Fantasy league legend

[文档副标题]

215959

Table of content

Table of content

[1 introduction 2](#_Toc69055106)

[2.background research 4](#_Toc69055107)

[The team Elo rating system 4](#_Toc69055108)

[Fibonacci sequence [4] 5](#_Toc69055109)

[Memorization [5] 6](#_Toc69055110)

[Per(Player Efficiency Rating) [6] 6](#_Toc69055111)

[3.professional considerations 7](#_Toc69055112)

[4.requirement analysis 9](#_Toc69055113)

[Functional requirements 9](#_Toc69055114)

[Non-functional requirements 9](#_Toc69055115)

[5.system design 10](#_Toc69055116)

[5.1 learning objectives 10](#_Toc69055117)

[Modules 10](#_Toc69055118)

[Modelling team chemistry 11](#_Toc69055119)

[The function to calculate the strength of groups of players 13](#_Toc69055120)

[The function to calculate the team chemistry 14](#_Toc69055121)

[Data memo function 16](#_Toc69055122)

[Tkinter library [8] 16](#_Toc69055123)

[UI design 17](#_Toc69055124)

[Game rule 18](#_Toc69055125)

[6.implementation 20](#_Toc69055126)

[Explore the suitable model for data structure 20](#_Toc69055127)

[The rating implement method change 23](#_Toc69055128)

[Implementation issue 23](#_Toc69055129)

[7.testing 25](#_Toc69055130)

[Unit test 25](#_Toc69055131)

[Integration test 26](#_Toc69055132)

[System test 26](#_Toc69055133)

[7.evaluation 27](#_Toc69055134)

[8.conclusion 30](#_Toc69055135)

[Future work 30](#_Toc69055136)

[9.appendix 31](#_Toc69055137)

# 1 introduction

As more and more sports move towards professionalism, sports analytics is gaining more and more attention, especially after the great success of baseball and basketball, and now terms such as "magic ball theory" [1]and "penalty kicks" have gained considerable fame in both fan and AI circles. At the same time, "fantasy leagues" have become very common. A fantasy league tournament is a model in which players create their teams as professional basketball managers.

The goal of this report is to create a card game based on the fantasy league model. Two players will choose their respective players as professional basketball managers and the system will score the players individually and by the bonds between them, with the team with the highest score winning. Another objective is to clarify the difference between the scoring system and other scoring systems. The main difference is that the scoring system takes into account the overall relationship between the players and the impact of the level of understanding on the overall team score. This will be achieved through memoisation. This report will demonstrate in more detail the program requirements, practices, implementation, and evaluation of this project.

Team chemistry in NBA[2]

Most teams have great surface strengths, but when they don't produce results, the team says 'wait, give me time, all we're missing is chemistry. It could be concluded as two main part:

A sense of community - when every player on the team finds their place, plays to their strengths and is willing to contribute to their team.

The level of understanding - players understand each others’ strengths and weaknesses and can work well together e.g. when one player is free to shoot, his teammate would pass the ball to him in time.

In this project, the team chemistry will be considered as a value, and the team rating system will calculate this value. The value will be taken into account in the final team score.

This document describes the project in eight sections:

Research: this section details the preliminary research carried out before starting to design the solution. This includes an outline for some of the scoring systems already in place.

Professional considerations: this section includes how the project will take into account the public interest, professional competence and integrity, and accountability to relevant authorities.

Requirement analysis: this section outlines the requirements of the project to determine the size and focus of the anticipated final project. This includes the functional and non-functional requirements of the final system.

Design: This section describes the overall game and the design process for the rating system. The learning objectives of the project are discussed, as well as the functionality that the rating system will have. Furthermore, the selection and comparison of models for player data structures will also be discussed in this section. Finally, the creation of the user interface and the explanation of the rules of the game are explained in detail.

Implementation: this describes all the problems encountered during the implementation of the project. It outlines the general order in which the problems occurred and how they were dealt with.

Testing: the program is tested using unit tests, integration tests, and system tests.

Evaluation: It includes a comparison of this scoring system with other scoring systems and an assessment of the accuracy of the project using R-values. It also includes a self-reflection on the overall results of the project and highlights any other outstanding issues with the system.

Conclusion: which summarizes all the elements covered in this report. And discusses further work that could be done on this project in the future, reflecting on the outstanding requirements and explaining how they could be implemented.

