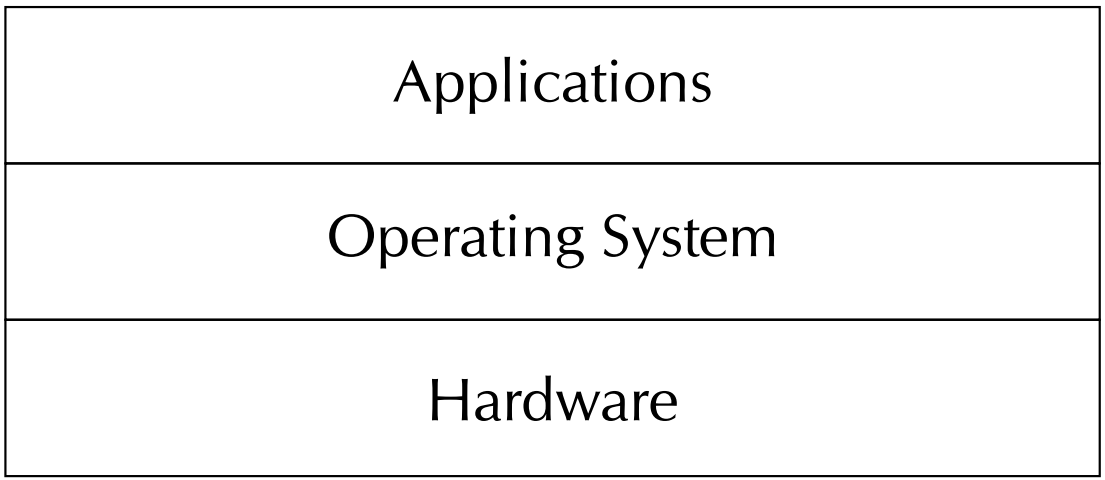
# What Is This Java Thing?

CSC207H Summer 2009

June 16th, 2009

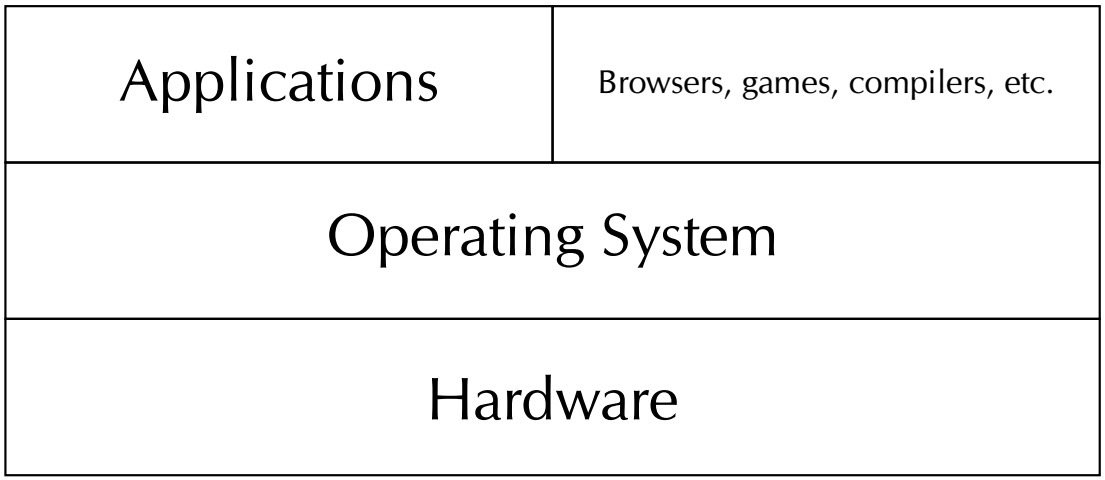
# Computer Architecture

- The operating system (OS) provides services to running software applications by managing hardware resources.
- The operating system abstracts away differences in hardware to provide a standard set of features for application programmers to use.
- Operating systems are typically written in C and assembly language.



