LOUDOUN ROBOTICS

ROBOTICS EDUCATION FOR EVERY STUDENT IN LOUDOUN COUNTY

25343 Kinsale Pl, Aldie, VA 20105 aravind@loudounrobotics.org · 703-932-9934 · loudounrobotics.org

Instructor Lesson Plans — Fall 2026

Stone Hill Middle School · Wednesdays 6:00–7:00 PM · Sep 16 – Dec 9

Students 15, in fixed teams of three or more **Kits** LEGO MINDSTORMS EV3 **Session** 60 minutes
Ends with Sumo tournament, Dec 9

Every session runs the same shape: robots out, **guess first**, test, write it down, pack up. The guess is not decoration — it is the part that makes the result mean something. A student who predicted the wrong answer and found out why has learned more than one who simply watched it work.

The rule that holds the whole thing together

Before any test, students write down what they think will happen. After the test, they write what did. When those disagree, that gap is the lesson. Protect the two minutes this takes — it is the first thing that gets skipped when a session runs late, and it is the last thing that should.

Standing setup

- Arrive 15 minutes early. Kits on tables, one per team, before students walk in. Three students to a team is the minimum, so 15 students means no more than five teams.
- Robots stay assembled all term. Each team labels theirs and it goes back on the same shelf.
- A 1.2 m circle of black electrical tape on a light floor is the sumo ring. Put it down week 1 and leave it.
- Laptops are only needed from week 5 onward.
- **Roles rotate every week.** Every team has at least a builder, a programmer, and a tester — that is why three is the minimum. A fourth student takes notes and swaps in. Call the swap out loud at the start of the session — left to themselves, the same student builds for twelve weeks and the other two watch.

If a student misses a week

Nothing is torn down, so their robot is waiting for them. Give them the notebook pages they missed and put them with a team that did the build. The concepts stack, but no single week is load-bearing except week 4, when they commit to a design.

Two adults, two high-school assistants

Adults handle the room and the clock. Assistants circulate during build and test time — their job is to ask "what did you predict?" before they help, not to fix the robot. A student whose robot gets fixed for them learns nothing; a student who gets asked one good question usually fixes it themselves.

WEEK 1 · SEPTEMBER 16

Build and test

By the end: every team has a driving robot and knows exactly how far off “one meter” really was.

BEFORE STUDENTS ARRIVE

- Kits out, one per team. Building instructions printed or on a tablet at each table.
- Tape a straight 1 m line on the floor with a clear start mark. Tape the sumo ring down too — students will ask about it, which is useful.
- Notebooks and pencils at each seat. Tape measures, one per two tables.
- Build one robot yourself beforehand. You will need it for the demo and as a loaner.

RUN OF SHOW

0:00	**Welcome and form teams.** Hand out notebooks. Names on the cover. Keep this short.
0:05	**The demo.** Run your robot and tell them it is programmed to drive one meter. Before it moves: “How close will it get?” Take guesses out loud, write three of them on the board. Then run it and measure. It will not be exact. That surprise is the hook for the whole term.
0:12	**Build.** Driving base, from the instructions. Circulate. Resist fixing anything — point at the step number instead.
0:42	**Predict, then test.** Each team writes their guess in the notebook first. Then three runs down the 1 m line, measuring actual distance each time. Record all three.
0:53	**Notebook.** Guess versus actual. Why were the three runs different from each other?
0:58	**Pack up.** Label the robot with the team's name. It stays built.

> **What usually goes wrong**
>
> - **A team is still building at 0:42.** Lend them your demo robot so they still get to test. The measurement matters more than finishing the build tonight.
> - **Wheels slip and distances scatter wildly.** Do not fix it. Point it out and say it is next week's problem. That is exactly the setup for week 2.
> - **Students race to finish and skip measuring.** Redirect firmly — the measuring is the session, the building is just how you get there.

Files `ev3-rem-driving-base` · **Notebook** Weeks 1 pages · **Laptops** not needed

WEEK 2 · SEPTEMBER 23

Grip and strength

By the end: students can say which surface grips best and prove it with their own numbers.