# 2.background research

## The team Elo rating system

Elo is a simple measure of strength based on game-by-game results; it is used to calculate the relative skill level of players in a zero-sum game. A player's Elo score is expressed as a number, which may vary depending on the outcome of the rated game played. At the end of each game, the winning player receives points from the number of points lost. The difference between the ratings of the winners and losers determines the total number of points gained or lost after the game. If the player with the higher rating wins, the player with the lower rating will only gain some rating points. However, if the lower-rated player gets a losing victory, many rating points will be transferred. In the case of a tie, the lower-rated player will also gain some points from the higher-rated player. This means that the rating system is self-correcting. In the long run, players who are rated too low or too high should accordingly perform better or worse than the rating system expects and thus gain or lose rating points before the rating reflects their true strength.

So, in this program, Elo ratings depend only on the final score of each game and the location of the game (home-field advantage). They include both regular season and playoff games.



-Figure1. the graph about Los Angeles Lakers team Elo[3]

This graph shows the change in Elo scores for the Los Angeles Lakers team from 1950 to the present day, with the scores at each stage and the slope giving a good indication of the overall strength of the team at that time. To work out who would win and who would lose between the two teams, you first have to calculate the difference in points between them. To map this difference into probabilities, Elo is often modelled by the sigmoid function (logistic). Specifically, it is a function that maps (−∞,+∞)→[0,1], where σ(0)=0.5. Elo rating is often used by the sigmoid function (logistic) for statistics and research specifically, it is a function that maps the difference between Elo rating. The sigmoid function is used here to get a better look at the data comparison of a zero-sum game.

****

**-** Figure **2. sigmoid function diagram**

## Fibonacci sequence [4]

Fibonacci sequence is a sequence in which each number is the sum of the first two numbers. The number at a specific position in the Fibonacci sequence can be obtained using recursive methods.

The pseudocode for fib can be conclude as:

Define Fib(n)

If n = 0

Return 0

Else

Return Fib(n-1)+Fib(n-2)

****

**-** Figure **3.the fib diagram**

## Memorization [5]

**In computing, memoisation is an optimisation technique that focuses on speeding up computer programs by storing the results of expensive function calls and returning the cached results when the same input occurs again.** **memoization is an optimization of the recursion function. Unlike the fib function, it sets up an object called a cache, which stores all the return values we have obtained, and when the function is executed it will check whether the cache has the required data, and if it does, it can be called directly, if not, the next calculation will be done, which can effectively reduce the number of function calls.**

****

## Per(Player Efficiency Rating) [6]

The Player Efficiency Rating is a per-minute rating developed by John Hollinger[6]. In his word. “The PER sums up all a player’s positive accomplishments, subtract the negative accomplishments, and returns a per-minute rating of a player’s performance.”

PER Benefit;:

It will show the personal average ability especially the offense part due to the huge weight in the algorithm.

PER Limitation:

But it is irrational to create the team due to the PER , we cannot just choose five players who are in the top PER List. What is more, the question also pointed by John that the PER always showed that the person who is good at the offense part but not the defense part.

# 3.professional considerations

3.1. BCS Code of Conduct

This is the BCS Code]of Conduct[4] which relate to my project:

1.1 have due regard for public health, privacy, security and wellbeing of others and the environment;

The game is a team rating game, it will store the data about the name for the team, the member of the player’s ‘fantasy team’ and the final rating for their team when they finished their selection. Also the statistical data for the NBA players could be found through the internet.

1.2have due regard for the legitimate rights of third parties;  
Due to the free charge of this game , there will have fewer obligation for the third part.

1.3  conduct your professional activities without discrimination on the grounds of sex, sexual orientation, marital status, nationality, color, race, ethnic origin, religion, age or disability, or of any other condition or requirement;

The game designed for all age group without any discrimination, the older and the children all will be able to enjoy this game, In the process of game the children should be supported by their teacher or parents, which could help them expand their team manager ability to create their own team.

1.4  promote equal access to the benefits of IT and seek to promote the inclusion of all sectors in society wherever opportunities arise.  
This game is a free charge game which means the it will promote the benefit of IT equally. It will not consider anything about economic background.

2.5. respect and value alternative viewpoints and seek, accept and offer honest criticisms of work;

I will show my biggest respect of all of the feedbacks from the user, rather than ignoring these.

