

웹 기반의 교육용 CPU 스케줄링 시뮬레이터의 설계 및 구현

고정국*

Design and Implementation of a Web-based educational CPU Scheduling Simulator

Jeong-Gook Koh*

Department of Computer Engineering, Tongmyong University, Busan 608-711, Korea

요 약

운영체제는 추상적 개념과 기법들을 다루고 있지만, 대부분 이론 수업 위주로 진행되고 있다. 이론 수업은 강의 내용에 대한 이해와 집중력 저하를 유발하므로 학습 내용의 이해를 돕고 흥미를 유발하는 다양한 방법들이 시도되었다. 본 논문에서는 CPU 스케줄링 알고리즘들의 이해를 돕기 위해 학습자가 언제, 어디서나 웹 브라우저나 스마트폰을 사용하여 다양한 상황을 설정하고 알고리즘의 스케줄링 과정을 모의 실험해 볼 수 있는 웹 기반의 교육용 CPU 스케줄링 시뮬레이터를 설계하고 구현하였다. 구현된 시뮬레이터를 수업에 활용한 후 2014학년도 수강생들을 대상으로 구현된 시뮬레이터의 사용 편의성과 유용성에 대한 만족도 설문조사를 통해 시뮬레이터가 학습 내용에 대한 이해도 증진과 문제해결 능력을 배양하는 교육용 도구로 유용함을 확인할 수 있었다.

ABSTRACT

Operating Systems is a discipline which handles abstract concepts and techniques. However, most of OS courses have been textbook-oriented theoretical classes. Theoretical classes lead to the decline in the understanding of a lecture and hurt their concentration. Many instructors have tried various ways to help students understand lectures and arouse interests. This paper describes the design and implementation of a web-based educational CPU scheduling simulator which enables learners to set up various situations and simulate scheduling processes using a web browser or a smart-phone. After utilizing the implemented simulator for the class, the survey result for 2014's students shows that it is useful as an educational tool which enhances the understanding of a lecture and develops their problem solving skills.

키워드 : 운영체제, CPU 스케줄링 알고리즘, 교육용 도구, 시뮬레이터

Key word : Operating System, CPU Scheduling Algorithm, Educational tool, Simulator

Received 20 May 2015, Revised 09 June 2015, Accepted 23 June 2015

* Corresponding Author Jeong-Gook Koh(E-mail:jgkoh@tu.ac.kr, Tel:+82-51-629-1171)

Department of Computer Engineering, Tongmyong University, Busan 608-711, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2015.19.7.1653>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

컴퓨터 분야의 교육에서 실습은 이론 수업에서 학습한 내용의 이해를 돕고 활용 능력을 배양하기 위해 필요하다. 수업 현장에서도 이론적인 내용만 설명하는 것보다 실습이 추가되면 학습자의 이해도가 훨씬 높아진다[1]. 이론 수업은 시간이 경과할수록 집중도가 저하되고 지루함이 증가하지만, 실습을 병행하면 집중도를 유지할 수 있다[2]. IEEE Computer Society와 ACM의 합동 태스크 포스트는 “컴퓨터 분야의 교육과정은 이론과 실습을 병행하도록 편성해야 한다.”라고 권고한다[3].

한편 운영체제는 컴퓨터 시스템의 자원들을 효율적으로 관리하고 편리한 사용 환경을 제공하는 시스템 소프트웨어이다. 운영체제 수업에서는 구성 요소의 동작 원리와 개념, 기법 등을 다루지만 운영체제의 동작을 직접 확인하기 어려워 대부분 이론 위주로 진행한다. 새로운 개념이나 내용을 학습할 때 조작가능한 도구의 활용이 바람직하네, 이론 강의는 학습 내용에 대한 이해도와 집중력 저하를 초래하여 재미없다고 인식하는 경향이 나타난다[4]. 따라서 멀티미디어 자료나 교육용 도구를 활용하여 흥미를 유발하고 강의 내용에 대한 이해도 증진을 위한 다양한 시도가 이루어지고 있다[5].

본 논문에서는 학부 운영체제 수업에서 다루는 CPU 스케줄링 알고리즘들에 대한 이해를 돕기 위해 프로세스 스케줄링 과정을 시각적으로 보여주는 웹 기반의 CPU 스케줄링 시뮬레이터를 설계하고 구현하였다.