BEFORE STUDENTS ARRIVE

- Robots out on tables from last week.
- Test surfaces: bare floor, a carpet square, a sheet of paper, a strip of masking tape laid down flat.
- Spare tires and treads from the kits, sorted into a tray.
- Something to tow — a book or a taped-shut box with a bit of weight in it, plus string.

RUN OF SHOW

0:00	**Robots out.** Recap: read out two teams' distance errors from last week. Ask why they were not zero.
0:06	**Guess first.** Show the four surfaces. Each team writes down which one they think the robot will pull hardest on, and which it will slip most on. No discussion yet.
0:12	**Traction test.** Drive the same distance on each surface, measure the error. Four surfaces, one run each, all recorded.
0:32	**Tow test.** Tie the weighted box on. Which surface lets the robot drag it? Add weight until it fails.
0:46	**Swap the tires.** Same tow test with a different wheel or tread. One change at a time.
0:53	**Notebook.** Did your guesses match? Where were you wrong, and what would you guess now?
0:58	**Pack up.**

What usually goes wrong

- **Teams change two things at once** — new tires and a new surface — then cannot say which mattered. Catch this early and make them redo one run.
- **The tow box is too heavy to move at all.** Start light. A robot that fails on every surface teaches nothing.
- **"Grippier is always better."** Push back — ask what happens to turning. Save the answer for week 3.

Files none new · **Notebook** Week 2 pages · **Laptops** not needed

WEEK 3 · SEPTEMBER 30

Speed or strength?

By the end: students can predict which gear setup wins a tug of war, and explain the trade in their own words.

THE THREE SETUPS

All three gear pairs add up to 48 teeth, so the axle spacing is **3 studs** for every one of them. Build the bracket once and swap gears in and out.

1 : 1	24-tooth driving 24-tooth. The baseline — same as no gearing.
1 : 3	12-tooth driving 36-tooth. Slow and strong. Wins the tug of war.
3 : 1	36-tooth driving 12-tooth. Fast and weak. Wins the sprint, loses the pull.

RUN OF SHOW

0:00	**Robots out.** Hold up the three gear pairs where everyone can see them.
0:06	**Guess first.** Which pulls hardest? Which is fastest? Every team writes both answers before anyone builds.
0:12	**Build the gear attachment.** Then fit the 1:1 pair.
0:28	**Tug of war.** Robots back to back, joined with string. Run each gear pair against each other pair. Record who won and who was dragged.
0:42	**Speed run.** Same three setups, timed over 1 m. Record the times.
0:52	**Notebook.** Were your predictions right? What surprised you? Which setup would you pick for a sumo match — and why is that a harder question than it looks?
0:58	**Pack up.** Keep the gear pieces with each team's kit.

> **What usually goes wrong**
>
> - **Gears grind or skip.** Almost always the wrong axle spacing. Check it is 3 studs and the beams are parallel.
> - **Everyone wants 1:3 for the tournament.** Good — but ask how they will get across the ring before the other robot does. Leave the tension unresolved; week 4 is where they decide.
> - **Tug of war turns into a wrestling match.** Keep the string short and the robots straight, or you are testing something else entirely.

Files `gear-building-instruction` · **Notebook** Week 3 pages · **Laptops** not needed

WEEK 4 · OCTOBER 7

Design your fighter

By the end: every team has committed to a design and can say what they traded away to get it.

This is the pivot of the term. The robot they finish tonight is the one they keep, program, and bring to the tournament on December 9. Say that out loud at the start.

BEFORE STUDENTS ARRIVE

- Sumo ring taped down and clear of tables. Print the ring rules, one per team.
- All gear pieces, tires, and spare beams out on a shared parts table.

