**결과 보고서**

**-머신 러닝을 이용한 영화 추천 프로그램 -**

9조

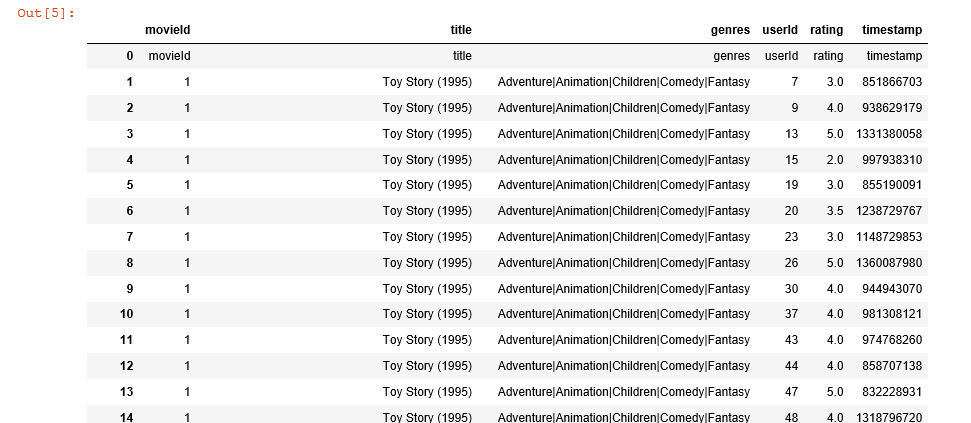
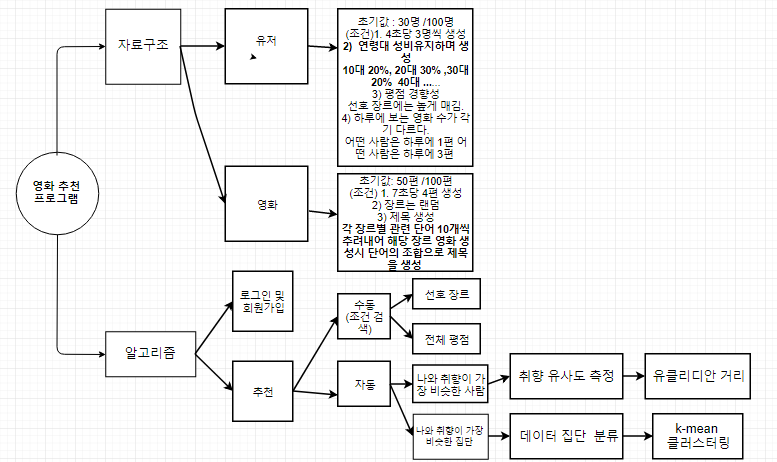
학번 이름

|  |  |
| --- | --- |
| 2017707049 | 안시현 |
| 2014707083 | 원동욱 |
| 2015707058 | 이재성 |
| 2017707037 | 이휘연 |

1. 주제 선정 배경

과목에서 배운 이론적인 부분들을 활용하고 현재 기술 트렌드인 머신 러닝을 학습하고 싶었고 또한, 소프트웨어 관련해서 설계 역량을 기르고 싶었다. 그래서 영화 추천 프로그램이 적합하다 생각하여 이를 주제로 선정하였다.

1. 설계 과정



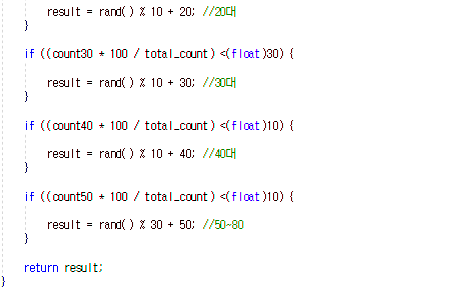
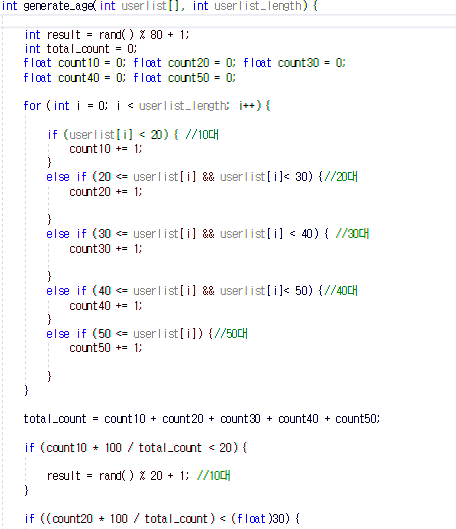
파이썬 통합개발툴인 아나콘다를 이용하여 프로젝트를 구현하려 했으나 현실적으로 공부할 시간이 많이 부족하여 결국 중도 포기하고 C언어로 구현했습니다.