본 논문의 구조는 다음과 같다. 2장에서는 운영체제 교과목의 강의방법 개선 시도와 교육용 시뮬레이터의 필요성을 설명한다. 3장에서는 CPU 스케줄링 시뮬레이터의 기능 설계에 대해 기술한다. 4장에서는 시뮬레이터의 구현 내역과 기존 CPU 스케줄링 시뮬레이터들과 본 논문에서 구현한 시뮬레이터의 특징을 비교한다. 마지막으로 5장에서는 본 논문의 결론 및 향후 연구방향을 제시한다.

II. 관련 연구

2.1. 강의방법 개선 시도

운영체제 수업에서 복잡하고 추상적인 개념과 처리 기법 등을 학습하지만 대부분 이론 위주로 진행되면서

학습자들의 관심과 흥미가 저하되는 문제를 해결하기 위해 다양한 강의개선 방법들이 시도되었다. 대표적인 사례로는 개인이나 팀 단위로 운영체제의 일부 기능을 구현하는 단기 프로젝트 수행, 교육용 운영체제나 실제 운영체제에 특정 기능을 구현하는 소스 코드 수정, 그리고 추상적인 개념에 대한 이해를 돕는 교육용 시뮬레이터 활용이 있다[6].

교육용 시뮬레이터는 활용 범위에 따라 다양한 체험이 가능한 범용 시뮬레이터와 특화된 기능을 제공하는 단일용도 시뮬레이터로 구분할 수 있다. 범용 시뮬레이터는 활용도는 높지만 개발 기간이 길고 개발 비용도 많이 소요되는 단점이 있다. 그러나 단일용도 시뮬레이터는 범용 시뮬레이터에 비해 활용 범위는 제한되지만, 개발 기간이 짧고 비용도 적게 소요되는 장점이 있다. 대표적인 범용 시뮬레이터로는 OSP[3], SOsim[6], RCOS.java[7] 등이 있으며, 단일용도 시뮬레이터에는 BACI[8], MOSS[9], Alg_OS[10], OSM[11], Robbins[12], CPUSim[13] 등이 있다.

한편, 다양한 웹 기반 e-Learning 시스템과 교육용 도구들이 학습자들에게 시간적, 공간적 제약이 없는 학습 환경 제공을 위해 개발되었다. 컴퓨터 분야에서는 자료 구조나 알고리즘, 운영체제 등의 효과적인 학습을 위해 활용되고 있으며, 대표적 사례로는 Alg_OS[10], PageRepSim[15], WebGeneOS[16] 등이 있다.

2.2. 교육용 시뮬레이터의 필요성

이론 수업에서는 교수자가 학습자들에게 지식을 일방적으로 전달하는 방식이 많이 사용된다. 이 방식은 학습자들의 독립적인 사고와 상호작용을 저해하여 흥미와 강의내용 이해도를 저하시키는 문제가 있다[5]. 특히 운영체제처럼 추상적인 개념을 많이 다루는 교과목에서는 상황이 더욱 악화될 수 있다.

문제의 해결책은 수업의 중심을 교수자에서 학습자로 옮기는 것이다[14]. 예를 들어, 교수자는 학습자들에게 CPU 스케줄링 알고리즘들을 설명한 후 교육용 시뮬레이터를 활용하여 처리 과정을 시연한다. 학습자들은 시뮬레이터를 이용하여 다양한 상황을 설정하고 알고리즘의 동작 방식과 실행 결과를 확인하고 이들을 비교함으로써 알고리즘에 대한 올바른 이해가 가능해진다. 따라서 이러한 용도로 사용할 수 있는 교육용 시뮬레이터를 개발할 필요가 있다.