RUN OF SHOW

0:00	**The tournament.** Explain the rules: two robots start on opposite edges, first one pushed fully out of the ring loses, matches are 60 seconds, a draw goes to a rematch. Take questions.
0:10	**Sketch before bricks.** In the notebook: what gear ratio, what tires, what shape, and one sentence on why. Nobody touches a kit for these ten minutes.
0:20	**Build.** This is the longest uninterrupted build of the term. Assistants circulate and ask questions rather than giving answers.
0:48	**Shakedown matches.** Thirty seconds each, no scoring, no bracket. Purely to find out what falls off.
0:56	**Notebook.** What broke? What will you change?
0:58	**Label and store.** This robot has a name now.

> **What usually goes wrong**
>
> - **Teams copy the fastest robot in the room.** Fine, as long as they can say why. Ask them — if they cannot answer, send them back to their sketch.
> - **Someone builds a huge robot that cannot turn.** Let the shakedown matches teach it. Do not pre-empt the lesson.
> - **A team wants to start over at 0:50.** No. They can revise in week 11. Committing to something imperfect is part of the exercise.

Files none new · **Notebook** Week 4 pages · **Laptops** not needed

WEEK 5 · OCTOBER 14

Motors and movement

By the end: every team can make their robot drive a set distance and turn a set angle, on purpose.

First night with laptops. Budget more setup time than you think you need — software launches, cables, and pairing will eat the first fifteen minutes if you let them.

BEFORE STUDENTS ARRIVE

- Laptops out, charged, software already open. Log in ahead of time on every machine.
- USB cables at every seat. Test one full connect-and-download before students arrive.
- Floor marked at 1 m again, plus a taped 90° corner.

RUN OF SHOW

0:00	**Connect.** Robot to laptop, download a program that does nothing but beep. Everyone gets a beep before anyone moves on.
0:12	**Three ways to move.** Seconds, rotations, degrees. Demo all three doing roughly the same thing.
0:20	**Guess first.** How many rotations to cross 1 m? Write the number down before testing. Then test and adjust.
0:35	**Turns.** Same exercise for a 90° turn. Predict the degrees, test, correct.
0:48	**Square challenge.** Drive a square and come back to the start. Nobody lands it exactly — that is the point.
0:55	**Notebook.** Your first guess versus your final number. How many tries did it take?

> **What usually goes wrong**
>
> - **Software or driver trouble on one laptop.** Have a spare. Do not spend the session debugging in front of fifteen students.
> - **Students change the program without saving or downloading** and conclude it does not work. Teach download-then-run as a single habit tonight.
> - **Seconds gets used for everything** because it is easiest. Push toward rotations — battery level changes what a second means.

Files `ev3-medium-motor-driving-base` · **Notebook** Week 5 pages · **Laptops** required

WEEK 6 · OCTOBER 21

Loops

By the end: students can make a robot repeat something without copying and pasting the same blocks four times.

RUN OF SHOW

0:00	**Connect and recap.** Reload last week's square program.
0:08	**The problem.** Show the square program — eight blocks, the same two repeated four times. Ask what happens if they want a hexagon.
0:15	**Rebuild it with a loop.** Two blocks inside a loop set to four. Same behaviour, quarter the blocks.
0:25	**Guess first.** What shape will a loop of 3 make? Of 6? Write the predicted shape and turn angle before running it.
0:35	**Shape challenge.** Triangle, hexagon, and one shape of their choosing. One loop each.
0:48	**Forever loops.** Introduce a loop that never ends — this is how the sumo program will be shaped.
0:55	**Notebook.** Draw your loop. What is inside it, and how many times does it run?

What usually goes wrong

- **Turn angle for a triangle trips everyone up** — students say 60° and get a mess. Let them run it wrong first, then work out why it is 120°.
- **A forever loop cannot be stopped** and a student panics. Show the stop button before you show the loop.
- **Blocks get nested by accident.** Zoom in and check placement when a program behaves strangely.

Files none new · **Notebook** Week 6 pages · **Laptops** required

WEEK 7 · OCTOBER 28

Touch sensor

By the end: the robot reacts to hitting something instead of driving blindly into it.