설계하기에 앞서 역할 분담은 조원 이재성군이 화면 인터페이스(결과창) 개발, 나머지는 자료 구조를 구현하고 알고리즘을 개발했습니다.

우선, 자료 구조는 Generic코드로 구현한 Genereal Linear list를 이용하려 했으나 데이터 몇 개 집어넣자마자 메모리가 고갈되어 마땅한 자료 구조를 찾지 못하여 그냥 배열로 정적할당했습니다.

1. 확률 분포에 의거하여 연령 생성

/\*================= 10대이하 20% 20대 30% 30대 30% 40대 10% 50대이상 10% =============\*/



1. 영화 제목 생성



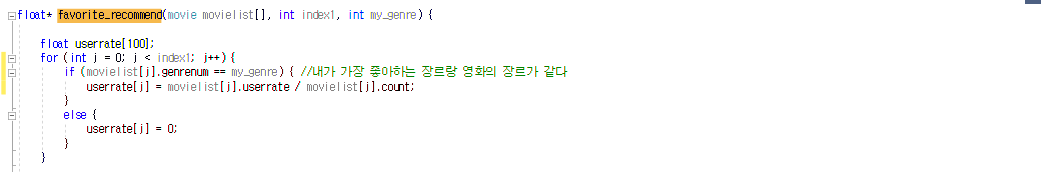
위와 같이 장르별로 관련 단어를 조사하여 랜덤으로 추출하여 결합해서 제목을 생성했다.

1. 데이터 베이스 갱신

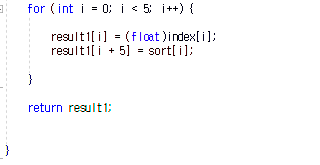


위와 같이 조건을 걸어 유저와 영화를 새로 만들었다.

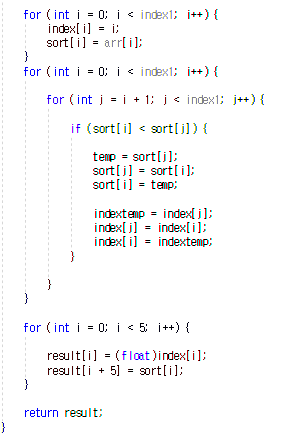
1. 좋아하는 장르로 추천하는 알고리즘



유저들이 본 영화 중 내가 선호하는 장르만 조건문에 걸리게하여 평점을 구했다. 그리고 나중에정렬을 시켜서 BEST 5만 추려 크기가 10인 result1[]에 집어넣는다. 그중 rate는 result1[5]~result[9] 오름 차순으로 넣고 그 rate에 해당하는 영화의 번호값(인덱스값)을 result1[0]~result[4]에 집어넣는다.. 그리고 다음과 같이 반환한다.

