|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **공개 소프트웨어 프로젝트 최종보고서** | | | | | | | | |
| **과제 수행원 현황** | | | | | | | | |
| **수행 학기** | □2018년 1월~6월     ◼2018년 3월~6월     □2017년 9월~2018년 6월 | | | | | | | |
| **프로젝트명** | 동국대학교 컴퓨터 공학과 학과 시간표 자동 생성 프로그램 | | | | | | | |
| **팀명** | 2조 | | | | | | | |
|  | **학과** | | **학번** | **성명** | | **성별** | **연락처** | **E-mail** |
| **팀장** | 컴퓨터 공학과 | | 2014112254 | 박세진 | | 남 | 010-5591-8544 | Leo8544@naver.com |
| **팀원** | 컴퓨터 공학과 | | 2016112185 | 박지수 | | 여 | 010-7371-8976 | qkrwltn8976@naver.com |
| 컴퓨터 공학과 | | 2016112128 | 지선아 | | 여 | 010-9185-9271 | jisuna314@gmail.com |
| 컴퓨터 공학과 | | 2016112116 | 정경원 | | 여 | 010-6577-5994 | Tab0228@naver.com |
|  | |  |  | |  |  |  |
| **지도교수** | **교과목명** | | 공개 SW 프로젝트 | | | | | |
| **소속** | | ◼컴퓨터공학전공               □정보통신공학전공   □멀티미디어공학전공          □융합소프트웨어연계전공 | | | | | |
| **성명** | | 손윤식 교수님 | | | | | |
|  | | | | | | | | |
| **# Key Words** | 자동 생성 | 유전 알고리즘 | | | UI | |  |  |
| **1.개발동기/**  **목적/필요성 및 개발 목표** | 처음 프로젝트 주제는 “개인시간표 자동 생성 프로그램” 이었지만 “개인시간표 자동 생성 프로그램” 은 고려해야 할 점이 너무 단순해서 프로젝트로서의 의미가 없다는 교수님의 피드백을 받고 학과 시간표 생성프로그램으로 주제를 바꾸게 되었다.   실제로 동국대학교 컴퓨터공학과 학과 사무실 직원분들께 학과시간표에 대해 문의를 드린 결과 학과 사무실에서 프로그램으로 시간표를 짜는 것이 아니라 작년도 시간표를 참고해서 직접 짠다는 것을 알게 되었고 이를 통해 학과 시간표 생성프로그램의 필요성을 확인할 수 있었다.  또한 유사시스템을 조사해본 결과 특정 대학의 특정 학과시간표를 짜기 위해서는 그 학과만의 특수한 조건들이 충족되어야 하는데 유사시스템들은 기본적인 조건들로 프로그램이 구현되어있기 때문에 직접 적용하는데 어려움이 있다는 것을 알게 되었다. 따라서 학과사무실에서 실제로 쓸 수 있는 시간표 생성 프로그램을 만들어야겠다는 목표를 가지게 되었다  우리의 개발 목표는 기존 오픈소스의 제약조건 이외의 동국대학교 컴퓨터공학과의 여러 조건이 적용된 최적의 시간표 생성 프로그램을 만드는 것이다.  **기존 오픈소스**의 필수조건은 아래와 같다  1. 학생, 교수는 한번에 여러 개의 수업에 배정될 수 없다.  2. 수업은 반드시 빈 교실에 배정되어야 한다.  3. 