SE271 Fall 2020

Team Project Report

1. Title of our project

Will You Match Me? UGRP! (Matching program for UGRP Team Members)

2. List team members:

name	Email address	Student ID
고낙헌	skrgjsdl23@dgist.ac.kr	201911010
김주형	kimmold@dgist.ac.kr	201911031
이의정	dmlwjd317@dgist.ac.kr	201911131
한현영	hyhan@dgist.ac.kr	201911189

3. Motivation for our project

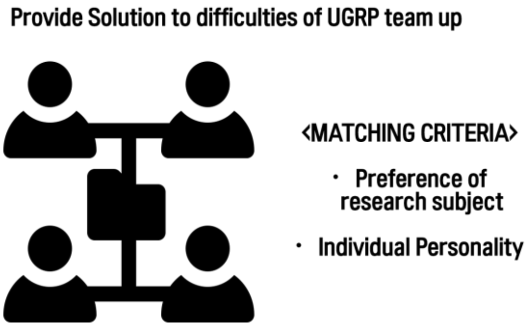

Provide Solution to difficulties of UGRP team up
<MATCHING CRITERIA>
· Preference of research subject
· Individual Personality

fig 1. motivation

As a student of DGIST, UGRP is essential. It is important to find a team that fits well before starting a full-fledged study by the UGRP. However, it is not easy to team up with students who have similar interests. To solve these difficulties, we decided to design a UGRP matching program. Simply, after entering student information such as major field, student id, and interested research field, the program prints list of students who are suitable as team members by comparing various factors. We aimed to help students form a well-matched team by considering not only the field of research or major, but also interpersonal factors.

4. Summarizing what we planned to do and what have done and haven't done.

Planned to do	Implementation State	Remarks
Sign-up, Sign-in	○	
Enter user information	○	Save to in-program database
Matching function based on Students' interests	○	Use Euclidean distance
Matching function based on Students' personality(MBTI)	○	
Print out students who fit	○	
Matching function that link user with professor(based on	○	

user’s interest)		
Print out list of professors	○	
Serving to multiple users using a Web-based database	△	Deploying in-program database

5. Describe the implementation design of your project,

a. Provide a class diagram which includes classes with their major member function and other functions.

The program is largely divided into three classes. A structure in which a parent class named Person is inherited by a Student and a Professor, and the Person class basically contains member variables such as Name, Gender, Mail, and Subject Array. In the case of a Student class, it inherits the member variables of the Person class and additionally has the ID, Password, MBTI, Self-Introduction, and Matching Score as the member variables. The Processor class inherits the Person class and has Office location and Major subject as member variables. Through inheritance, common variables were efficiently organized and utilized. In particular, the information received by instantiating the student class stores each object's information in a csv file.

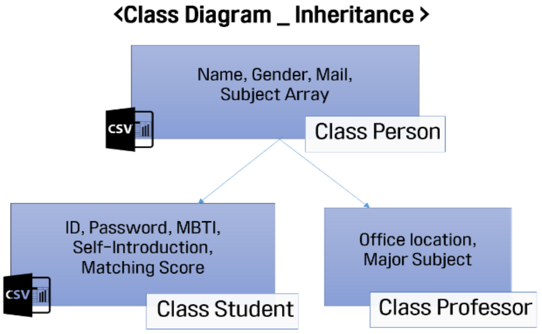
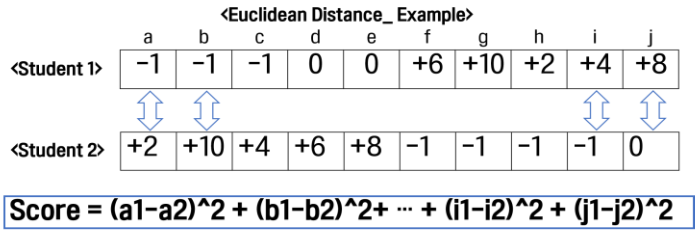

<Class Diagram _ Inheritance >
Name, Gender, Mail, Subject Array
csv
Class Person
ID, Password, MBTI, Self-Introduction, Matching Score
csv
Class Student
Office location, Major Subject
Class Professor

fig 2. Class Diagram

Through using hash function into sign up step, time complexity is changed O(n) to O(1). It helps to find specific student instance in csv file immediately. Hash function provide the inherent number for each student, it makes compile more effectively. It changes the process of finding student instances sequentially from the beginning to approach by specific index.