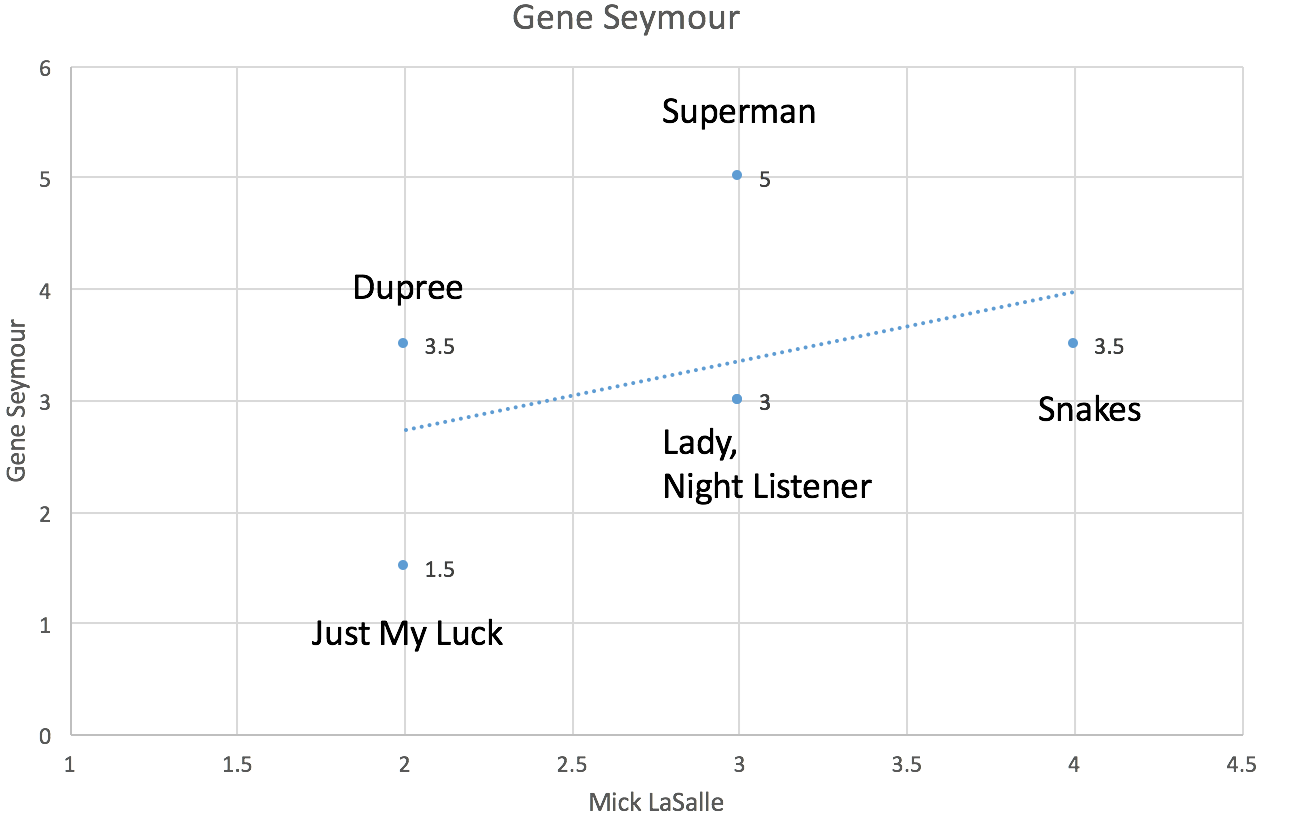


1. 전체 평점으로 추천하는 알고리즘

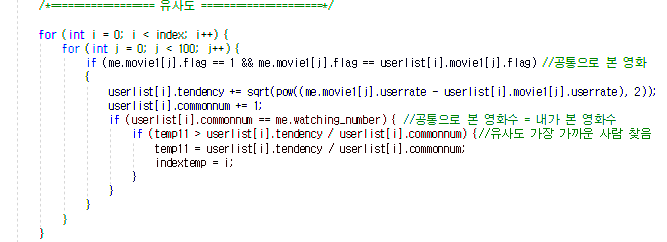


전체 평점을 오름차순으로 정렬해서 favorite\_recommend랑 유사한 방식으로 return한다

1. 유사도를 이용한 추천



점과 점사이의 거리를 이용하여 나와 가장 거리가 가까운 사람을 찾는다.



…..

