“Arduino” track competition

CodeRun 2025, Cluj-Napoca

Safety

⚠️ Inform us ASAP if you suffer from any kind of **PHOTOSENSITIVE EPILEPSY.** There will be bright quickly-flashing lights, which we can hide.

- We are working with real hardware. Be careful so you don’t hurt yourselves.
- Please take care of the kits, don’t destroy them

Competition Rules

🤖 **AI? OK**, but please don’t just feed it the whole task and let it autopilot. The tasks should be doable in 2h without AI assistance. If you solve using AI, there is no penalty, but I’ll ask you to show me how.

Solution grading

Solutions will be graded here, live, during the competition. Bring them to the front.

You may come test your solution as many times as you want, without penalty.

You get points if your solution works at least once.

The two challenges are scored separately.

If time doesn’t allow for testing, upload the code to Moodle, as well as enough info for me to be able to build and test it afterwards.

Problem statement

Following the latest robbery of the Louvre, the French government have installed new electronic locks in semi-critical parts of the museum. We have managed to get our hands on one unit and a small fragment of its documentation. It seems… shoddy.

Your task is to build an automated safe cracker tool to bypass this electronic vault lock.

We have to go in 2 hours, there’s not much time!

Kit contents

- Arduino Uno
- Breadboard
- Wires
- MPU6050 Inertial Measurement Unit (accelerometer + gyroscope)
- Light sensor (CdS photoresistor based)
- 180-degree Servo (Futaba S3003 clone)
- Small screwdriver (not plenty for everyone, pls share)
- Electronic lock datasheet

The challenges

Manual mechanical input (100 points)

Build an automatic safe cracker that listens for vibrations inside the lock to coordinate its movements.

-50 points if not autonomous

PulsedLight Unlock Protocol (100 points)

Build an automatic safe cracker that implements the PulsedLight Unlock Protocol in Challenge-Response mode.

-50 points if not autonomous

Easy = solve either one

Hard = solve both