Runtime Systems: Introduction

Michael Jantz

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE

What is a Runtime System?

- Broadly defined: system that implements services necessary to execute programs according to their execution model
  - EM defines how the program will execute
- Distinction b/w compilers and runtimes:
  - Static choices in the compiler
  - Dynamic choices in the runtime system

Runtime Services

- C has a very lightweight runtime (crt0.o)
- Responsible for:
  - Initializing stack and frame pointers
  - Clearing BSS data
  - Invoking constructor initialization and ensuring destructors are called on exit
  - Calling main
  - Returning the appropriate error code

Source: "The C Runtime Initialization, crt0.o"

Runtime Services

- Java, etc. implement many more services
  - Portability
  - Garbage collection
  - Dynamic class loading
  - Security
  - Profiling application behavior
  - Exception handling
  - Debugging

Course Overview

- Virtualization / Virtual Machines
  - Structure, advantages, disadvantages
  - Types of VM's
  - Achieving portability (emulation)
  - Profiling and optimizations
  - JVM design and implementation
- Runtime Memory Management
  - Automatic vs explicit memory management
  - Garbage collection routines
  - Performance and implementation issues