4.6 encourage and support fellow members in their professional development.  
If any of my fellows get any question in their professional development, I will do my best to help and support them.

2.2. Ethical Issues[5]

This project is a specific type of game, so it is necessary to consider the ethical approval in order to make sure it will not damage social healthy. This project is only a rating game do not include any feature about risk, for the participants will have a try on this rating game and could provide the feedback for this project if they want. Due to this game do not have age limitation, this game will make sure there is nothing harmful to the health of minors . The

6 basic rule of the game and the rating of user’s fantasy team will be show in the interface, and the information about their fantasy team only be used after the participants sign the consent form. All of the feedback from the participant will be stored in personal laptop which protected by the password

# 4.requirement analysis

The main use of the scoring system is to allow users to choose as well as match their 'fantasy league' and get feedback. This part will consider an rational rating system, which the functional requirement and non-functional requirement will be outline.

## Functional requirements

Here are the List for Functional requirements:

* The project shall be playable by two player on one computer
* The user will be able to input their team name
* The system should provide the user the order of pick
* Users will be able to choose the player from the random set
* Users will be able to see the information and image of the player
* The system should rate the team fairly
* The game system should determine the winner
* The game system should remind the user about total cost and the rest of the cost after they chose a NBA player
* The game should show the 20 random player’s information clearly(include their name, and their cost)

## Non-functional requirements

Here are the List of Non-Functional Requirements

* The project will be developed using Python and Python Graphical User Interface
* The User could go through the user interface easily
* The project will compatible on desktop device

# 5.system design

## 5.1 learning objectives

* Explore and figure out the most suitable model for players data structure
* Strength python knowledge
* Learn to build the User Interface by Python Tkinter library
* Explore how to calculate the value of the relationship and how the player relation will impact the final rating
* Come up with a reasonable function to rate the player group

The main learning objectives of this project were to learn and find the most suitable model for the player data structure and to build a team scoring system by finding the core of the relational ability values between players. As the final form of the project will be a game, it was necessary to find a suitable framework library. The last and most important part of the project is to find a suitable scoring method for the players. I found that memoization is the easiest and most efficient way to Implement and integrate the results rating of each team, so understanding how to use memoization became another learning objective. The last learning objective was to learn more about the use of the python library and to increase proficiency in python. The choice of programming language for this project was simple as python is simple and easy to understand with many libraries for data analysis and statistics.

### Modules



* data flow diagram

The picture above shows how the whole system works and how each part interacts with each other，As this project would eventually be presented as a game. Firstly the overall flow of the rating system, the player information, and the team Elo rating are the main data used to calculate the final team rating. Player rating and team Elo rating are used to calculate the team chemistry value, which is the main goal of the project. The resulting team chemistry values are stored in the Data memo, which provides some information to the Strength function.

The UI is indispensable as the project will eventually be presented as a game, the overall presentation of the UI is dependent on the rules of the game, the player will select 5 players to form a team through the UI and receive a score feedback, when each of the two players has selected a team the selection data will be sent to the Strength function. Then Strength function will calculate the final team scores from the data in the Data memo.

### Modelling team chemistry

The model for team chemistry has evolved through three stages.

Initially, the model only considered the relationship between individual team members.



- Figure4. the player chemistry relation display

As shown in the picture, the lines between each player represent their relationship and the value on each line represents the level of chemistry between the players, the higher the value the higher the chemistry between the players, in other words the higher the value the better the relationship between the players, which makes it easier for the players to produce good results. And this is where the main focus of this project will be.

The previous model was refined to also consider the relationship between groups of people or individuals



- Figure5. the relationship graph between groups

Unlike the first type, the second type is an evolution of the first type, which describe the relationship between groups and individuals, as expressed in the diagram, where player A and player B are a combination, and player C is associated with this combination in some way, and the addition of player C will potentially make the combination stronger

Because the relationship between large groups within an encompassing group will generate many chemistry values without adding extra information, the model was changed to evaluate the strength of each subset of the players.



- Figure6. the graph show the final type

The final team chemistry design will show the relationship between groups. As the diagram above shows, player A is a group with player B, and player A is a group with player E. When group AB is combined with group AE, the effect this has on the group ABE is modelled in the value of the group ABE.

### The function to calculate the strength of groups of players

The function to calculate the strength of groups of players is attempt to use the pure recursion, which is mentioned in the background research. We can draw on fib recursion to design our strength function.