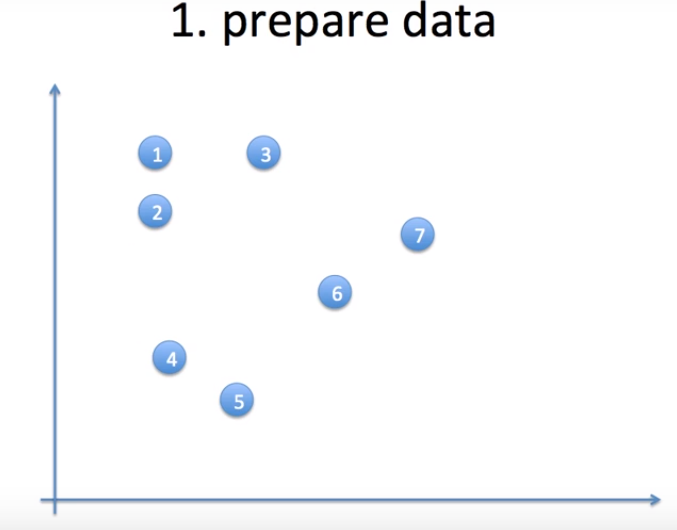


…

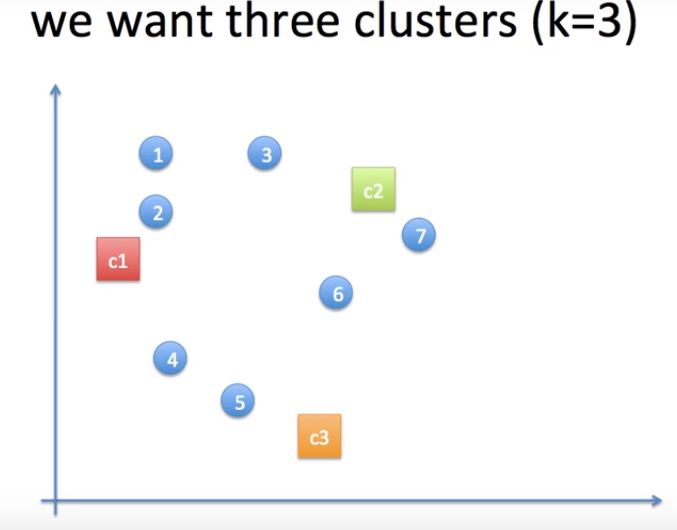
우선, 내가 본 영화 수와 공통으로 본 영화 수가 같은 사람들을 찾고 유클리디안 거리를 구한다. 거기서 가장 가까운 사람을 얻고, 그 사람이 내가 안 본 영화중 평점 가장 높은 것 best 3개를 반환해서 추천한다.

1. K-mean 클러스터링을 이용한 추천

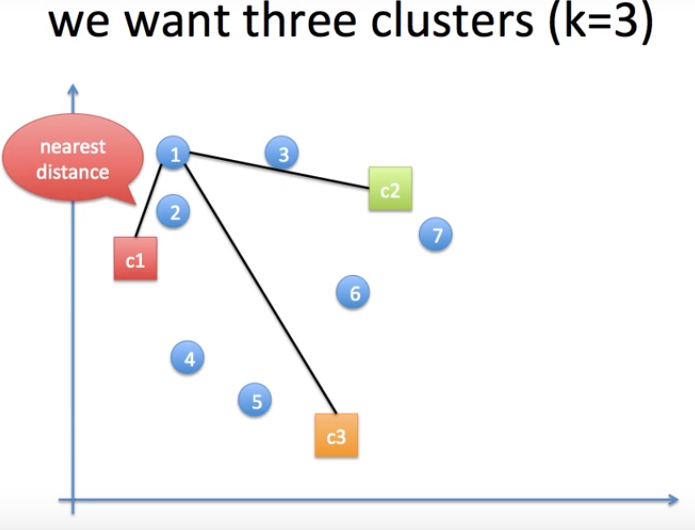
이론 간단하게 소개



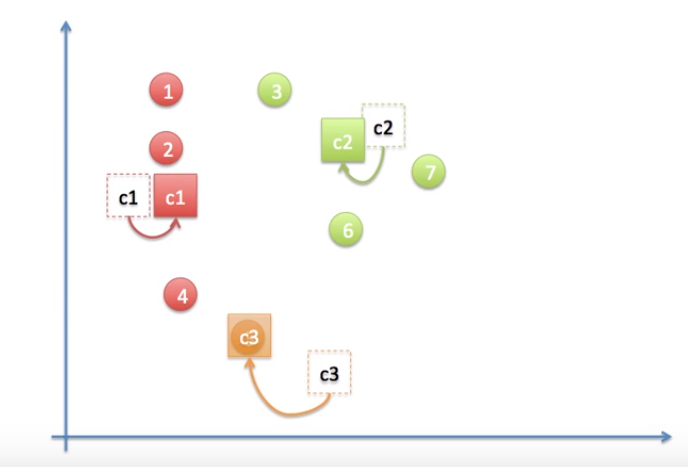
순서를 매기고



가장 가까운 것을 찾아서 자기 집단에 넣는다.



그런 후 중심점으로 센터로이드를 이동시킨다.

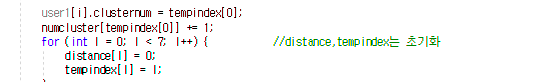
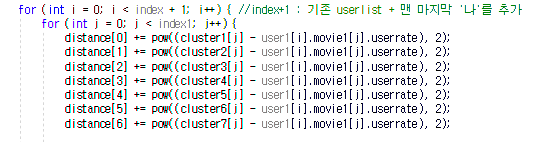


이걸 클러스터의 개수가 변화가 없을 때 까지 반복 행한다.

