

연구계획서

제목	Automatic Schedule Assistant (a.k.a ASA)
----	--

참여연구원	학번	이름
	201502023	김민기
	201502101	임승민
	201502093	이익수

< 연구개발의 목적 및 필요성 >

수강신청 할 과목에 대해 미리 시간표를 짜보는 방법은 여러 가지가 있는데, 그 중 충남대에서 대표적으로는 유어유니브라는 홈페이지에서 수강신청을 할 수 있다. 유어유니브는 강좌의 목록을 보여주어 시간표의 양식에 맞게 내가 원하는 데로 짜 볼 수 있고, 이 시간표를 저장하여 유어유니브의 홈페이지에 들어가 내가 과거에 저장했던 시간표를 불러와 수강신청을 할 때 맞는 과목의 학수 번호를 입력하여 수강신청을 할 수 있다. 하지만 이러한 유어유니브의 시간표 미리 짜보는 기능으로는 몇 가지 문제들이 있다.

그 중 하나는 자신이 원하는 교과목을 미리 알아보고 선택하는 데에 있어서 많은 시간이 든다는 것이다. 유어유니브가 시간표 작성을 도와주긴 하지만 사용자 본인이 시간표를 직접 비교하면서 작성을 하기 때문에 시간이 많이 소모 된다. 정작 이렇게 만든 시간표가 수강신청 당일에 실패하는 경우도 발생할 수 있다. 이러한 상황일 때 계획 B, C, D가 없으면 수강신청 정정 기간에 철회와 신청을 반복하게 되어 시간표가 꼬여 계획에도 없는 강좌를 들 수 있는 경우가 생기고, 결국에는 본인이 원하는 교수님의 과목을 수강하지 못하게 될 뿐만 아니라 더 나아가 학점을 못 채우는 경우도 발생한다. 실제로 수강신청 정정 기간이 끝나고 나서야 최종 시간표가 나오는 경우도 주변에서 많이 볼 수 있다.

또한, 공강을 우선시하는 사람들의 경우에도 해당 날짜에 공강을 만들기 위해서 해당 조건에 부합되는 모든 강의를 비교해봐야 한다. 말 그대로 수기로 수강시간표를 짜 넣는 것 대신 온라인으로 적어서 짜 넣는 모양새인 것이다. 이것은 단순히 수업시간이 겹칠 때 팝업창을 띄워 시간이 겹친다는 경고를 띄워주는 것 빼고는 부가적인 옵션이 없다.

< 연구 전체목표와 범위 >

사용자로부터 우선순위를 입력받아 본인이 원하는 가장 합리적인 시간표를 출력해주는 프로그램이다. 시간표는 사용자의 우선순위를 기반으로 가장 적합한 것을 제시해주며, 만약 사용자가 해당 시간표가 마음에 들지 않을 경우 프로그램은 차선의 시간표들을 제시해준다. 그리고 원하는 요일에 공강을 선택하여 그 요일을 제외하고 시간표를 생성할 수도 있다. 또한 하루에 듣는 강의 수가 많으면 강의를 듣기도 힘들고 시험이 같은 날에 겹쳐서 힘들 수도 있는데, 이런 경우를 대비하여 하루에 듣는 강의의 개수를 정하는 기능도 추가한다.

수강 신청을 할 경우에 전공과목은 정해진 틀이 있을 뿐더러 중요도가 높기 때문에 우선적으로 선택하는 경우가 많다. 그렇기에 교양과목 같은 경우에는 미리 형성된 시간표에서 남는 시간에 수강 가능한 과목을 찾아서 넣어버리는 경우가 많다. 이러한 점을 토대로 ASA에서는 일일이 해당 시간에 맞는 과목을 본인이 직접 찾을 필요 없이 남은 학점만큼 내가 들을 수 있는 교양과목을 추천해 주는 기능을 추가한다. 이러한 기능들을 추가하여 프로그램을 개발함으로써 사용자가 시간표를 직접 비교하는 번거로움을 줄일 수 있다. 사용자의 범위는 일단 충남대학교 컴퓨터공학과 학생들을 기준으로 하였으며, 시간이 된다면 사용자의 범위를 더 늘릴 수 있도록 할 계획이다.

위와 같은 계획을 만족시키기 위해서는 다음과 같은 코드구축 과정이 필요할 것으로 생각된다. 데이터베이스와 검색을 위한 트리형태의 알고리즘이 대표적인 예이며 이 부분을 한 명씩 도맡아서 코드를 구현하여 목표를 만족시키는 동시에 부가적인 기능을 고려하여 진행을 해나갈 방침이다.

<연구개발의 구체적 내용 및 방법>

우선 이 프로젝트에서 가장 중요하면서 우선적으로 구현해야 하는 것은 시간표를 자동으로 작성하는 알고리즘을 개발하는 것이다. 이를 만족하기 위해서는 다음과 같은 부가적인 요소가 필수불가결하다. 우선 바탕이 될 데이터가 필요하다. 모든 학과에는 해당 학생이 들어야하는 전공이 있으므로 이를 위해서는 학과정보가 요구되며 또한 다양한 과목들 각각의 강의 날짜와 시간, 해당 과목을 담당하고 있는 교수님들의 정보, 분반 등 여러 가지 정보들도 포함되어야 한다.