III. CPU 스케줄링 시뮬레이터의 설계

3.1. CPU 스케줄링 시뮬레이터의 기능 설계

이전 연구[13]에서 CPU 스케줄링 알고리즘 교육을 위해 교육용 도구(CPUSim)를 구현하였다. CPUSim은 자바로 구현한 설치형 프로그램이며, 프로그램이 설치된 PC에서만 사용할 수 있어서 활용시간과 장소에 제약이 있었다. 본 논문에서는 이러한 단점을 보완하여 언제, 어디서나 CPU 스케줄링 알고리즘의 동작을 모의 실험할 수 있도록 하면서 학습내용 이해를 돕는 기능을 추가적으로 제공하고는 웹 기반의 CPU 스케줄링 시뮬레이터(CPUSim2)를 제안한다. 그리고 스마트폰의 보급 추세와 편리함을 감안하여 스마트폰에서도 구동되도록 한다. 본 논문에서 제안하는 시뮬레이터는 그림 1과 같이 클라이언트, 웹 서버와 CPU 스케줄링 프로그램으로 구성된다. 시뮬레이터의 역할을 담당하는 CPU 스케줄링 프로그램은 환경 설정부와 결과 출력부로 구성된다.

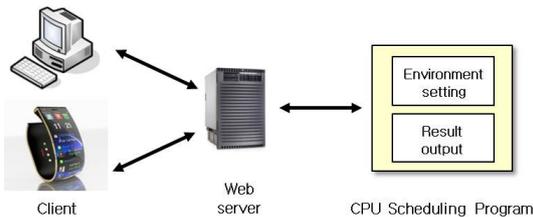


그림 1. CPU 스케줄링 시뮬레이터의 시스템 구조
 Fig. 1 The system architecture of CPU scheduling simulator

학습자는 PC의 웹 브라우저나 스마트폰을 활용하여 웹 서버에 접속한 후 시뮬레이터를 사용한다. 웹 서버는 클라이언트와 CPU 스케줄링 프로그램사이에서 중계자 역할을 담당한다. 학습자가 시뮬레이션 매개변수들을 설정하고 알고리즘을 선택하면 CPU 스케줄링 프로그램이 해당 알고리즘의 스케줄링 결과와 성능 수치를 보여준다.

3.1.1. 환경 설정부

환경 설정부는 시뮬레이션 매개변수 설정과 알고리즘 선택 기능을 제공한다. 학습자는 프로세스 개수(2~10개)와 프로세스 관련 정보들(도착시간, 서비스 시간, 우선순위)의 생성 방법을 선택한다. 즉 프로세스 개

수를 콤보 박스에서 설정한 후 정보생성 방법을 선택한다. ‘수동 입력’을 선택하면 학습자가 직접 정보를 입력하며, ‘자동 입력’을 선택하면 설정된 개수만큼 정보들이 생성된다. 매개변수 설정이 완료되면 6가지 종류(FCFS, HRN, SJF, SRT, 우선순위, 라운드 로빈)의 스케줄링 알고리즘을 선택할 수 있다. 학습자가 라운드 로빈 알고리즘을 선택할 때에는 프로세스에게 할당될 CPU 시간(Time quantum) 설정 기능이 활성화된다. 그리고 시뮬레이터는 시간 단위로 ms를 사용한다.

한편, 시뮬레이터는 2가지 종류의 실행 방식을 지원한다. 학습자가 ‘일괄 실행’을 선택하면 스케줄링 결과가 간트 차트에 표시되며, 스케줄링 결과를 기법별로 비교할 수 있도록 간트 차트가 스택 차트에 표시된다. 그리고 ‘단계별 실행’을 선택하면 해당 버튼을 클릭할 때마다 스케줄링 결과가 간트 차트에 1ms마다 출력되며, 프로세스의 잔여 실행시간이 도넛 차트로 표시된다. 그 밖에 프로세스가 실행됨에 따라 준비 큐의 변화 과정이 그래픽으로 표시되며, 학습자가 기설정된 매개변수들을 변경할 수 있도록 ‘재설정’ 기능도 제공한다.

3.1.2. 결과 출력부

결과 출력부는 설정된 프로세스 정보들에 근거하여 스케줄링 결과를 간트 차트로 출력하고 해당 알고리즘의 성능 수치(평균 대기시간과 평균 반환시간)를 보여준다. 그리고 동일한 상황에서 실행된 알고리즘들의 성능 수치가 화면에 누적되어 표시되므로 알고리즘간 성능 비교도 가능하다.

