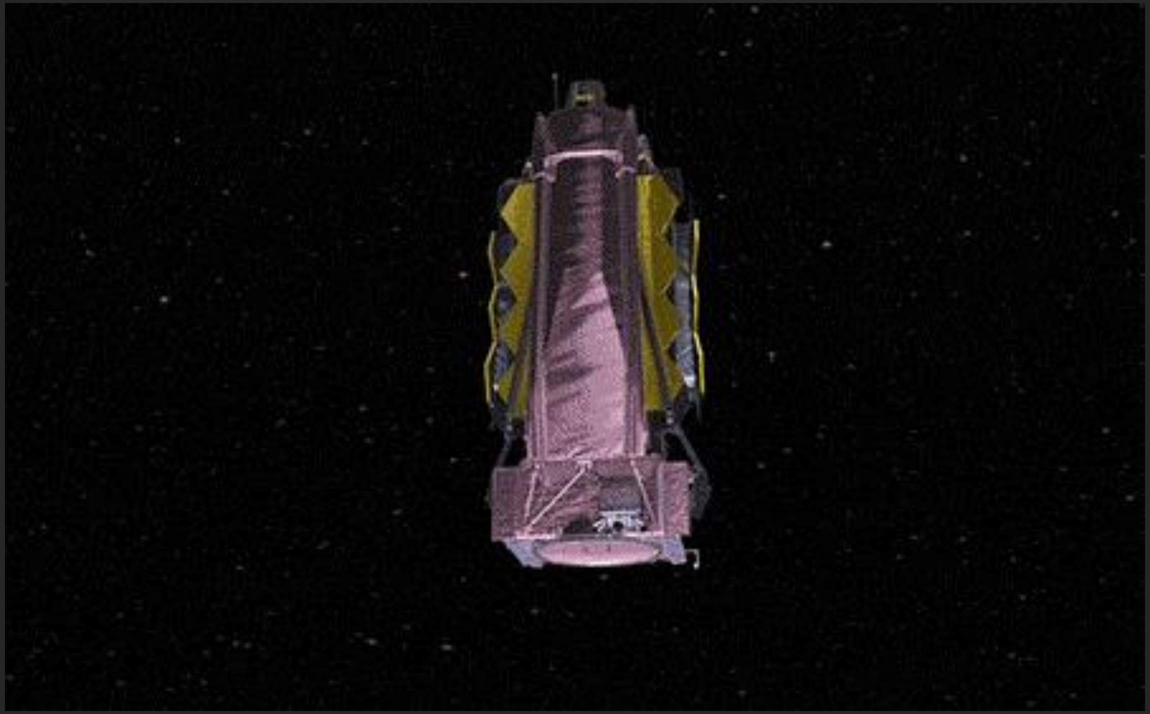
The Problem Solving Mindset

Think about what an engineer does

- **Solves problems**
- Problem is presented
- Thinking starts and an idea is born
  - Questions
    - Is this idea cost effective?
    - Can this actually be done?
    - How reliable is this solution?
- Uses **logical thinking + creativity**

“Even without knowing how to code, you can still *think* like an engineer. This mindset will help you learn any technology faster.”

Real World Example : JWST



- “..over 300 are “single points of failure”. Meaning that if any one of them fails then the telescope as a whole will fail…”
- 10B $ USD
- How do you build something to work 100% of the time?
- You can’t
- You can only test, test, test and build redundancy into your system

