The robot has mechanical systems and electrical hardware, but needs a program to tell it what to do.

The program collects inputs from the drivers and sensors, and uses them to decide what motor output should be.

Different programming “languages”:
- LabVIEW
- C++
- Java
Why C++?

- Powerful and fast language
- Used widely in industry

Steep learning curve, but after that development is fast. Programming tools are less complicated, smaller, faster than LabVIEW.
C++ Overview

- Invented in the 1980’s
- Built as an extension on top of C
- Object-oriented programming language
Variables

- Used to store information (data)
- Different types: e.g. int (integer), bool (true or false), float (decimal number)
- Can also create custom types (e.g. classes – discussed later)

```cpp
int myVar;
myVar = 5;

float myOtherVar = 15.03;
```
• Sections of text ignored by the robot
• Used to illustrate and explain things in plain English to people looking at the code

```c
int sensors; // Number of sensors.

/* This next block of code gets the value of the joystick Y-axis */
Joystick* stick = new Joystick(1);
float tilt = stick->GetY();
```
Conditionals

- Used to make decisions in programs
- Comparisons using variables and numbers are made

```cpp
if (myVar > 5) {
    // do something
}
else if (myVar < 2) {
    // do something else
}
else {
    // do another thing
}
```
Classes

- Representation of physical “things” in a program
- Used like custom variable types
- Examples: Joystick, Victor, Gyro, Relay

```cpp
Victor* theMotor;
Joystick* stick;
Gyro* gyro;
```
Functions

- Represent individual tasks
- Used to do things or get information

```cpp
int Add(int a, int b) {
    return a + b;
}
int sum = Add(723, 780);

void StartMotor() {
    motor.Set(0.5);
}
```
Files

- C++ has two different types of files
- Header (.h) files summarize the structure of classes
- Code (.cpp) files contain actual code
- By convention, each class has a .h file and a .cpp file
- Example: class Robot has Robot.h and Robot.cpp
WPILib

- Already-written code provided by FIRST to make robot programming easier
- Consists of classes that represent all common robot hardware
- Example: Compressor, DigitalInput, DriverStation, Solenoid, Accelerometer
The Windows program used to write robot programs and download them to the robot.
Contact Information

Philipp Schrader
Firmware Developer, Nuvation Research
philipp.schrader@spartonics.org

Tom Brown
Computer Science, Brock University
tom.brown@spartonics.org