IV. CPU 스케줄링 시뮬레이터의 구현

4.1. 구현 환경

본 논문에서는 CPU 스케줄링 시뮬레이터를 구현하기 위해 개발 언어로 C#과 ASP.NET을 선정하고, 개발 도구로 Visual Studio 2008을 사용하였다. IIS 7을 사용하여 웹 서버를 구축하였고, 구현된 시뮬레이터가 웹 브라우저뿐만 아니라 스마트 기기들에서도 정확하게 동작하는지 확인하기 위해 대표적인 스마트 기기인 스마트폰을 대상으로 3가지 종류의 단말기(갤럭시 S3, 갤럭시 노트 3, 옵티머스 G Pro)를 선정하고 기능 시험을 실시하였다.

4.2. 구현 내역

4.2.1. 사용자 인터페이스

학습자가 웹 브라우저나 스마트폰을 이용하여 웹 서버에 접속하면 그림 2의 시작 화면이 나타난다. 시작 화면의 용도는 다음과 같다. 화면 왼쪽은 환경 설정부로서, 상단에서는 시뮬레이션 매개변수 설정과 알고리즘 선택, 스케줄러의 동작을 제어하며 하단에는 설정된 프로세스 관련 정보들이 표시된다. 화면 오른쪽은 결과 출력부로서, 상단에는 스케줄링 결과와 스케줄링 기법별로 프로세스들의 실행시간이 스택 차트로 표시된다. 하단에는 스케줄링 기법별 성능 수치가 표시된다.



그림 2. 시뮬레이터의 시작 화면
Fig. 2 The starting window of a simulator

ID	우선순위	도착시간	서비스시간
p1	5	2 ms	8 ms
p2	7	7 ms	2 ms
p3	7	1 ms	2 ms
p4	7	0 ms	1 ms
p5	8	6 ms	2 ms
p6	3	0 ms	1 ms
p7	4	9 ms	3 ms
p8	5	8 ms	3 ms
p9	2	7 ms	6 ms
p10	2	1 ms	7 ms

그림 3. 프로세스 관련정보 설정 화면
Fig. 3 The window for process related information setting

4.2.2. 환경 설정부

환경 설정부에서 프로세스 관련정보 설정 화면은 그림 3과 같다. 설정된 정보들은 프로세스 ID를 기준으로 정렬된 후 우선순위, 도착시간과 서비스 시간이 표시된다. 한편 학습자가 ‘우선순위’ 기법을 선택한 경우에는 프로세스의 우선순위 정보가 표시된다.

4.2.3. 결과 출력부

프로세스 관련 정보를 생성한 후 ‘실행’ 버튼을 클릭하면 스케줄링 결과가 그림 4와 같이 간트 차트로 표시되며, 스케줄링 기법별 프로세스들의 작업 처리시간은 그림 5와 같이 스택 차트로 표시된다.

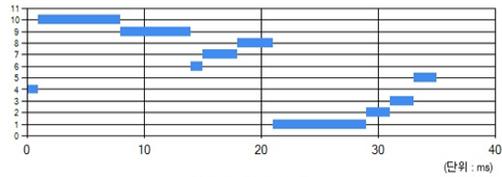


그림 4. 프로세스 스케줄링 결과
Fig. 4 The result of process scheduling

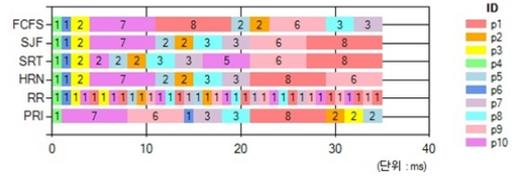


그림 5. 스케줄링 기법별 프로세스의 작업 처리시간
Fig. 5 The job processing times of processes based on scheduling algorithms

‘단계별 실행’ 버튼을 클릭하면 스케줄링 결과가 단위 시간마다 출력되며, 준비 큐의 변화 과정, 실행 중인 프로세스의 총 실행시간 및 잔여 실행시간은 그림 6과 같이 표시된다. 결과 출력부의 하단에는 스케줄링 기법별로 그림 7과 같이 2가지 종류의 성능 수치가 출력된다. 학습자가 성능 수치 산출 내역 확인과 성능 비교를 할 수 있도록 ‘자세히’ 버튼과 ‘차트’ 버튼을 제공한다. ‘자세히’ 버튼을 클릭하면 성능 수치 산출 과정이 표시되며, ‘차트’ 버튼을 클릭하면 성능 수치 표가 그림 8의 성능 비교 차트로 변경된다.