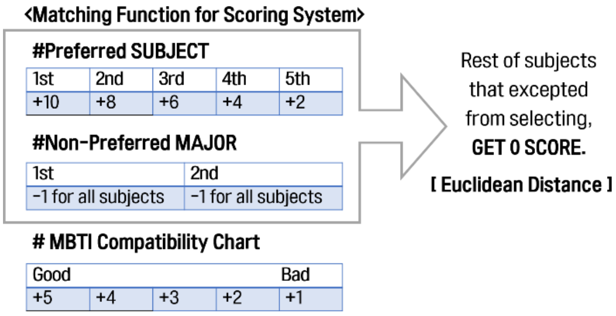
	a	b	c	d	e	f	g	h	i	j
<Student 1>	−1	−1	−1	0	0	+6	+10	+2	+4	+8
<Student 2>	+2	+10	+4	+6	+8	−1	−1	−1	−1	0

Score = (a1−a2)^2 + (b1−b2)^2+ ⋯ + (i1−i2)^2 + (j1−j2)^2

fig 3. Euclidean Distance

The method based on the matching algorithm of this program is Euclidean Distance. For Euclidean Distance, priority of preferred subjects can be considered and used as an efficient method for recommendation. Hamming Distance method provides more effective way than we've been trying to use before. There is an example that calculate the scores of students who have different interests based on their own score, as shown in the fig 3.

The way of scoring followed represented standard in fig 4. Ultimately reflected in the scores are preferred or non-preferred subjects, and MBTI. Based on this, the scores are calculated and the students are given a list of suitable students. Through using Euclidean Distance, we should realize that between two students who have similar interests get low score from each other. It’s because their score distribution will reduce the difference of scores.


<Matching Function for Scoring System>
#Preferred SUBJECT

1st	2nd	3rd	4th	5th
+10	+8	+6	+4	+2

1st	2nd
−1 for all subjects	−1 for all subjects

Rest of subjects that excepted from selecting, GET 0 SCORE.

[Euclidean Distance]

# MBTI Compatibility Chart

Good				Bad
+5	+4	+3	+2	+1

fig 4. Matching Algorithm

b. Provide details about what we are each in charge of.

name	Contribution(%)	Main Work Performance
고낙헌	25	Management given information from users with csv file, Matching function about preference of subjects, Presentation
김주형	25	Sign up & login code, Hash functions, Professor recommendation, Testing program with making sample data
이의정	25	Data collect and processing, Class overall management, Presentation and Report Preparation
한현영	25	Matching function about personality with MBTI, Overall project code management, Arrange code effectively

There are some explanation about main work performance from each team members;

Nakheon Go: I was in charge of creating an array of objects from the csv file during the initial execution of the program, and writing a matching function that scores according to the subject preferences of other students based on the currently logged in visitor.

Juhyeong Kim: I wrote the entire sign up & login code and designed the hash function. Also, I designed a student matching algorithm, designed and implemented a function that show recommended professors, and made students & professors sample data and conducted tests to see if the program was working well.

Uijeong Lee: I collected various data which is needed from the programs setting, and processed the data to make it easier to handle in sorting. I manage the relationships of classes and functions such as inheritance, connection. Also, I prepared final presentation with total understanding of our project.

Hyunyoung Han: I've dataized the MBTI (myers-briggs type indicator), a popular personality indicator for young people these days. Based on this, I designed a function that matches the personality of students. I also took charge of overall code version management (git) of the project, enabling efficient project progress.

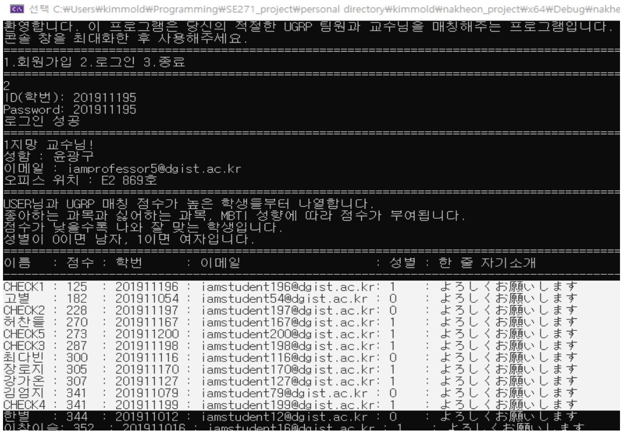
6. Explain how to compile our code and how to test.

We have verified that the program is working properly by executing a series of processes we have devised from the user's perspective. During the sign-up process, the user's student number, password, name, gender, school email, MBTI, user's preference of subjects or majors, and self-introduction were entered. Subsequently, we checked these information was stored in the database. It was necessary to verify that the matching algorithm based on Euclidean distance measuring technique worked well in practice, and that the results from the entered information were consistent with those from the test process. Since then, we have confirmed that the process of entering and storing the information (name, email, and office location) of professors we randomly generated. We also checked professor matching system which can suggest the professor who is in the field of user's interest is working properly.

7. Results section

a. Describe our experiment setup and metrics of success.

