

Lesson 3 Complete Teaching Packet

Level 1 C++ Foundations Input and Decisions

Lesson 3 is the final pure desktop C++ readiness lesson before the course moves toward ESP32. Students collect terminal input, check conditions, and use if else logic to make a robot readiness decision.

Teacher Quick Start

Item	Details
Time	60 to 75 minutes
Student setup	Completed laptop setup and Lesson 1 to 2 workflow
Primary outcome	Students build a robot readiness checker using cin and if else
Collect	Skill check code plus one pass test and one fail test
Best pacing	Have students read conditions out loud as yes or no questions

Learning Targets

- I can ask the user for input with cin.
- I can write a condition that compares numbers.
- I can use if and else to choose between two messages.
- I can test both outcomes of a program.
- I can explain my decision logic in plain language.

Lesson Flow

Time	Teacher Move	Student Move
0 to 8	Review variables and cout.	Predict a short program output.
8 to 18	Explain cin and conditions.	Mark key words in workbook.
18 to 30	Demo battery check program.	Test 80, 50, and 49.
30 to 48	Guide comparison and else practice.	Fill in missing conditions.
48 to 65	Run skill check.	Build robot readiness checker.

65 to 75	Exit response.	Explain one ready case and one not ready case.
----------	----------------	--

Starter Program

```
#include <iostream>

using namespace std;

int main() {
    int batteryPercent = 0;

    cout << "Enter battery percent: ";
    cin >> batteryPercent;

    if (batteryPercent >= 50) {
        cout << "Battery check passed." << endl;
    } else {
        cout << "Charge the battery before driving." << endl;
    }

    return 0;
}
```

Common Problems

Problem	Fix
Using << with cin	cin uses >> to read input.
Missing parentheses after if	Write if (condition) before the brace.
Using = instead of ==	Use == when checking exact equality.
Only testing one input	Run one value that should pass and one that should fail.
Confusing else	Else means every case that did not pass the if condition.

Assessment

A ready student can read two numeric inputs, compare values, write a correct if else statement, test both outcomes, and explain the condition in everyday language.