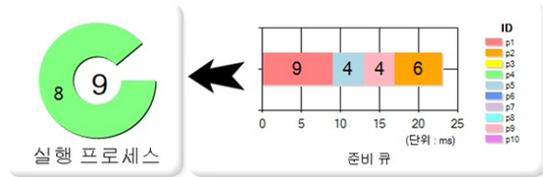


그림 6. 준비 큐의 변화, 프로세스의 실행시간 정보
Fig. 6 The change of ready queue and execution information of a process

스케줄링 기법	평균 대기시간	평균 반환시간
FCFS	10.1	13.6
SJF	7.1	10.6
SRT	5.1	9.6
HRN	7.3	10.8
Round Robin	2.2	15.4
Priority	12.9	16.4

그림 7. 스케줄링 기법별 성능 수치
Fig. 7 Performance values of scheduling algorithms

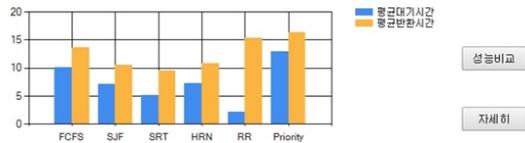


그림 8. 성능 비교 차트
Fig. 8 A chart for Performance comparison

4.2.4. 스마트 기기 지원

스마트폰의 보급 속도를 고려할 때 스마트 기기에서도 시뮬레이터를 사용할 수 있다면 활용도가 더욱 높아질 것이다. 본 논문에서는 스마트 기기에서 시뮬레이터를 활용할 수 있도록 웹 서버 접속에 사용된 기기의 해상도를 인식한 후 해당 기기에 적합한 시뮬레이터를 구동하도록 구현하였다. 학습자가 스마트폰을 이용하여 웹 서버에 접속하면 그림 9의 화면이 나타난다. 스마트 기기는 PC용 웹 브라우저에 비해 화면 크기가 작기 때문에 모바일 버전에서는 프로세스 개수를 최대 5개로 제한하는 점 외에는 웹 버전과 동일한 기능을 제공한다.

4.3. 기존 연구 사례와의 비교

기존의 CPU 스케줄링 시뮬레이터들과 본 논문에서 구현한 CPUSim2의 특징 비교 결과는 표 1과 같다.

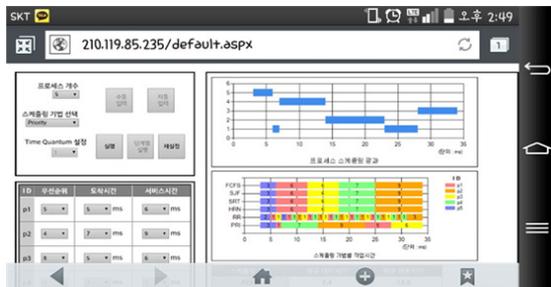


그림 9. 스마트 기기용 시뮬레이터의 화면
Fig. 9 A screen of a simulator for smart devices

표 1. 기존 시뮬레이터들과 CPUSim2의 비교
Table. 1 Comparison between the previous simulators and CPUSim2

Sort Item	SOsim	Robbins	OSM	CPUSim2
Running form	Stand-alone	Standalone+Web browser	DB interlock	Standalone+Web browser/Smart device
Range of use	General-purpose	Single-purpose	Single-purpose	Single-purpose
Number of Algorithms	Two	Six	Four	Six
Setting up a situation	Manual	Manual	Manual	Automatic/Manual
Information offering	Process-oriented	Result-oriented	Process-oriented	Process-oriented
Screen composition	Single window	Multiple window	Multiple window	Single window
Difficulty of using	Low	High	Medium	Low
Performance comparison	×	×	○	○
Execution environment Construction	Easy	Easy	Difficulty	Easy
Provision of learning process	×	×	○	×

SOsim[6]은 다중 프로그래밍, 프로세스 관리, 스케줄링, 메모리 관리 기능을 제공하는 독립 실행형 범용 시뮬레이터이다. 2가지 유형의 알고리즘(비선점형, 선점형)을 지원하며, 로그 저장 기능을 제공한다. 그러나 프로세스 스케줄링에 한정하면 지원 알고리즘 수와 제공 기능이 미약하다.