이러한 정보를 확보하고 있는 상태에서 사용자로부터 정보를 입력받는다. 입력받게 되는 정보는 원하는 과목의 우선순위이며 만일 해당 과목이 다수의 교수님, 분반으로 나뉘어져있는 경우에는 각 경우에 맞는 우선순위 또한 입력받는다. 위의 데이터베이스와 입력받은 정보를 이용하여 우선순위 순의 트리 형태로 프로그램 내부에 저장 후에 트리를 순서대로 순회하면서 각각의 경우를 비교한다. 만일 트리 내부에서 데이터들을 비교했을 때 두 과목의 시간이 겹치게 된다면 해당 서브트리는 자동으로 무시되고 다음으로 넘어가게 된다. 트리를 모두 탐색하고 나면 위의 요구사항을 만족하는 시간표가 여러 개 나올 것이고 우선순위 순으로 트리를 탐색했으므로 사용자의 만족도가 높을 것으로 예상되는 시간표 순으로 결과가 나올 것이다. 여기서 사용자가 첫 번째로 출력된 시간표를 마음에 들어 하지 않을 수도 있으므로 해당 시간표를 얼마나 출력할 것인지도 입력받아야 한다. 만일 10이라는 숫자를 입력받으면 해당 프로그램은 우선순위 1부터 10까지의 시간표들을 출력시켜 줄 것이며 사용자는 이 10개의 시간표 중 마음에 드는 것을 골라 사용하면 된다.

그리고 위의 기능에서 추가적으로 원하는 요일에 공강을 선택하는 옵션을 추가하여 해당 요일을 제외하는 시간표를 생성하거나 하루에 듣는 강의의 수를 제한하는 등의 사용자에게 필요할 것이라고 생각되는 기능을 추가할 수 있다. 공강은 해당 옵션 자체를 해당 날짜 전체 시간을 차지하는 과목형태로 옵션에 추가하여 구현할 계획이며, 트리를 탐색할 때 하루에 듣는 강의의 수를 넘어가면 트리를 넘기는 형식으로 추가할 수 있다.

<연구 추진 일정 및 계획 >

- 10. 04 ~ 10. 10 :

요구사항명세서 작성. 작성하면서 더 개선된 아이디어 혹은 수정해야 될 부분 등과 같은 여러 가지 사항에 대해서 토의를 하면서 대략적인 프로젝트의 방향성을 확립 및 초안을 작성한다.

- 10. 11 ~ 10. 25 :

유스케이스 명세서 작성. 계획을 거의 확정짓는 단계로 MVC 모델링을 구조로 하여 대략적인 코드의 형태를 객체지향적으로 설계한다.

- 10. 26 ~ 11. 08 :

각각 맡은 역할을 수행하여 코드를 작성한다. 트리알고리즘과 DB를 작성하고 MVC 모델을 만족하는 뼈대를 코드로 작성해놓는다. 이 때 팀원들 간의 메소드 명칭이나 호출에 대해서 지속적으로 의견의 교환이 있어야 하며 작성된 코드를 하나로 규합할 때 최대한 적은 오류가 발생할 수 있게 노력한다.

- 11. 09 ~ 11. 21 :

트리알고리즘과 DB를 완성시킨 후, 각자가 작성한 모든 코드를 하나로 합친다. 이 때 발생할 수 있는 수많은 에러를 테스팅과 디버깅을 통해 해결하는 과정을 거친다. 이때 썸 베타버전의 실행 가능한 프로그램을 완성시킬 것으로 예상된다.

- 11. 22 ~ 12. 13 :

디버깅을 최대한 마무리하고 팀원들 간의 토의를 통해 보완할 곳과 부족한 곳들을 파악하여 최대한 개선하고 수정한다.

<연구팀 구성 및 역할>

201502023 김민기(조장) - 총괄지휘 및 코드 통합, GUI

이 프로젝트에서는 트리알고리즘, DB와 같은 코드가 사용될 것으로 예상되는데 이러한 부분을 확인하고 규합하는 과정을 담당하여 각자가 작성한 코드들이 모두 원활하게 돌아갈 수 있게 하는 역할이 반드시 필요하다. 따라서 이러한 역할을 조장이 담당하여 전체적인 개발방향을 지휘함과 동시에 코드를 통합하여 완성도 있는 프로젝트를 만들 수 있게 하며 또한 프로젝트에서 출력되는 결과를 보여줄 GUI를 설계 및 구현함으로써 프로젝트의 전체적인 마무리를 책임진다.

201502101 임승민 - 트리알고리즘 및 예외처리 담당

각 과목마다 입력된 우선순위 및 과목정보 등을 이용하여 만족도가 높을 것으로 생각되는, 즉, 종합된 우선순위 순으로 결과를 출력할 수 있는 알고리즘을 담당한다. 해당 알고리즘은 트리형태로 구현할 것으로 예상된다. 또한 입출력, 혹은 메소드를 호출할 때 발생할 수 있는 모든 일련의 예외사항을 처리할 수 있게 하여 완성도를 높인다.

201502093 이익수 - 데이터베이스 및 테스트 담당

알고리즘에서 우선순위를 비교 및 설정하기 위해서 해당 과목들의 정보가 담겨있는 데이터들을 DB에서 받아와야 한다. 이에 관련된 모든 부분을 담당하며 또한 중간중간 디버깅 및 테스트를 통하여 수정할 부분이 있으면 리팩토링을 통해 코드를 개선하는 과정을 시행한다.