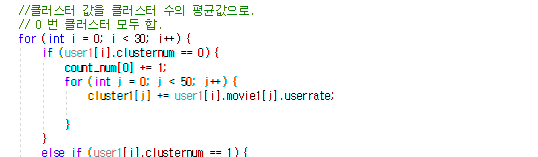
위의 예제는 2개의 벡터에 대해서 클러스터 3개로 설정한 것이고,

아래는 약 100개의 벡터에 대해서 클러스터 7개로 설정하였다.

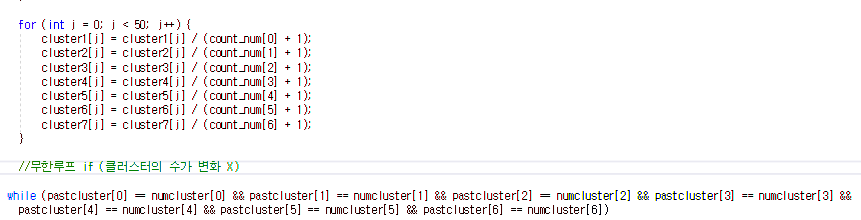
C:\Users\SEC\AppData\Local\Microsoft\Windows\INetCache\Content.Word\K-033.png



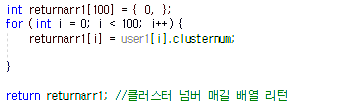
각 거리를 계산하면서 최소 거리를 찾아서 자기 클러스터에 넣는다.



클러스터의 센터로이드를 클러스터의 중심으로 이동시킨다.

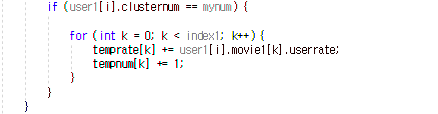


클러스터의 수가 변화가 없을 때까지 루프를 돌린다.

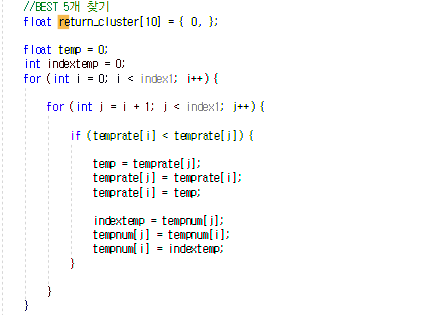


이렇게 해서 클러스터 넘버가 담긴 배열을 리턴함으로써

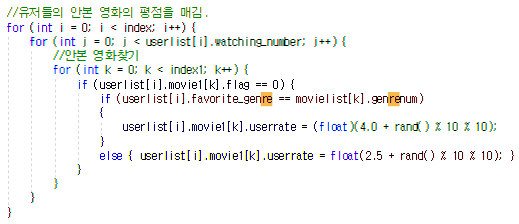
유저리스트의 모든 유저들이 자신의 클러스터 넘버를 할당 받았다. 이제 추천을 하면 된다.



if문에 자기랑 클러스터 넘버가 똑 같은 사람만 추린 후.. 앞의 과정들처럼 오름차순 정렬해서 가장 큰 5개의 값과 인덱스가 담긴 배열을 리턴한다.

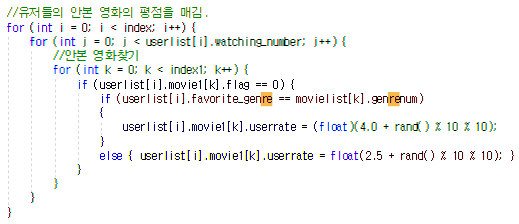


1. 유저들이 본인이 안 본 영화 평점 매기기

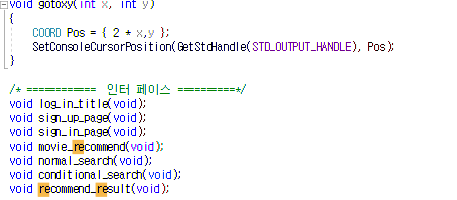


1. 그 밖…

시간이 지남에 따라 유저들이 본인이 안 본 영화를 평점 매긴다.