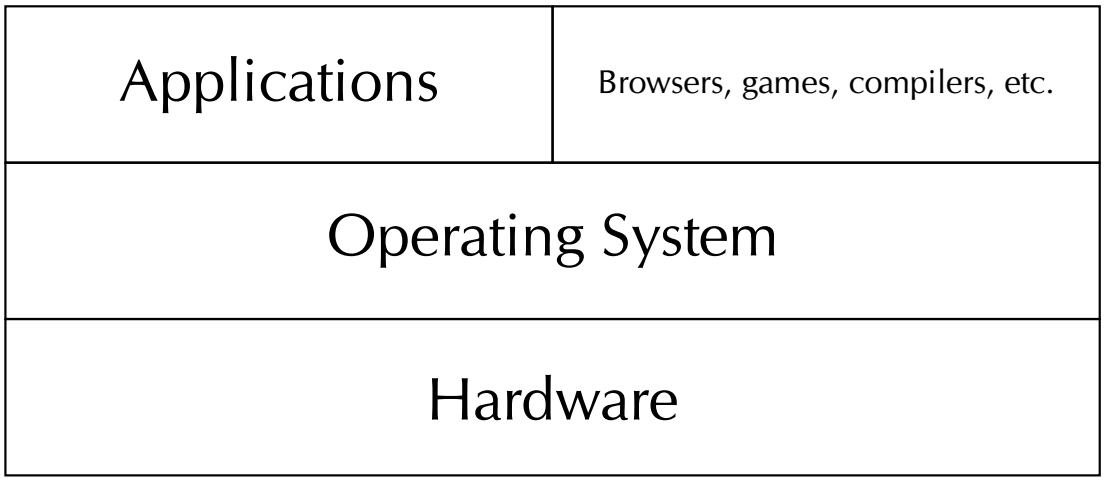
# Native Applications

- Programs written in languages like C, Pascal, and Fortran are compiled from their source code into applications whose instructions are directly run on the CPU and can take advantage of the services of the OS.
- Modern applications are typically under the control of an OS which can regulate an application's access to hardware resources, including terminating the application completely.



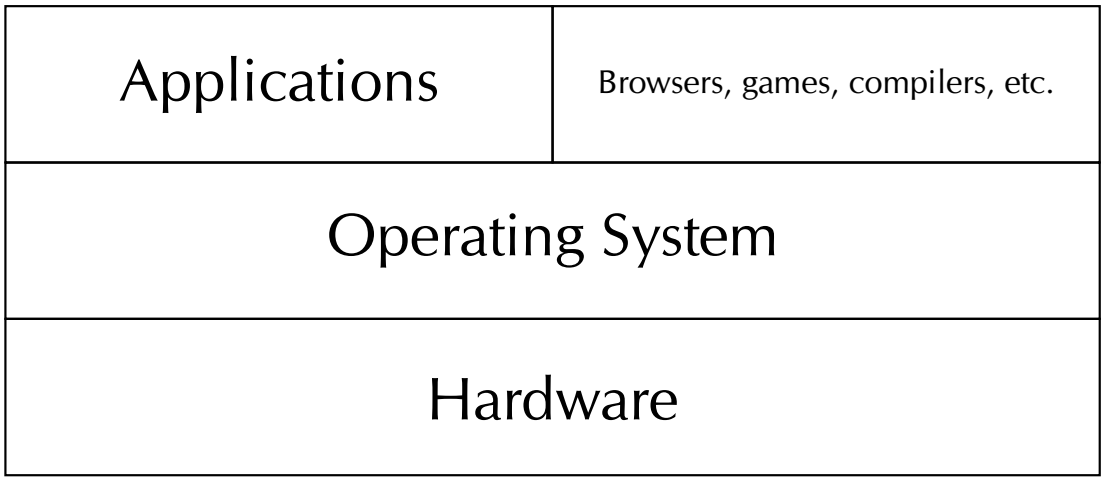
# Native Applications

- Compiling a program results in source code transformed into machine code; sometimes, assembly code is produced as an intermediary. (CSC258H, CSC369H, CSC488H)
- Applications are compiled for a single architecture-OS combination (*platform*) and generally will not run on other platforms.



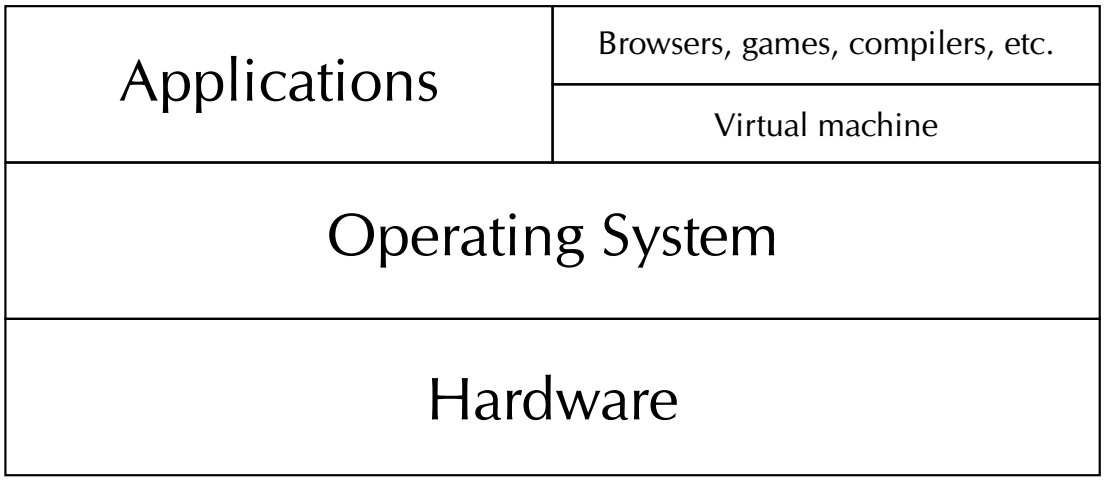
# Native Applications

- Code that does not require rewriting for different platforms is considered *portable*, even if it requires recompiling to run on them.
- The process of rewriting code to run on a different platform is known as *porting*. Isolating platform-dependent code helps improve portability since the changes that need to be made are restricted to fewer parts of the code.