Robbins[12]는 웹 브라우저와 독립 실행이 가능한 단일용도 시뮬레이터로서 6가지 알고리즘(FCFS, SJF, 선점형 SJF, SJFA, RR, 우선순위)을 지원한다. 메인 화면에서 메뉴를 선택하면 실행 결과가 별도 화면으로 제공된다. 매개변수 설정과 스케줄링 결과 저장 기능을 제공하지만, 화면 구성이 복잡하고 사용이 어려운 편이다. 스케줄링 정보는 결과 위주로 제공한다.

OSM[11]은 DB를 활용하는 단일용도 시뮬레이터로서, 4가지 알고리즘(FCFS, SJF, 우선순위, RR)을 지원한다. CPU 스케줄러에 대한 이론적 설명을 포함한 3단계 학습 과정을 제공하며, 알고리즘의 처리 과정을 시각적으로 보여준다. 처리 결과를 DB에 저장하며 결과

값을 이용하여 알고리즘간 성능을 비교한다. DB를 활용하므로 실행환경 구축과 시뮬레이터의 독립 실행은 어렵다. CPUSim2는 독립실행형 프로그램인 CPUSim의 개량형으로 PC용 웹 브라우저와 스마트폰에서도 동작한다. CPUSim의 제공 기능 외에도 실행방식 선택 기능, 스케줄링 결과 비교용 스택 차트와 성능 비교 차트를 제공한다. 또한 프로세스 실행에 따른 준비 큐의 변화 과정과 프로세스의 잔여 실행시간을 그래픽으로 보여준다. 한편, CPUSim은 매개변수 설정 방법을 알려주는 도움말 기능을 제공하였으나 CPUSim2에서는 입력란을 콤보 박스로 대체하여 입력 오류도 줄이고 사용의 편리성도 제공한다.

4.4. 만족도 조사

CPUSim2를 2014학년도 수강생 22명에게 제공한 후 설문조사를 통해 사용 편의성과 유용성에 대한 만족도 조사를 실시하였다. 편의성과 유용성에 대한 3가지 항목을 분석 대상으로 선정하고 항목당 5점 척도로 측정하였다. 조사 결과는 표 2와 같으며, ‘-’ 표시는 해당 항목에 대한 응답이 없음을 표시한다.

조사 결과를 살펴보면 우선 사용 편의성 평가에서는 ‘정보 습득의 편의성’을 높게 평가했다. 이것은 학습자가 다양한 상황 설정과 스케줄링 기법의 동작 방식, 스케줄링 결과, 성능 수치 등을 손쉽게 산출할 수 있기 때문이라고 판단된다. 다음으로 제공 기능들에 대한 유용성 평가에서는 ‘성능 비교 기능의 유용성’이 높게 나타났다. 이것은 스케줄링 기법들에 대한 성능 수치와 산출 내역이 표시되고, 성능 비교 표/차트를 활용한 기법별 성능 비교가 가능했기 때문이라고 판단된다.

표 2. 만족도 조사 결과

Table. 2 The satisfaction survey result

Evaluation item		Too much	Enough	Some	Average
convenience	Layout	6	14	2	4.18
	Functional operation	14	6	2	4.55
	Information acquisition	20	2	-	4.91
usability	Chart	16	2	4	4.55
	Performance Comparison	18	4	-	4.82
	Simulator	16	6	-	4.73

V. 결 론

본 논문에서는 운영체제 교과목의 주요 학습 주제인 CPU 스케줄링 기법들에 대한 학습자의 이해도 증진을 위해 언제, 어디서나 웹 브라우저나 스마트 기기를 사용하여 다양한 상황을 설정하고 알고리즘의 동작을 모의 실험해 볼 수 있는 웹 기반의 CPU 스케줄링 시뮬레이터를 설계하고 구현하였다. 구현된 시뮬레이터를 운영체제 수업에 적용한 후 수강생을 대상으로 설문조사를 통해 시뮬레이터의 사용 편의성과 유용성에 대한 만족도 조사를 실시하였다. 조사 결과 구현된 시뮬레이터가 사용하기 편리하며, 학습 내용에 대한 이해도 증진과 문제 해결능력 배양에 매우 유용함을 확인하였다.