인터페이스 부분



ㅁ

1. 애로사항 및 해결 방안

* 생성된 영화에 제목을 부여하기 위해 문자열로 이루어진 배열 2개에서 랜덤으로 추출하여 합치는 알고리즘이 있었다. C언어에서 이것을 구현하는 데 상당히 어려움을 겪었다. 찾으려는 문제와 비슷한 시중 서적을 참고해서 해결하려 했으나 비쥬얼 스튜디오의 버전이 달라 다음과 같은 오류가 지속적으로 발생했다. “ const char \* 형식의 값을 사용하여 char \* 엔터티를 할당할 수 없습니다 “ 결국 변수명을 다 상수 char형 포인터로 선언하여 이를 해결하였다.
* 메모리 고갈 : 데이터들을 효과적으로 관리하기 위해 자료구조 및 알고리즘 시간에 배운 Generic coding 으로 구현한 General Linear list을 이용하려 했으나 자료 구조에 데이터를 몇 개 집어 넣자 메모리가 고갈되어 프롬프트가 중지됬다. 그래서 정적으로 배열을 선언해서 자료구조를 만들었다. 영화 Id를 영화배열의 인덱스로 , 유저 아이디 Id를 유저 배열의 인덱스로 설정하여 효율적으로 편하게 데이터를 search하고 관리할 수 있었다.
* 프로젝트 구현 언어 선정 문제:

빅데이터를 다루기 위해 파이썬 통합 개발툴인 아나콘다를 이용하여 프로젝트를 구현하려고 했으나 결국 실패하고 C언어로 구현했다. 애당초에 C언어로 구현했다면 더 나은 결과가 나왔을 텐데하는 아쉬움이 남는다.

* C언어 배열 return 하는 방법 :

C언어는 배열을 return 할 수 없다. 하지만 이것이 가능하다면 상당히 편하다. 많은 함수를 만들지 않아도 된다. 그래서 약간 정통적이지 않은 방법으로 그것을 구현했다. 포인터 함수를 선언해서 포인터 배열을 리턴했다.

float\* total\_recommend(float arr[], int index1) {

float result[10] = { 0, }; //배열 길이 :10

//// 생략 ///

for (int i = 0; i < 5; i++) {

result[i] = (float)index[i]; //배열 0~4는 인덱스

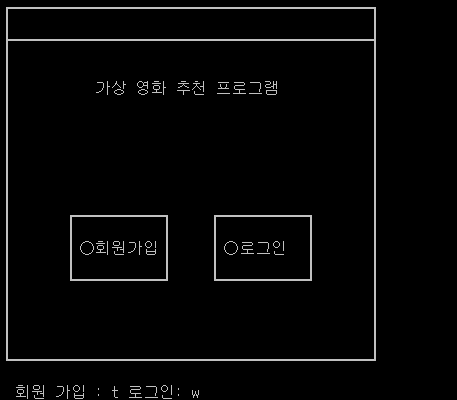
result[i + 5] = sort[i]; //배열 5~9는 0~4의 평점값

}

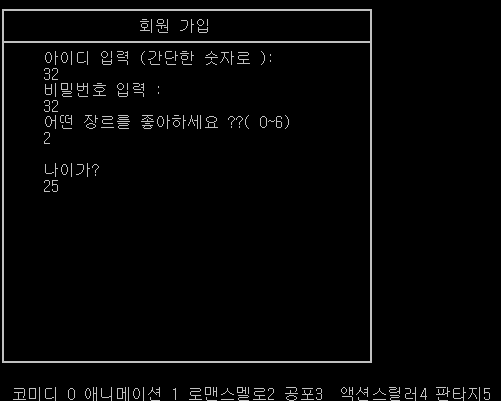
return result;