These problems have very specific “*ENGINEERING*” requirements

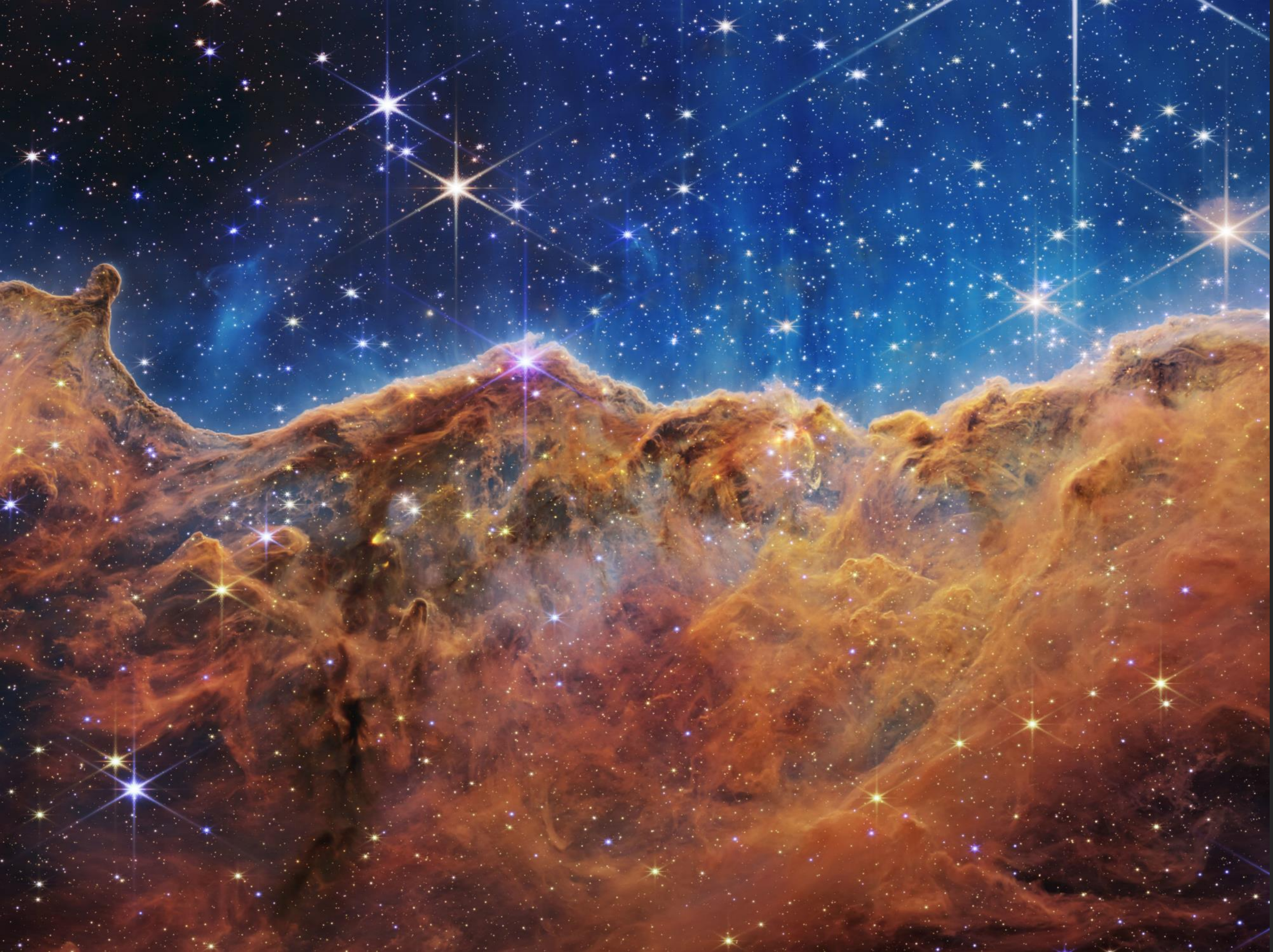
Quality of components

- Can my design afford to fail?
- Do I send someone to mars to fix it? 🤔
- $ 1 Billion to fix an incorrect design

It all starts with people ;
(*very smart people*)

- Over 1200 skilled scientists
- Engineers to develop, test and research the design
- Suppliers (because not a single person can make everything)
- Launch site crews (ESA)
- Not to mention the 1000s of people who built to the rocket

Results of Hard Work



Cosmic Cliffs

JWST
NASA

The Knowledge Gap

- Everyone is born knowing nothing
- People learn… 🧠
- What most people learn is consistently insufficient for the task at hand.

But

- It’s good enough

- There’s always someone with greater knowledge, but in their absence, you hold that responsibility.

Bridging the Gap

- The modern world is vastly different than it was 10 years go

- 20 years ago things were very different

- The transition
- Age of Artificial Intelligence
- Learning is not about finding answers, it’s about discovering them

20 years ago : 2005

- **Then**: Smartphones were a novelty—Apple’s first iPhone hadn’t even launched yet (that came in 2007). Flip phones ruled the day, and texting was a skill.
- **Now**: We carry pocket-sized supercomputers. AI assistants, augmented reality, and cloud computing are integrated into our daily lives.

📺 Entertainment & Media:

- **Then:** DVDs, CDs, and cable TV were mainstream. YouTube had just launched (2005), and Netflix was still a DVD-by-mail company.
- **Now:** Streaming dominates. Short-form video culture (think TikTok, Reels) shapes trends, and content is on-demand and global.

📞 Communication:

- **Then:** Email, SMS, and phone calls were primary. Social media was in its infancy—Facebook was just opening to the public.
- **Now:** Instant messaging, video calls, and social media platforms have redefined how we connect.

💼 Work & Education:

- **Then:** Office cubicles and chalkboards. Remote work and virtual classrooms were rare.
- **Now:** Hybrid work is common. AI and digital tools support everything from business strategy to classroom learning.

Program Content

You are here

- Introduction to the Program and Computing
- Introduction to Web Dev
- UI/UX Principles
- Building First Website : NoCode
- Working with the Backend (Web)
- Intro to Programming (via Scratch)
- Scratch Programming II + Game Development
- Skills for Modern Computing

THANK YOU