Introduction to Service-Oriented Computing and Applications

Syllabus and Course Information

http://srlab.eas.asu.edu/highschool/soc/

Course Description

Introduces visual programming, service-oriented computing, and their applications in robotics programming, game programming, and Web programming.

Prerequisite: computer literacy and algebra

Text


Instructor and Teaching Assistants

Yinong Chen (PhD), yinong@asu.edu, (480) 965 2769
Calvin Cheng, calvin.cheng@asu.edu
Larry Xu, lxu24@asu.edu, (480) 965 9038

Background

Computer science is becoming the foundation of many science, technology, and engineering disciplines. Teaching basic concepts, principles, and programming in computer science in high schools is becoming as crucial as teaching mathematics and other science topics. Because of the different requirement, shifting a university’s computer science freshman course, such as programming in Java, C++, or C#, is not an adequate approach. Sponsored by U.S. Department of Education, the Computer Science and Engineering Department and College of Education at Arizona State University, has design a new computing curriculum for high schools. The course has been pilot-taught at Coronado High School in Spring 2007 and Fall 2008. As a part of our dissemination effort, we are teaching this course to teachers who are interested in computer science, robotics, game programming, and Web programming. No prior programming experience is required.

Objectives and Outcomes

The objectives and outcomes of the course are

1. To discover the excitement and creativity in the practice of computer science and engineering
   - Students will have a working knowledge in computer science and engineering disciplines.
   - Students will understand the concept of abstraction and the organization of a complex system.
   - Students will have knowledge of basic teamwork skills.
2. To understand the basic computer architecture and the paradigms of computing.
   - Students will understand the major computer components, their connection, and their roles in performing computation.
   - Students will understand major paradigms of computing: imperative, object-oriented, and service oriented.
   - Students will understand basic concepts in service-oriented computing.

3. To be able to write simple programs in an object-oriented programming language.
   - Students will understand the basic constructs of an object-oriented programming language.
   - Students will be able to apply programming constructs to write simple programs.
   - Students will be able to apply a programming development tool to compile, debug, and execute a program.

4. To understand service-oriented computing and service-oriented software development.
   - Students will be able to lookup a service directory to find existing services.
   - Students will understand how to compose an application using existing services.

5. To apply the service-oriented computing in problem solving in different application domains.
   - Students will be able to program a robot to move and to take actions through remote control.
   - Students will be able to program a robot to move and to take actions by pre-programmed artificial intelligence.
   - Students will be able to create a 3-D movie or game using existing components.

Major Topics Covered in the Course (Tentative)

1. Basic computer science concepts
   1) The development of computer systems
   2) Basic building blocks and computer organization
   3) Basic computation process and computing paradigms
   4) Software development process

2. Introduction to visual programming language C# and .Net
   1) Basic data and data types
   2) Language constructs
   3) Classes and methods
   4) Graphic user interface design

3. Introduction to service-oriented and component-based computing
   1) Basic concepts in service-oriented architecture and computing
   2) Game programming using components and services
   3) Web service programming
   4) Application composition
   5) Service directories

4. Robotics programming
   1) Service-oriented Robotics Studio
   2) Visual programming language for robotics programming
   3) Robotics programming
   4) Robotics programming competition