}

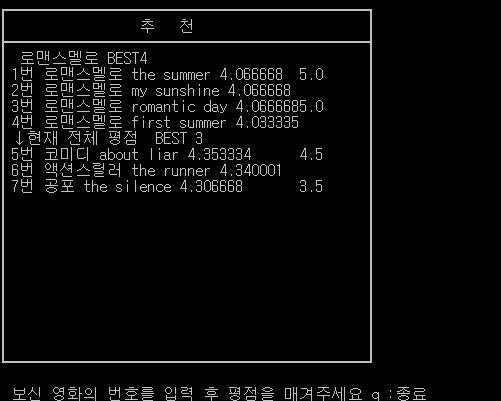
1. 실행결과



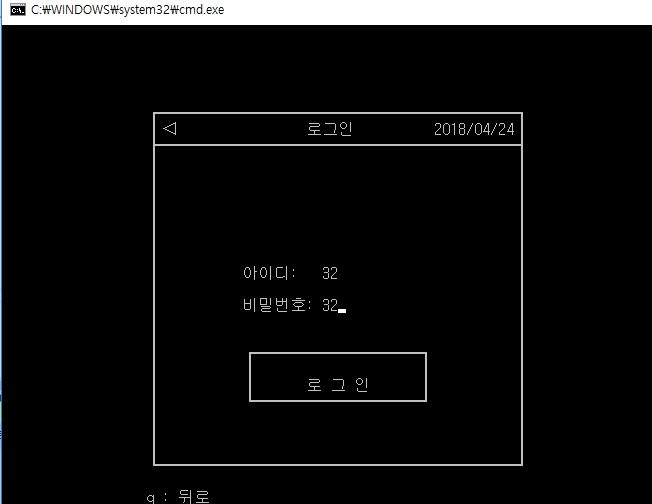
회원가입 클릭



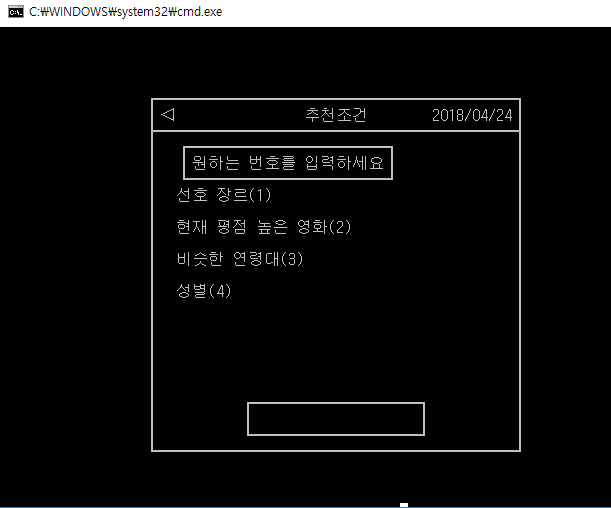
인적 사항 다 입력하면 가장 좋아하는 장르 기반으로 추천 실행



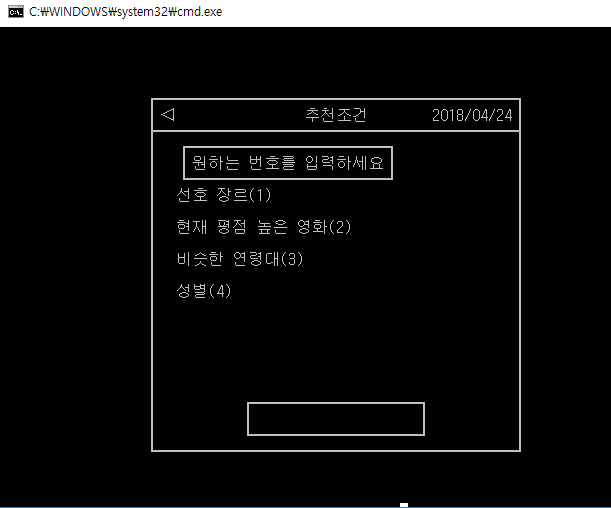
로그인



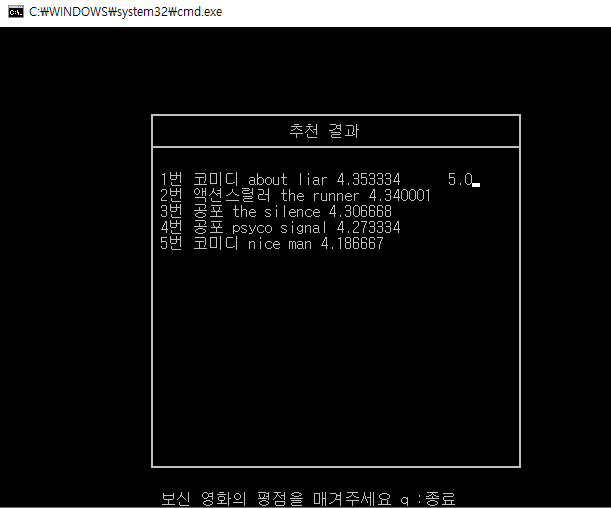
수동 추천 모드



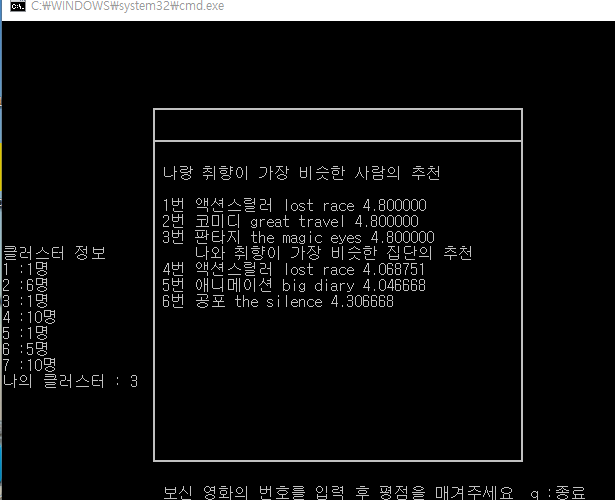
선호 장르 추천모드



전체 평점 추천



자동 모드 추천



1. 느낀점
   * 원동욱 :

이번 팀프로젝트 아이디어를 빅데이터와 머신러닝에 적합한 파이썬으로 구현하려 했다. 인강과 도서관에 있는 책들을 찾아 필요한 지식들을 쌓으며 관련 이론을 공부했다. 데이터 시각화에 뛰어나고 빅데이터,머신러닝에 특화된 파이썬의 통합개발툴인 아나콘다를 이용하려 했으나 현실적으로 시간이 너무 부족해서 제작하다가 중도 포기하고 급하게 C언어로 구현했다. 애당초에 현실적인 한계를 잘 가늠하고 받아들였으면 이런 사태(?)는 일어나지 않았을 텐데 아쉬움이 남는다. 그래도 개인적으로 정말 많이 공부 됬고, 이번 프로젝트를 통해서 많이 얻어가서 좋았다. 이런 팀프로젝트는 처음인데 나중에 캡스톤 설계할 때 많은 도움이 될 것 같다.

* + 이재성 :