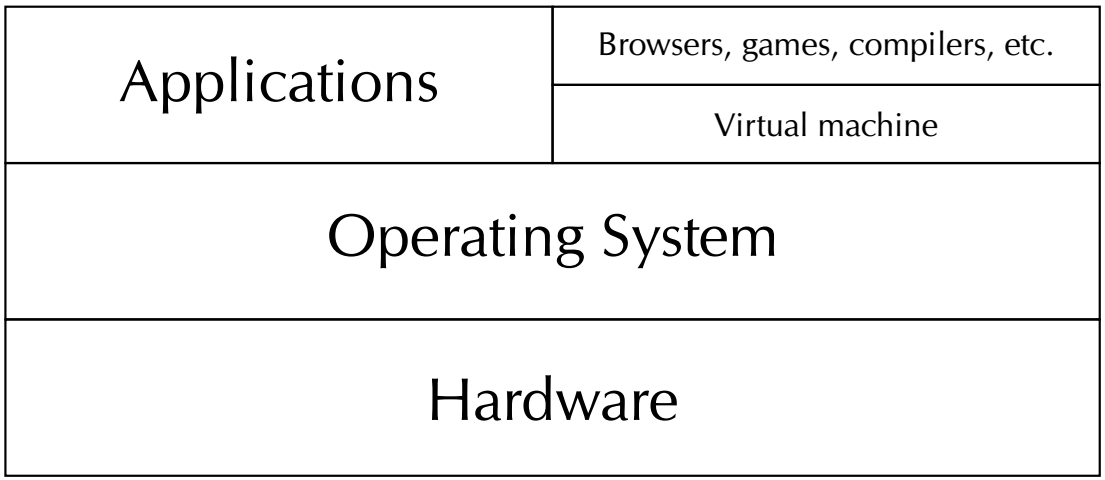
# Virtual Machines Architecture

- Virtual machines (VMs) present a virtual piece of hardware whose machine code is standardized to software. This VM often includes many services usually provided by an operating system.
- Since the VM itself is usually intended to run on several platforms, the VM itself is often written in a portable language such as C or C++.



# Virtual Machines Architecture

- Virtual machines greatly improve program portability; not only can programs have *source portability* (same code runs on multiple architectures), they can also have *binary compatability* (same compiled output runs on multiple architectures – but the same virtual architecture).
- Java, Python, Scheme, etc. each have their own VM. Java's is known as the Java Virtual Machine (JVM).



# Enter Java, Stage Left

- A project at Sun Microsystems lead to the development of a new object-oriented language, Oak, and associated VM.
- Oak was pitched for use for interactive set-to-boxes, but was turned down.
- Repositioned for providing interactive content on the world wide web, Oak was renamed Java.

# Enter Java, Stage Left

- Java gained traction and has since spread to a wide assortment of devices including mobile phones and PDAs.
- Java comes in several “flavours”:
  - Java 2 Standard Edition (J2SE)
  - Java 2 Enterprise Edition (J2EE) – provides an extended API useful for developing more robust and reliable software.
  - Java Mobile Edition (Java ME):
    - Connected Device Configuration (CDC) – a subset of J2ME found in some PDAs and other devices. CDC features different “profiles”; one of these, the Personal Basis Profile, is the foundation for Blu-ray Disc Java (BD-J).
    - Connected Limited Device Configuration (CLDC) – a subset of J2ME CLC found in less powerful and/or capable devices.

# Java Architecture

- The JVM and a set of libraries that implement the Java API are distributed together as the Java Runtime Environment (JRE).
- A JVM runtime is responsible for executing the machine code instructions (Java bytecode) of a Java program contained in a `.class` file. The machine code is for the JVM, not for the *host* hardware.
- Any program implementing the JVM specification can run a Java bytecode; in CSC207, we are using Sun Microsystems' HotSpot™ VM (`java`).
- Any program capable of outputting appropriate bytecode for the JVM can be used to write Java programs. The program most commonly used is Sun Microsystems' Java compiler (`javac`), though other alternatives exist.

# Java Architecture

- Sun Microsystems' Java compiler can compile programs written in the *Java programming language* into Java bytecode for the JVM.
- The Jython compiler can compile programs written in the Python programming language into Java bytecode for the JVM as opposed to Python's bytecode found in CPython.
- Since the JVM and Java API are consistent across different platforms, the same `.class` files can generally be reused on different computers.
- While each Java bytecode instruction is checked to ensure it cannot interact with the host platform in unintended ways, various tricks can be employed to allow Java programs to run at near-native speeds.