Figure7. The player rating implement by fib method

As the diagram above shows, the objective is to try to obtain the rating of a team of three players (Player1,2,3). In order to calculate the rating of this team, firstly we need to obtain the individual values of each player, and secondly we need to take into account the possibility that each two of the three players will be combined together, as shown in the second row of the diagram above. In simple terms, the strength of the second row is the strength of the third row plus k, e.g. (player 1,2) = strength (player 1) + strength (player 2) + k, and so on. The formula given in formula 1 is the recursion made by this method, so that the final team rating can be derived.

Here the Strength function is used to describe the rating of a player or team, S is the total number of the player and t is the player which is a number of T.

formula 1

In the function, the recursion will consider and go through the powerset of the groups of player. If we want to get the overall strength values of the group, exploring the strength values between players and players is an indispensable step

### The function to calculate the team chemistry

In the background research, we found the Elo rating could be a good way to represent the strength of each team. I searched the Elo ratings of the 14 teams and selected the highest Elo rating when the players were all in this team to calculating the team chemistry value because 14 outstanding teams were selected for this project.



-team Elo histogram

As the graph shows, most of the teams have an Elo score between 1750 and 1800, which ensures that these teams are very strong, as the initial Elo score is 1500. For the purpose of this project, we need to find the team chemistry values of the best combinations from these 14 teams. Each of the 14 teams will have several role players in addition to the famous combinations. The chart below summarizes the information on all players, including the famous combinations, which will be used to calculate the chemistry value.



-the player rating information histogram

The mean for player rating is 79.188.

According to the background research, Elo rating is often used by the sigmoid function (logistic) for statistics and research, so we can conclude that Elo ratings are a logistic distribution. However, it is hard to find a distribution that fits the player's rating. For simplicity and ease in computations, I assume that both player ratings and ELO ratings can be normally distributed.

Here is the formula to calculate the team chemistry value:

In this project both player rating and Team Elo are considered to follow normal distribution, so we could find a specific ratio by the mean of team Elo/ the mean of player rating. This formula uses this ratio value and all the information about the team at the time to calculate the team chemistry value of the best combination within that team.

### Data memo function

As mentioned in the module above, class set for powerset and team chemistry are used to integrate and calculate some information which will be used in the class Strength. class team chemistry is used to calculate the team relation value k.

As mentioned earlier, the Strength function is implemented and calculates team scores using memoization. The main feature of memoization is that there is a cache object, which is used to store some of the computed data. While the recursion is in progress, the data can be used directly, if the required data can be found directly in the cache. In this project, the data memo function can be understood as a cache function. the parameter of the data memo function is a set. This set is obtained using the powerset function, which gives all the possibilities for grouping players. The data memo function uses the team chemistry value to score and store each grouping.

Here is the pseudocode of Data memo:

Def Data\_memo(set):

New Memo={ }

For set in set:

If len(set)==1:

New Memo[set]= player rating

If len(set)==2:

New memo[set] = team chemistry value + team

If len(set)==3:

New memo[set] = team chemistry value + team

Return New Memo

The powerset size is set to a maximum of three, which means that the maximum possible combination is three players per set, as shown above. The reasons for this are explained in part 6.

### Tkinter library [8]

For methods of making UIs, Tkinter (reference) is the most common library for developing GUIs (graphical user interfaces) in Python. It is a standard Python interface that provides a toolkit for making UIs. The main language used for this project is python, and Tkinter is more accessible and efficient, so finally, Tkinter was chosen to build the user interface

### UI design



-the front page for UI

This is the starting page for the game, both of the players could type the name they prefer to their team, and start the game, the question mark at the bottom is used to show the rule and gameplay of this game.



-Main UI diagram

The final user interface has been partially simplified and compared to the initial prototype it shows less detailed information about the players, such as their abilities and base positions. The overall layout of the UI depends mainly on the rules and gameplay of the game. Before the game start, the player will give the name of their team and given the order by rolling the dice. The player could choose five players which would be displayed with images and combine their fantasy league team. When the player has selected the final player, they will be taken to the final scoring page, which will show the final winner and the scoring of their respective team. From this page, players will be informed of the reasons for the high and low ratings. Which is shown below:



-the End page

### Game rule

For the rules of the game, the order of the players is determined by rolling the dice. To ensure the competitiveness of the game, the players are selected in a fixed order, but the players are free to mix and match and to use the order rationally, for example, to break up the opponents' upcoming combinations. The order in which players are selected is shown in figure xxx. To add interest to the game, we set and filter the players in the player pool in different tiers, all players with a rating of 80 or more will be selected as data, while the final reality is 20 players selected randomly from the data, and the players are divided into 5 tiers, tier 1 players with a rating of 96-100, tier 2 players with a rating of 92-96, and Tier 3 players are rated 88-92, tier 4 players are rated 84-88, tier 5 players are rated 80-84. Different salaries are set for each tier, for example, tier 1 players are paid 5 coins, tier 2 players are paid 4 coins, and so on. To make it more challenging to build a team, we give each player 19 coins, In general, each player will choose two tier 1 NBA players, so they will both be left with 9 coins, which will test the player's ability to match and make trade-offs. For example, if player choose two 4 coins cards, then he can only choose one more 1 coins card. and the player has to use a limited number of coins to build a team.



Figure 6 -the order for the players to choose from a pool of basketball players

# 6.implementation

## Explore the suitable model for data structure

For the basic setting of the data, the main objective of this project was to explore and calculate the tacit values of the relationships between players (team chemistry). Based on the background research, I selected 14 famous combinations and compiled their team and teammate data for this project.

The initial stage of the model is considering which method to use to present the graph

-The player relation model

This image shows how the relationships between players can be represented by the adjacency list[9].

the adjacency list is a chain structure of the graph it will have the vector, states, and a line connecting the state. In this diagram, Player A connects to Player B, which means Player A has a relationship with Player B. And the result for the adjacency list is easy to represent Here is the graph to show the relation result:



-adjacency list to show the relationship

However, there are other ways to represent the relation between players.

Adjacency list can also be represented as arrays:



-The adjacency array

In this table, the horizontal rows represent the number of connections of that point and the vertical columns represent each unit, with column 0 indicating how many connections there are.

The graph is used to be design as the model to show the relationship combine between each player. A Graph is a non-linear data structure consisting of nodes and edges. The nodes are sometimes also referred to as vertices and the edges are lines or arcs that connect any two nodes in the graph. And it is consists of a finite set of vertices(or nodes) and set of Edges which connect a pair of nodes .The Adjacency Matrix is the main factor which give us a way to represent our player relation graph in an efficient and structured procedure. It is easy to represent nodes and edges by creating a matrix table as the graph show below:



-adjacency matrix graph show the value

Adjacency list memory usage depends more on the number of edges (rather than the number of nodes), if the data is sparse, using an adjacency list will save a lot of space compared to an adjacency matrix, but the data in this project is more dense, so the adjacency matrix is the better choice. In addition, the adjacency matrix can be used to find and check the values of specific edges more quickly, which is important for getting the team chemistry values.

Finally, the power set model is set up, which not only identifies relationships between individuals and groups but also between groups and groups



-adjacency matrix graph show the value between power set and power set

In the final stage, both horizontal and vertical nodes will be power sets, which represent the possibility of all player combinations and the function represented can be shown as P(p) \* P(p) = Graph. And edges are used in this project to show the relationship between each power group, if there is no relationship between the groups then the edge will be entered as null, otherwise, the actual value will be entered and saved in the table to show that a relationship does exist between the selected power groups.

### The rating implement method change

The project initially used pure recursion to implement the final scoring of the integrated teams, but during the run, it was found that the program took a long time to run using pure recursion. However, during the run, it was found that the program took a long time to run using pure recursion, and when testing with larger data, it was found that there was a Stack overflow situation.

Here is the function of Strength method.

The pseudocode code for this function by using recursion is shown below, As it is rely on the way of recursion, it is necessary to set up the base case, so for this project’s pure recursion is if the number of set one, it will straight return that player’s rating.

Define Strength(set)

If num == 1

Return Rating(set[0])

Else

strength = 0;

For each subset in itertools.combination(set, len(set)-1)

Strength += Strength(subset)

Strength / (len(set))

Return Strength()

It will cause stack overflow condition, which is the program uses more memory space than is available on the call stack, for example like the graph to calculate the overall rating of (player1,player2,player3). The algorithm will calculate the rating of (player1,player2) and (player1,player3) and (player2,player3) then combine their rating to get the rating of three player, if there are request to rating a player of 4 group it will do a recursion until find all factor it need to calculate which will be explained in detail in the implement part, In this case, the recursion will do a lot of redundant work and have a low efficiency. So I try to find the more efficient way to solve the calculation (improve the fib recursion), which is the Memoization, the basic idea for memoization, is attempt to save all of the result that obtain from the recursion. And take out the result directly when we use it to calculate further information. Therefore, an array is defined to store the calculated data, and then query from the array when needed, which will save unnecessary calculations and improve the efficiency of the program.