교실은 모든 학생을 수용할 수 있는 충분한 자리가 있어야 한다.  4. 수업에 요구될 경우 실습실이 있어야 한다.  **동국대학교 컴퓨터공학과**의 조건은 아래와 같다  1. 수업시간에 수업을 배정하지 않는다  2. 3시간짜리 수업은 2개로 분할한다  3. 퇴근시간 이후는 수업을 배정하지 않는다  4. 특정 과목(컴종설 등)은 같은 시간 같은 요일에 고정시킨다.  5. 이론 수업은 일반 강의실, 실습 수업은 실습실로 배정한다  6. 3학점 수업은 3시간으로 고정되던 것을 실습수업이면 4시간으로 변경되도록 한다.  7. 같은 수업은 같은 강의실에 하루 이상 건너서 배치한다.  또한 기존 오픈소스의 UI를 개선시킬 것이다. 특히 기존 오픈소스는 텍스트파일로만 입력이 가능하기 때문에 엑셀입력기능을 추가시켜 사용성을 높일 것이고 교수님 별로 시간표의 색상을 변경하여 가독성을 높일 것이고 시간표의 디자인을 감각적으로 변화시킬 것이다. | | | | | | | |
| **2.최종**  **결과물 소개** | 1. 시작화면에는 CONVERT DATA, LOAD DATA, START 3개의 버튼으로 구성되어 있다. 2. 시작화면의 CONVERT DATA버튼을 누르면 위와 같이 엑셀 파일 선택 화면과 옆의 윈도우 창에 설명이 뜬다. 설명에 적혀 있는 바와 같이 첫 번째로 수업정보.csv 와 강의실 정보.csv 파일을 차례로 선택한다. 그 결과로 output.txt 파일이 자동으로 생성된다. 3. C:\Users\정경원\Documents\카카오톡 받은 파일\KakaoTalk_20180621_222035716.png2번 과정에서 생성된txt 파일은 두 개의 엑셀 파일을 프로그램에서 지정한 형식의 txt 파일이다. LOAD DATA 버튼을 누르면 옆의 윈도우 창에 output.txt 파일을 선택 하라는 설명과 파일 선택 창이 뜬다. output.txt 파일을 선택하면 프로그램에 시간표 데이터가 로드 된다. 4. START 버튼을 누르면 시간표 생성이 시작된다. 시간표 창에는 왼쪽 상단에 적합도와 세대 수에 대한 정보가 출력된다. 적합도가 1이 되었을 때 모든 조건을 만족하는 시간표가 완성된다. | | | | | | | |
| **3.프로젝트  추진 내용** | 프로젝트 진행과정  \* 해당항목은 추후 디렉토리 및 보도자료로 활용 예정이오니 상세하고 쉽게 작성 요망  2018.04.02 오픈소스 선택  2018.04.03 visual studio 2017버전에 맞게 소스코드 수정  2018.04.04 한글 텍스트 인식 가능하도록 수정  2018.04.06~05.02 유전 알고리즘 및 코드 분석  2018.05.04~06.10 엔진 파트 : 세부 기능 추가  2018.05.03~05.12 엑셀 파일로 데이터 수정하도록 convert data 기능 구현  2018.05.10~05.11 홈 UI 수정 : 스크롤바 기능 추가  2018.05.18~05.26 홈 UI 수정 : 메인 화면에 버튼 추가  2018.05.28 엔진 & 디자인 1차 병합  2018.06.10~06.13 홈 UI 수정 : 교수님별 색상 추가  2018.06.12 엔진 & 디자인 2차 병합  2018.06.21 엔진 & 디자인 최종 병합 | | | | | | | |