팀 프로젝트를 하면서 평소에 프로그래밍 공부를 많이 해야겠다는 생각이 들었다. 1학년때도 프로그래밍 공부를 많이 안해서 기본적인 것도 할줄 모르는 상태였는데 이 상황에서 프로젝트를 하려니 많이 어려웠다. 특히 그동안 했던 과제들은 단순한 프로그램을 코딩하는 것이었는데 그것을 넘어서서 큰 규모의 프로그램을 만드는 것은 처음이어서 내용을 이해하는 것도 좀 힘들었다.

프로젝트에서 주로 인터페이스를 구성하는 부분을 맡았는데 이것을 진행하면서 그동안 보지 못한 새로운 함수들을 많이 접하게 되었고 이를 많이 사용하면서 공부가 많이 되었던 것 같다. 아직 실력이 많이 부족해서 프로젝트를 수행할 때도 좀 어려워했는데 앞으로 프로그래밍을 꾸준히 공부해서 다른 프로젝트를 할 때는 잘 해내고 싶다.

* + 이휘연 :

프로젝트를 진행하기 전까지 저는 ‘구조체가 굳이 필요가 있을까?’ 이런 생각을 했습니다. 여태 구조체를 쓰지 않아도 원하는 프로그램을 짤 수 있었기 때문입니다. 하지만 이번 프로젝트를 맡게 되면서 구조체의 필요성을 깨닫게 되었습니다. 구현하는 프로그램의 규모가 평소보다 몇 배로 커지니 데이터를 관리하는 게 굉장히 복잡하고 까다로웠습니다. 그리고 이 문제를 구조체를 적극 활용함으로써 이를 극복할 수 있었습니다. 프로젝터를 진행하면서 많은걸 배울 수 있었습니다.

* + 안시현 :

팀을 이루어 프로그래밍을 하면서, 평소보다 훨씬 큰 프로젝트를 만들어 보았고, 이를 통해 적절한 함수 사용의 필요성과 주석처리의 중요성을 깨닫게 되었다. c언어를 사용하였는데 머리 속에 떠오르는 대로 쉽게 구현되지 않아 힘든 점이 많았지만 많이 고민해보며 문제 해결능력을 기를 수 있었다. 이번 프로젝트를 통해 쌓은 경험을 기반으로 이번에 배운 파이썬이나 다음에 배울 언어들을 사용해서도 프로그램을 짜보고 싶다.