We checked our project validity by using samples of 200 students and 100 professors._Through setting random students with interests in various majors, we predicted students who will be at the top of the recommended list. And check if they fit the program's result list. To measure the effectiveness of the program more accurately, we used the checking data which named “CHECK” as a practical case at set up step of our program. We predicted that our program will recommend sample students which was named “CHECK 1, 2, 3, 4, 5” to student who one’s id is 201911195. We set their propensity similar, we could prove the significance of our program by showing that five samples exist at the top of the list. And we were able to confirm that checking data was included in the top 11 of recommendation list, as shown in the fig 5. These results can prove the validity of our program.



```
선택 C:₩Users₩kimmold₩Programming₩SE271_project₩personal directory₩kimmold₩nakheon_project₩x64₩Debug₩nakhe
환영합니다. 이 프로그램은 당신의 적절한 UGRP 팀원과 교수님을 매칭해주는 프로그램입니다.
콘솔 창을 최대화한 후 사용해주세요.
==========================================================================================
1.회원가입 2.로그인 3.종료
==========================================================================================
2
ID(학번): 201911195
Password: 201911195
로그인 성공
==========================================================================================
1지망 교수님!
성함 : 윤광구
이메일 : iamprofessor5@dgist.ac.kr
오피스 위치 : E2 869호
==========================================================================================
USER님과 UGRP 매칭 점수가 높은 학생들부터 나열합니다.
좋아하는 과목과 싫어하는 과목, MBTI 성향에 따라 점수가 부여됩니다.
점수가 낮을수록 나와 잘 맞는 학생입니다.
성별이 0이면 남자, 1이면 여자입니다.
==========================================================================================
이름   : 점수 : 학번      : 이메일                    : 성별 : 한 줄 자기소개
==========================================================================================
CHECK1 : 125  : 201911196 : iamstudent196@dgist.ac.kr: 1    : よろしくお願いします
고별   : 182  : 201911054 : iamstudent54@dgist.ac.kr : 0    : よろしくお願いします
CHECK2 : 228  : 201911197 : iamstudent197@dgist.ac.kr: 0    : よろしくお願いします
허찬들 : 270  : 201911167 : iamstudent167@dgist.ac.kr: 1    : よろしくお願いします
CHECK5 : 273  : 201911200 : iamstudent200@dgist.ac.kr: 1    : よろしくお願いします
CHECK3 : 287  : 201911198 : iamstudent198@dgist.ac.kr: 1    : よろしくお願いします
최다빈 : 300  : 201911116 : iamstudent116@dgist.ac.kr: 0    : よろしくお願いします
장로지 : 305  : 201911170 : iamstudent170@dgist.ac.kr: 1    : よろしくお願いします
강가온 : 307  : 201911127 : iamstudent127@dgist.ac.kr: 1    : よろしくお願いします
김엄지 : 341  : 201911079 : iamstudent79@dgist.ac.kr : 0    : よろしくお願いします
CHECK4 : 341  : 201911199 : iamstudent199@dgist.ac.kr: 1    : よろしくお願いします
한별   : 344  : 201911012 : iamstudent12@dgist.ac.kr : 0    : よろしくお願いします
이찬이솔: 352  : 201911016 : iamstudent16@dgist.ac.kr : 1    : よろしくお願いします
```

Fig 5. Recommendation List

b. Present results with discussion.

Although we could confirm that the previously set virtual models are matched with a fairly high probability, if considering the various situations that actually exist, we think there are a few things that need to be supplemented to produce more valid list of team member matching. In forming a UGRP team, the relationship between team members is as important as the subject of interest. Considering that a simple personality test called MBTI may not be really accurate, we discussed that it is necessary to utilize a slightly more complex and diverse approach in the matching function associated with propensity. Furthermore, assuming that make a team of convergence topic, students who wants students in different field may not be able to make full use of our programs. This can be solved by dividing the optional parts of the program into more detail. It is expected that the revision of additional settings will help more various students.

8. Conclusion of our work and what we learned.

Unlike Python, which used to be used a lot, I felt that programming using C++ was more difficult. Also, compared with Python, which was easy to program by modularizing it based on function, even if C++ is the same function, there were some difficulties in modularizing even though it was a little different for each individual code. Through team project with C++, we felt the importance of sophisticated programming. From the basic grammar of C++ to the challenging contents learned through the lecture, we could see that each of the roles played in creating a single program and connected to each other by organizing and structuring multiple classes. In addition, we could experience the convenience of using STL and the efficient connection of class due to the utilization of Inheritance. Furthermore, in the process of realizing common goals, we realized the importance of communication and collaboration.

9. Link of the source code repository

https://github.com/h0han/SE271_project/tree/master/code/SE271_project