| 2. 프로젝트 구현과정  \* 개념설계 / 상세설계 / 구현 이론 등 문제 해결 방안 및 과정 설명  그림과 표의 하단 및 상단에 제목을 기입할 것  [스크롤 바 코드]  스크롤 바 끝에 있는 버튼을 누르면 다음 시간표로 넘어가도록 구현하기 위해서 시간표의 길이를 구해서 SB\_LINELEFT, SB\_LINERIGHT의 단위1을 시간표 길이로 변경시켰다.    [GDI객체 계층도]  시간표를 감각적으로 디자인하기 위해서 시간표의 색상 변경, 테두리 제거, 폰트 변경 등의 작업을 수행하였다.  먼저 GDI객체를 생성한 후 CPen으로 시간표의 테두리 색을 변경하여 시간표끼리 겹쳤을 때 구분이 가능하도록 하였으며 교수님들의 이름을 받아와서 교수님 별로 특정 색의 시간표를 갖도록 구현하였다. 또한 시간표 틀의 테두리를 투명으로 바꿔서 감각적으로 디자인하였다.  http://postfiles2.naver.net/20140211_145/toplms78_1392085196064KbXvG_JPEG/1.jpg?type=w2  [CSV 파일 및 기타 엑셀 파일 비교]  Convert Data 기능 구현  사용자의 편의를 위해 데이터를 쉽게 조작할 수 있도록 csv파일에 강의실 정보와 수업 정보를 입력하여 프로그램에서 데이터 파일로 사용되는 txt파일로 변환하도록 하였다. csv파일은 각 셀이 comma(콤마)로 구분되어 있어 비주얼스튜디오에서 엑셀 파일을 입출력 하기에 가장 간편한 방법이라 판단하여 기존의 xml파일을 csv파일로 변환하여 사용하였다.  엑셀파일을 csv파일로 변환하면 텍스트 입출력과 동일하게 C++의 fstream함수와 getline함수를 사용하여 파일의 데이터를 읽어올 수 있다. (열은 콤마(,), 행은 개행문자(\n)로 구분되어 있다.)    [MFC 프로그램 구성요소]  mainframe에 버튼 추가  기존 오픈소스의 프로젝트가 MFC dialog가 아닌 single document로 되어있어 디자인을 수정하는데 어려움이 있었다. 단순하게 mainframe의 onCreate함수에 버튼을 추가하였을 경우 기존 코드에서 이미 화면을 생성하는 동시에 또 다른 자식 view를 생성하여 추가된 버튼이 자식 view에 가려져 활성화되지 않았다. 따라서 버튼이 활성화되도록 하기 위해 생성된 자식 view를 비활성화시켜 놓고 버튼 기능을 수행 완료 한 뒤 다시 자식 뷰를 활성화시켜 자식 view에서 생성된 시간표를 띄우도록 구현을 하였다.  \* 기존 작품을 Upgrade한 경우, 기존 작품과의 차이점 기술 요망  우선 시각적으로 Upgrade한 부분은 다음과 같다. 기존 작품에서는 시간 간격이 1시간이였으나 현재 Upgrade된 작품에서는 30분 간격으로 현재 학사제도의 기준을 반영했다. 그리고 디자인 측면에서 상당히 발전시켰다. 우선 예전 버전에서는 모든 시간표가 동일한 색깔로 되어 있고 크기도 한눈에 들어오지 않아서 시간표를 읽는 것 자체도 불편했다. 하지만 현재 버전에서는 교수님 마다 시간표 색을 다르게 함으로써 보다 보기 좋게 업그레이드 했다.  알고리즘 측면에서 Upgrade한 부분은 다음과 같다. 이전 버전은 단순히 과목을 입력하면 무작위로 수업을 배치하는 방식으로 시간표를 만들었다. 즉 해당 과목이 특정한 시간, 특정한 요일에 위치 시키는 기능 자체가 없었다. 그러다 보니 우리가 생각하는 이상적인 시간표는 얻을 수 없었다. 예를 들어서 모든 시간표가 월요일에 가득 차 있을 수도 있고 점심시간이나 퇴근시간에 수업들이 몰릴 수 있다는 것이다. 