IDP: Code Structure and Algorithms

**GitHub Repo:** <https://github.com/arnav5011/IDP_M11>

**Initialisation**: Configures hardware components (motors, sensors, LEDs, servo) and sets initial states.

* + void setup();
* **Main Control Loop**: Manages button input to toggle AGV operation and coordinates the overall workflow.
  + void loop();
  + void run();
* **Line Following**: Reads inputs from four line sensors to adjust motor speeds and maintain alignment with the path.
  + void line\_follow(int extreme\_right, int right, int left, int extreme\_left);
  + void turn\_direction(int input);
  + void shift\_direction(int input);
* **Object Detection**: Utilises a time-of-flight sensor to detect objects and magnetic sensors to classify them as recyclable or landfill.
  + void detect\_object();
  + void check\_magnetism();
* **Path Planning**: Implements pre-defined paths based on junction counts (splits, lefts, rights) to navigate efficiently, returning to common nodes for simplified routing.
  + void path\_object\_**N**();
* **Servo Control**: Operates the servo motor to pick up or deposit objects based on object detection and classification.
  + void moveServo(int targetPosition();
  + void servo\_open();
  + void servo\_close();
* **LED Handling**: Flashes a blue LED during motion (2Hz ±10%) and uses red/green LEDs to indicate object type (magnetic or non-magnetic).
  + void handleLEDs();
* **Return-to-Start Routine**: Ensures the AGV navigates back to the start/finish box after completing its tasks.

A diagram of a path

Description automatically generated

Figure 1: Simplified Flowchart