향후 연구로는 학습자가 CPU 스케줄링 시뮬레이터를 사용할 때 스케줄링 알고리즘의 스케줄링 과정과 결과, 성능 수치 등을 입력하면 정오를 확인할 수 있는 자기주도 학습 및 평가 기능을 추가할 것이다. 그리고 평가 결과를 저장하고 저장된 결과들을 분석하여 학습자의 내용 이해도와 학업성취도 변화 이력을 보여줌으로써 활용 효과를 극대화시킬 예정이다.

ACKNOWLEDGMENTS

This Research was supported by the Tongmyong University Research Grants 2013.

REFERENCES

[1] Jae-Chun Kim, Jae-Yool Boo, Kyung-Hee So, Sun-Hee Chae, *Curriculum and Educational Evaluation*, 3rd ed. KyoyookBook, 2005.

[2] Gyu-Hyuk Lim, Ung Lim, *Educational Psychology for School Learning Effect*, 2nd ed. HAKJISA corp., 2010.

[3] M. Kifer and S. Smolka, "OSP: An Environment for Operating System Projects," *Operating Systems Review*, vol. 24, no. 4, pp. 98-100, Apr. 1992.

[4] Jung-Eun An, Young-Bong Kim, "A Web-Based Courseware for 3D Bicycle Museum on the basis of Constructivism Learning theory," *Proc. of the 2007 Spring Conference of the Korea Contents Association*, vol. 5, no. 1, pp. 19-23,

- 2007.
- [5] Sung-Ik Park, Cheol-Il Lim, Jae-Kyung Lee, Jeong-Im Choi, *Educational technology understanding of the teaching methods*, KyoyookBook, 2007.
- [6] L. Maia and A. Pacheco, "A simulator supporting lectures on operating systems," *Proc. of 33rd Annual Frontiers in Education*, vol. 2(F2C), pp. 13-17, 2003.
- [7] D. Jones and A. Newman, "RCOS.java: A simulated operating system with animations," *Proc. of Computer-Based Learning in Science Conference*, 2001.
- [8] B. Bynum and T.Camp, Alfonso: A Mutual Exclusion Toolkit, Available: http://inside.mines.edu/~tcamp/baci/baci_index.html.
- [9] R. Ontko, MOSS Memory Management Simulator User Guide, Available: http://www.ontko.com/moss/memory/user_guide.html.
- [10] A. Garpis, "Design and Development of a Web-based Interactive Software Tool for Teaching Operating Systems," *Journal of Information Technology Education*, vol. 10, no. 10, pp. 1-17, Oct. 2011.
- [11] Seong-Kyun Jeong, Samuel Sangkon Lee, "Design and Implementation of Simulation Program for CPU Scheduling in Operating Systems," *Journal of Korea Multimedia Society*, vol. 14, no. 6, pp. 449-461, Jun. 2011.
- [12] S. Robbins and K.A. Robbins, "Empirical Exploration in Undergraduate operating systems," *Proc. of the 30th SIGCSE technical symposium on Computer Science Education*, vol. 30, no. 1, pp. 311-315, 1999.
- [13] Jeong-Gook Koh, "CPUSim: A Simulator supporting the education of CPU Scheduling Algorithm," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 16, no. 4, pp. 835-842, Apr. 2012.
- [14] M. Ben-Ari, "Constructivism in Computer Science Education," *Proc. of the 29th SIGCSE technical symposium on Computer Science Education*, vol. 29, no. 1, pp. 257-261, 1998.
- [15] Jeong-Gook Koh, "Design and Implementation of a Web-based Simulator for Educating Page Replacement Algorithms," *Journal of Korea Multimedia Society*, vol. 15, no. 4, pp. 552-559, Apr. 2012.
- [16] Buendia, F. and Cano, J.C., "WebGeneOS: A Generative and Web-based Learning Architecture to Teach Operating System in Undergraduate Courses," *IEEE Transactions on Education*, vol. 49, no. 4, pp. 464-473, Apr. 2006.



고정국(Jeong-Gook Koh)

1992년 2월 부산대학교 컴퓨터공학과(공학사)
 1994년 2월 부산대학교 컴퓨터공학과(공학석사)
 1999년 2월 부산대학교 컴퓨터공학과(공학박사)
 1999년 3월 ~ 8월 위덕대학교 컴퓨터공학과 전임강사
 1999년 9월 ~ 현재 동명대학교 컴퓨터공학과 부교수
 ※관심분야 : 운영체제, 컴퓨터활용교육, 인터넷 응용