더구나 현실에서(동국대 시간표를 기준으로 생각하자면) 시간표를 만드는 작업 자체를 무작위성에 기대서 만들 수가 없다. 왜냐하면 현재 학사제도라는 것이 그리 간단치 않기 때문이다. 가장 간단한 문제는 시간대이다. 현재 컴퓨터공학과에서 12시부터 1시까지는 교수님들의 회의 시간으로 배정되어 있다. 만약 이때 시간표가 배치가 되면 이 시간표는 이상적인 시간표가 될 수 없다. 그리고 다소 복합한 문제는 다음과 같다. 우선 3학점 이상인 수업은 두 번에 나누어서 수업을 진행해야 한다. 즉 하나의 input으로 2개의 output을 내야 한다는 의미이다. 대부분의 사람들은 그게 문제라면 input을 2개로 늘리면 되지 않느냐라고 생각할 수도 있다. 그렇게 되면 직관적으로 그러한 문제는 자연스럽게 해결되기 때문이다. 하지만 불행히도 이후 더 까다로운 조건이 존재한다. 이렇게 나누어진 2개의 수업은 서로를 인식해야 한다. 서로를 인식하는 이 추상적인 말이 무슨 뜻인지 쉽게 와 닿지 않을 것이다. 쉽게 말해 서로 영향을 미친다는 의미이다. 예를 들어 형식언어는 3학점이므로 하나의 수업을 2개로 나누어서 배치해야 한다. 이때 수업 1개가 월요일에 배치 되었다면 나머지 수업은 적어도 수요일 이후에 배치되어야 한다. 그리고 같은 방으로 배치해야 한다. 즉, 서로 하루 이상의 간격을 두고 같은 방에 배치해야 한다는 의미이다. 이를 구현하기 어려운 이유는 형식언어라는 수업을 단순히 나누더라도 이 수업이 어떤 수업에서 나온 건지 알 수 없기 때문이다. 예를 들어 형식언어는 분반 자체가 4개나 존재한다. 이때 수업을 단순히 4개에서 8개로 나누었을 때 어떤 수업과 어떤 수업이 같은 분반인지 알 수 없다는 것이다. 그렇기 때문에 단순히 수업을 즉 input을 늘리는 방식으로 이러한 문제를 해결 할 수 없다. 그리고 이렇게 했을 경우 분반 표시 자체를 수동으로 작성해야 하는 번거로움이 존재한다. 그 다음 문제로는 실습이 있는 수업은 3학점이라도 단순히 1시간 30분씩 2번에 나누어서 수업을 진행하는 것이 아니라 2시간씩 수업을 진행한다. 한마디로 일반 수업과 실습 수업은 또 다르게 처리해야 하는 것이다. 또 다른 문제로는 교수님의 취향에 따라 시간표를 작성해야 한다는 것이다. 예를 들어 어떠한 교수님은 실습 수업이라도 이론 수업은 일반강의실에서 하기를 원하고 다른 교수님은 이론 수업도 실습실에서 수업하기를 원하신다. 이렇게 되었을 때 정말 문제가 복잡해진다. 예전 버전에서는 이 수업이 실습 여부만 입력 가능하기 때문이다(사실 예전 버전에서는 한 개의 수업을 2개로 나누는 것 자체가 불가능하기 때문에 이 문제자체를 해결 할 수 없다.). 마지막 조건으로 특정한 과목(예. 컴퓨터 종합설계)은 모두 같은 요일 같은 시간대에 열려야 한다는 것이다. 이렇게 제약사항이 너무 많고 복잡하기 때문에 기존 알고리즘처럼 무작위성에서만 의존한다면 우리가 원하는 이상적인 시간표는 사실상 만들어질 수 없다. 즉, 원숭이가 타자기를 무작위를 두드렸을 때 셰익스피어의 햄릿이 나오기를 기대하는 것과 같다. 이를 해결하기 위해서는 단순히 무작위성에 의존할 수 없다. 즉 프로그래머가 진화의 방향을 잡아줘야 한다는 것이다. 진화의 방향을 잡아 준다는 의미는 생각보다 단순하다. 