RUN OF SHOW

0:00	**Build the sensor on.** Mount the touch sensor facing forward.
0:15	**Read it first.** Program the robot to beep while the sensor is pressed. Do nothing else until that works — it proves the wiring and the port.
0:24	**Guess first.** If the robot drives forward until the sensor is pressed, then backs up, how far past the wall will it end up? Write a number.
0:32	**Wait-for-touch.** Drive forward, wait for press, reverse. Test against a wall and measure.
0:45	**Robot versus robot.** Two robots facing each other, both running the program. What happens?
0:54	**Notebook.** Your predicted stopping point versus actual. Why the difference?

What usually goes wrong

- **Sensor is on the wrong port** and nothing happens. Check the port number in the block matches the cable.
- **Robot stops well past the wall** and students call it broken. It is not — it is momentum, and it is a real thing they will fight in the tournament.
- **The sensor gets knocked off in testing.** Mount it low and braced; it will take hits in December.

Files `ev3-touch-sensor-driving-base` · **Notebook** Week 7 pages · **Laptops** required

WEEK 8 · NOVEMBER 4

Color sensor — staying in the ring

By the end: the robot detects the ring edge and backs away from it on its own.

This is the single most important programming session of the term. A sumo robot that drives itself out of the ring loses every match, no matter how strong it is.

RUN OF SHOW

0:00	**Mount the sensor facing down**, close to the floor at the front of the robot.
0:15	**Read the numbers.** Display the reflected-light value on the screen. Roll the robot slowly across the tape and watch it change. Write down the value on the floor and the value on the tape.
0:26	**Guess first.** What number should count as “this is the edge”? Every team picks a threshold and writes down why.
0:34	**Edge escape.** Drive forward; when the value crosses the threshold, reverse and turn. Test it by driving straight at the tape.
0:48	**Survive the ring.** Put the robot in the ring on a forever loop. Time how long it stays in. Aim for 60 seconds.
0:55	**Notebook.** Your threshold, the two readings, and how long the robot survived.

What usually goes wrong

- **Room lighting shifts the readings.** Have them re-read the values at the start of every session from here on. This is a genuinely good lesson, not an annoyance.
- **Sensor mounted too high** reads mush. It wants to be a few millimetres off the floor.
- **The robot detects the edge but reverses too slowly** and rolls out anyway. Momentum again — connect it back to week 7.

Files `ev3-rem-color-sensor-down-driving-base` · **Notebook** Week 8 pages · **Laptops** required

WEEK 9 · NOVEMBER 11

Ultrasonic sensor — finding an opponent

By the end: the robot can find another robot and drive at it.

RUN OF SHOW

0:00	**Mount the ultrasonic sensor** facing forward, above the color sensor.
0:14	**Read the distance.** Show the value on screen. Walk a hand towards it and watch the number fall.
0:22	**Guess first.** At what distance should the robot decide “that is my opponent, charge”? Pick a number, and predict what happens if it is set too large.
0:30	**Search and charge.** Spin slowly; when something comes within the threshold, drive at it.
0:45	**Test against a real robot**, not a hand. Hands are the wrong shape and size.
0:55	**Notebook.** Your threshold and what went wrong when you set it too far.

What usually goes wrong

- **The robot charges at the wall, a table leg, or a student.** Exactly why the threshold matters. Let them see it, then tune it.
- **Two ultrasonic sensors confuse each other** in close quarters. Real effect, worth naming — it will show up in the tournament.
- **Angled surfaces read as far away.** A wedge-shaped opponent is genuinely harder to see, which is a design insight for week 11.

Files `ev3-ultrasonic-sensor-driving-base` · **Notebook** Week 9 pages · **Laptops** required

WEEK 10 · NOVEMBER 18

If / else — the robot decides

By the end: one program that stays in the ring *and* hunts — the complete sumo brain.

Everything from weeks 5 to 9 comes together tonight. There is no session on November 25 — Thanksgiving — so students have two weeks to think about what they built here.

