2018 HAWAII UNIVERSITY INTERNATIONAL CONFERENCES
STEAM - SCIENCE, TECHNOLOGY & ENGINEERING, ARTS, MATHEMATICS & EDUCATION
JUNE 6 - 8, 2018 PRINCE WAIKIKI, HONOLULU, HAWAII

IMPLEMENTATION OF SOFTWARE CURRICULUM FOR GENERAL EDUCATION

NAH, JEONG EUN
UNIVERSITY COLLEGE
YONSEI UNIVERSITY
SEOUL, SOUTH KOREA

Prof. Jeong Eun Nah
University College
Yonsei University
Seoul, South Korea

Implementation of Software Curriculum for General Education

Synopsis:

As the intelligent information technology society has arrived, the need for software education has become more and more important. In Korea, software education has been introduced in the middle and high school curriculum, and software education has been strengthened in universities accordingly. As the need for software education for not only Computer Science majors but also non-CS majors became necessary, universities began to review and reorganize software education courses. In college, software education other than computer science major is taken care of in the Liberal Arts. It is difficult to organize software curriculum into various courses in stages. It is necessary to cover the overview of computer science and the programming language as basic contents, and to present a variety of elective courses to learn additional contents according to students' major and interest areas.

I. Introduction

The foundation of intelligent information technology that will lead the fourth industrial revolution is based on software skills, and competitiveness comes from people who can finally understand and apply to enabling capability of software skills. In addition to critical thinking, creative problem solving, communication, and collaborative skills, the skills required for 21st century should include understanding of information technologies and basic knowledge of software skills.

It is required because computer or information technology is related to all areas. It requires both universal IT knowledge to enhance computer literacy and specific IT skills needed to solve problems. The ability of computer literacy, including the concept of universality and expertise, is the basis for computational thinking. Computational thinking is not just for computer scientists, it is a fundamental skill for everyone. In the end, there is a need for students who are not familiar with computers to be provided with a curriculum that fosters the ability to easily access the computer and make strengthen people's ability to use computers.

Not limited to Computer Science major students, computer education is necessary for those who are in Liberal Arts curriculum. Designing curriculum for Liberal Arts major students should be different from Computer Science major students. Because it is not possible to take as many courses as major, it should be designed to acquire some degree of knowledge and understanding of information technology with several related courses to learn in Liberal Arts course.

II. Background: Reorganization of Computer Education

Software education in Korea is being emphasized. In the case of elementary school, ICT contents organized in some units are organized as software basic education. Starting with the 2019 school year, more than 17 hours of software training a year is required. Since 2015, more than 300 elementary schools are designated as the leading educational institutions in education every year. The middle school has changed from the elective courses since 2017, to learn problem solving, programming development, and simple algorithms based on computational thinking more than 34 hours a year. Since 2018, high school has also been organized as an elective course, expanding opportunities for software classes.

In the university, it is a trend that software education is needed, and curriculum is newly organized or new software course is opened. The software curriculum for non-Computer Science majors is mainly opened in the Liberal Arts. Students may study computer science as a minor or double major, but most of them can take one or two computer-related courses in Liberal Arts. It is necessary to organize course that teaches computational thinking rather than a curriculum that learns a programming language. Students have to learn to think about computing in order to solve problems in their majors. Various courses related to computing should be opened to foster computational thinking and to improve the computer application ability for non-computer science major students.

III. Methods: Guideline for Curriculum Development

As the curriculum is developed, the most important thing is to establish educational goals. Through this process, contents of the curriculum will be developed according to the setting as to what is expected as educational effects. Every curriculum should introduce the theoretical knowledge of computer science related to the contents, and should be combined with hands-on experience. Understanding of pre-given problem and analyze it, and comparing the alternative solutions and finding out the solution, and implementing the result is the process of hands-on practical training. This process is a hands-on training session for integrating computational thinking into the problem-solving process. The ultimate goal is not to gain computer programming skills. It is a process to understand the characteristics of a computer, what computers can do, and how computers can be used. Therefore, we proposed a basic curriculum that includes an understanding of computers and computer operation. Through this course, what we expect to do is to use computers as tools for solving problems in daily lives and to make computing thinking as an approach to problem solving.

Students taking this course are not Computer Science students but students who have their own majors. Major courses can be structured from level 1 to level 4, but Liberal Arts courses are difficult to compose courses in stages. Therefore, basically, it is divided into two levels, the basic level and the application level. Students can skip basic courses according to their level of understanding.

Students will be able to understand the introduction to computer science, learn programming language, and programming skills. Students will be able to their majors and areas of interest. Process for data processing including big data processing, multimedia processing in visualized process related to image or media, smart device application in wireless environment or cell phone environment. We want to broaden the range of application areas more and more. The curriculum should be composed of subjects of various application fields that can be selected according to students' interests. Because students have different levels of understanding of computers according to their various learning backgrounds, it is possible for them to take courses among basic subjects, applied subjects at their own discretion. In the future, it will be necessary to continuously develop application courses to support students' needs and computer skills required in their major.

IV. Results and Further Studies

The university plans to strengthen software education for non-CS major students. Computational thinking education was conducted as a pilot course and then opened as a liberal arts course. About 500 students take software basic courses each semester. In the case of basic subjects, we will teach introduction of computer science and software programming skills. To develop the computational thinking and the software programming in the Python language, we provide several exercises include the process of solving the problem to confirm the results in a given time. Analyzing and designing a given problem, then programming it in Python is the main focus of the exercise.

Software programming courses were opened as a course for students who want to strengthen their software skills. This course is aimed at students who want to learn more programming skills unlike basic subjects. This course is designed to help students who want to take a minor or double major in the future with the same level of programming as the computer science major.

We plan to open additional courses in near future. Understanding data processing will help students understand the system as it allows them to understand system structures and workflow processes. Physical computing using software to move hardware entities will also be the next step in understanding object computing. In order to make it easier to understand and utilize the computer as a practical approach in real life, we plan to establish the subjects with various contents of wide range of application.