임의로 시간표를 배치했을 경우, 경우의 수가 너무 많기 때문에 몇몇 조건을 걸어줘서 필요 없는 경우는 사전에 차단하는 것이다(사실 이러한 제약만으로 위 조건을 모두 충족시킬 수 없다. 업그레이드된 알고리즘을 설명하기에는 너무 길고 복잡하기 때문에 자세한 설명은 생략하겠다.). 이러한 식으로 사전에 필요 없는 경우의 수를 줄이고 특이한 조건을 만족시키는 알고리즘을 추가 시켜서 우리 원하는 다양한 조건을 만족시키는 즉 지금 당장이라도 사용할 수 있는 ‘학과 시간표 자동 생성기’를 완성했다.  다음은 알고리즘 측면에서 업그레이드 된 부분이다.  [기능 1. 12~13시는 교수님들 회의 시간이므로 수업을 배치하면 안 된다.]  [기능 2. 3학점 이상인 수업은 두 번으로 나눈다.]  [기능 3. 실습 수업인 경우에는 3학점이라도 2시간씩 수업을 배치한다.]  [기능 4. 그렇게 나누어진 수업은 하루 이상의 간격을 유지해야 한다.]  [기능 5. 그렇게 나누어진 수업은 같은 방으로 배치되도록 한다.]    [기능 6. 실습실이 부족하므로 실습이 없는 수업은 반드시 일반 수업에 배치]  [기능 7. 교수님의 취향에 따라 실습 수업이라도 이론은 일반실에 배치 가능]    이론은 일반실로 배치를 원할 경우 ‘lab\_H = true’로 입력  모두 실습실로 배치를 원할 경우 ‘lab = true’로 입력      [기능 8. 특정 과목은 같은 요일, 같은 시간대에서 배치]  [기능 9. 특정 과목 사용자가 원하는 요일 시간대 배치 가능]      F-13(요일-시간대)은 금요일 13시에 배치하도록 설정하는 것이다. 이때 이 수업 역시 3학점 이상인 수업이므로 금요일로 입력 시 자동으로 수-금으로 설정된다. 즉, 앞 뒤 하루 간격으로 요일을 배치하도록 설정했다. (M: 월, t: 화, M: 수, T: 목, F: 금 이렇게 설정 되어 있으므로 요일 설정 시 편의에 맞게 설정하면 된다.) | | | | | | | |
| **4.기대효과** | 학과 시간표 자동 생성 프로그램은 유전 알고리즘을 이용해 시간표를 생성하는 것이기 때문에 더 효율적이고 각 조건에 잘 맞는 최적의 시간표를 만들 수 있다. 시간표를 프로그램을 이용해 자동으로 생성함으로써 시간표 작성에 많은 시간을 절약할 수 있는 효과도 있다. 또한, 각 학기 별로 달라지는 수업과 강의실의 상황을 강의실 정보와 수업 정보 엑셀 파일을 통해 쉽게 변경해서 적용이 가능하다. 따라서, 시간표 자동 생성 프로그램은 획기적으로 시간을 절약하고 불만을 최소화 할 수 있는 효율적인 시간표를 작성 할 수 있게 한다. | | | | | | | |
| **5.구현과정에서의 문제점 및 해결과정** | 1. 엔진 파트 2. 한 개의 수업만을 입력으로 받아서 2개의 수업으로 만들려고 할 때 초기 단계에서 제일 난제였다. 3. 유전 알고리즘 자체가 무작위성에 기반해서 시간표를 만드는 것이다. 그런데 ‘2개로 나누어진 수업은 서로 같은 교실 하루 이상의 간격을 두고 시간표를 배치되어야 한다.’라는 까다로운 조건을 만족시키는 알고리즘을 고안하는 것이 불가능해 보였다. 그리고 만약 이 문제를 해결하지 못한다면 우리 프로젝트는 실패로 돌아가기 때문에 더욱 더 부담이 되었다. 4. 여러 까다로운 조건들을 모두 만족시키는 알고리즘을 모두 고안했지만 실험실이 있는 교실이 적어서 우리가 원하는 시간표를 얻을 수가 없었다 그리고 컴종 같이 고정된 수업들은 Mutation을 해도 잘 변동되지 않는다. 