RUN OF SHOW

0:00	**Recap the two behaviours.** Edge escape from week 8, search and charge from week 9. Both work. Neither is enough alone.
0:10	**Guess first.** Which should the robot check first each time round the loop — the edge or the opponent? Write down an answer and a reason.
0:18	**Build the decision.** Inside a forever loop: if the edge is detected, back up and turn; otherwise, if something is close, charge; otherwise, keep searching.
0:38	**Full ring test.** One robot alone, 60 seconds. Then two robots.
0:50	**Tune.** One change at a time, retest after each.
0:56	**Notebook.** Which check came first and why. What did you change and what did it fix?

What usually goes wrong

- **Opponent checked before the edge** — the robot charges gloriously out of the ring. The best possible demonstration of why order matters. Let it happen.
- **Students change five things then cannot say what helped.** Enforce one change per test, hard.
- **The program works alone but fails against another robot.** Being pushed is different from driving. Worth naming out loud.

Files none new · **Notebook** Week 10 pages · **Laptops** required

WEEK 11 · DECEMBER 2

Design the edge

By the end: every robot has one deliberate improvement, tested against a real opponent.

Last working session. Nothing new is taught — this is entirely about using what they already know. Remind them the tournament is next week.

RUN OF SHOW

0:00	**Scouting matches.** Ten minutes of open matches, no scoring. Everyone watches everyone.
0:12	**Pick one weakness.** In the notebook: what beat you, and what one change would fix it? One change only.
0:20	**Build or reprogram.** Wedges, wider stance, lower front, different gear ratio, tuned thresholds. Their call.
0:42	**Test the change.** Same opponent that beat them earlier, if possible.
0:52	**Notebook.** Did it work? What is still weak?
0:57	**Store carefully.** Nothing gets modified between now and the tournament.

> **What usually goes wrong**
>
> - **A team rebuilds from scratch at 0:20.** Stop them. There is no time to recover from a failed rebuild before next week.
> - **Wedges get built so low they scrape** and the robot cannot move. Test on the actual ring surface.
> - **Everyone forgets the color sensor threshold changed** when they moved the front of the robot. Prompt them to re-read the values.

Files none new · **Notebook** Week 11 pages · **Laptops** required

WEEK 12 · DECEMBER 9

Sumo tournament

By the end: every student has competed, and every student leaves with their notebook.

Parents are invited. Plan for a room with adults standing around the edges, and for matches to take longer than you expect.

BEFORE ANYONE ARRIVES

- Ring taped down and clear. Chairs pushed back for standing room.
- Round-robin grid drawn on a whiteboard where everyone can see it — every team plays every other team.
- Every robot charged. Plug them in as soon as students arrive.
- Certificates printed and signed in advance.

RUN OF SHOW

0:00	**Setup and charge.** Robots out, thresholds re-read under tonight's lighting. This step is not optional — the room is full of people and the light has changed.
0:10	**Rules, out loud, to everyone.** Sixty-second matches, fully out of the ring loses, no touching the robot once the match starts.
0:15	**Round robin.** Every team plays every other team. Keep matches moving — call a draw and move on rather than letting one match eat five minutes.
0:42	**Final standings and the design award.** Awards go to design as well as to winning — name the team whose build choices were smartest, not just strongest.
0:50	**Certificates and last notebook page.** What would you build differently if you started again?
0:58	**Robots come apart.** Kits back to Loudoun Robotics. Notebooks go home with students.

> **What usually goes wrong**
>
> - **A robot fails early.** Round robin already protects against this — nobody is knocked out, everyone keeps playing. Say so out loud before round one so no one thinks one loss ends their night.
> - **Lighting from a crowded room breaks edge detection.** The 0:00 re-read is what prevents this. Do not skip it.
> - **Matches run long and you lose the ending.** Protect the last ten minutes — certificates and the closing notebook page matter more than finishing every match.

Files none · **Notebook** Closing page · **Laptops** for emergency fixes only