이러한 문제 때문에 최적해를 얻는데 상당히 오랜 시간이 아니 사실상 최적해를 얻을 수 없었다.. 5. 디자인 파트 6. 시간표 출력을 강의실 별로 출력하기 위해 메뉴바에 강의실 항목을 추가하고 원하는 강의실 버튼을 선택하면 강의실에 해당하는 시간표를 출력하려 했으나 구현에 어려움을 겪었다. 7. 시작화면에 있는 3개의 버튼 중 START 버튼을 누른 후 시간표가 생성될 때 스크롤이 2중으로 생기는 문제점이 발생했다. 8. 시간표에서 과목별로 구분을 쉽게 하기 위해 과목별로 색상을 변경하려 했지만 과목은 학기 별로 달라지기 때문에 과목별로 색을 지정해 출력할 수 없었다. | | | | | | | |
| 1. 엔진 파트 2. 처음 이 문제를 해결할 때 입력파일을 처리하는 부분은 집중적으로 분석하지 않아서 쉽게 해결책을 찾지 못했다. 하지만 입력파일을 분석하는 부분을 집중적으로 분석했더니 하나의 입력 수업만으로도 2개의 수업을 만들 수 있었다. 3. 우선 서로 같은 수업은 서로 인식 할 수 있도록 Node처럼 서로의 주솟값을 정장하도록 하였다. 그리고 먼저 입력된 수업의 정보를 기반으로 두 번째 수업을 배치하니 이 조건을 해결 할 수 있었다. 그러나 CrossOver를 할 때는 서로 다른 염색체끼리 섞이기 때문에 서로의 정보를 공유할 수 없었다. 그래서 과감하게 CrossOver를 안하고 Mutation을 엄청 많이 하도록 수정했다. 그랬더니 위 조건은 유지되면서 경우의 수를 계속 늘릴 수 있었다. 4. 실험실이 적은 환경 이여서 실험실에 배치되는 수업에 따라 접합도 1이 나오나 나오지 않느냐 결정 되었다. 그래서 반드시 실험실에서 수업을 해야 하는 수업만 실험실 수업에 배치되도록 했다. 그리고 컴종은 각각의 실험실로 배치가 되도록 설정했다. 그랬더니 빠른 시간 내에 최적해를 얻을 수 있었다.   2. 디자인 파트  1) 시간표를 한 화면에 전부 출력하는 방식으로 변경했다. 하지만 여러 시간표가 세로로 이어져 출력됐기 때문에 보기 불편하다는 문제점이 있었다. 따라서, 시간표를 보기 편하게 하기 위해 스크롤 바를 시간표 크기로 이동하도록 수정했다.  2) 버튼 클릭 시 창을 새로 만드는 기존 소스코드는 스크롤바도 새로 만들도록 구현이 되어있었기 때문에 2중스크롤바 문제가 있었다. 따라서 창을 새로 만들 때 스크롤바는 생성하지 않도록 수정하여 2중스크롤바 문제를 해결하였다.  3) 담당 교원 목록은 학기 별로 달라지지 않기 때문에 교원 별 색상을 지정해 출력하여 해결했다. | | | | | | | |
| **6.참고문헌** | 인터넷 강의: Visual C++ 2013 (MFC) 제대로 배우기, 강사: 윤영혜, 모든 에듀  강경우. MFC 윈도우 프로그래밍. 한빛 아카데미. 2017  이아리. Visual C++ MFC 2017 윈도우 프로그래밍. 북스홀릭퍼블리싱. 2017  <https://www.youtube.com/watch?v=rmCr7GkAVtM&index=4&list=PL7mmuO705dG1o2j2p0oIYBu4NraxHvEIm>  (mfc programming)  <https://ko.wikipedia.org/wiki/%EC%9C%A0%EC%A0%84_%EC%95%8C%EA%B3%A0%EB%A6%AC%EC%A6%98>  <http://twinw.tistory.com/1> (유전 알고리즘에 대한 설명) | | | | | | | |