-how to implement the memoization in this project

The data in the green part of the image can be used without calculation, because memoization defines a cache to store the calculated data, which can be used if the same data is needed again during a loop, which is not possible with pure recursion.

Here is the pseudocode for Memoization:

Define Strength(set)

Memo = keys:powerset5 -> values: zeros(len(powerset5(players)))

If num == 1

Memo[set] = Rating(set[0])

Return Rating(set[0])

Else

If (Memo[set] == 0)

strength = 0;

For each subset in itertools.combination(set, len(set)-1)

strength += Strength(subset)

strength / (len(set))

Memo[set] = strength

Return Memo[set]

In the pseudocode, the function will always check the Set and do the recursion which will always check all of the powerset of len (set), until it find the personal rating, however, as the method mentioned before, the formula to calculate a group rating is Strength(playerA,B)= Strength(playerA)+Strength(playerB)+K, K is the major factor influence the total rating of a group combine.it is a process in which new players and previous players go from strange to familiar, as well as the formation of team play and mutual cooperation. So if the two player have a good relationship their combine K will be a large number, vice versa.

### Implementation issue

During the development of the project some problems were encountered, mainly due to the availability of player data. The initial set-up was to use the most famous combinations of players, but the rating system was built to require more data, so the data size and entry method was changed.

The data was initially set to be small, so it was entered manually, but team Elo was used to calculate the team chemistry value when the project was underway, as the team Elo rating was based on the strength of the team based on everyone in the team, but only the team Elo rating was used to This revealed that the data was not comprehensive enough and that it would be very difficult to add more information manually. This revealed that the data was not comprehensive enough and more information had to be added manually. Therefore, we chose to expand the overall data by searching for all players in the 14 teams and adding them to the csv, using the csv to store and modify the data.

Another problem arose during the testing of the Data memo when several teams ended up with values that differed significantly from the true value, and it was thought that there might be a problem with the application of the algorithm. However, after doing the most basic calculation of the average player ability values, the problem was discovered. As the project selected 14 teams with a good mix of players at different times, some players were present in several teams at the same time, and the final ability value of the player depended on the last team entered, which led to errors in the scores of several teams.

When designing the relationships between players, the final choice was to use the powerset to find the relationship values between players. Initially the maximum size of the powerset was set to 4.

As the team combination is four, we will get millions of possibilities, which will seriously affect the running time of the program, and as a large part of the data used is from combinations of two to three players, it was decided to reduce the maximum size to 3.

# 7.testing

## Unit test

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Module | Normal Value | | Extreme Value | | Anomalous Value | |
| Value | Test result | Value | Test result | Value | Test result |
| Team Elo  Get team Elo() | ‘Utah Jazz’ | 1764 | - | - | ‘Good’ | null |
| Team Elo Get team Average() | ‘ Nuggets’ | 81.375 | ’14 team’ | Error | ‘Not team’ | Error |
| Player  Get\_wholeGP | ‘Chris Bosh’ | Heat | - | - | ‘other people’ | Error |
| Player  Whole\_rating | ‘Ray Allen’ | 88 | - | - | ‘Hello’ | Can’t find |
| Set for Powerset  Powerset() | Player List | A list include all possible powerset for player | A huge Player List | Stack over  flow | [1,2,3,4] | List all possible powerset for the number in the List. |
| Team\_infor() | 'Utah Jazz', 'Karl Malone', 'John Stockton', 'Jeff Hornacek' | 93.2124 | - | - | 'Utah Jazz', 'A', 'B', 'C' | Error |
| Set for Powerset  Data memo() | Player powerser List | Rating the score for each powerset | A huge Player List | Stack over flow |  |  |
| Strength() | Karl Malone', 'John Stockton | 104.6507 | ’30 player’ | Stack overflow | ‘nobody’ | Value Error: r must be non-negative |
| Strength easy() | ‘Heat’ | 79.0687 | - | - | ‘Kobe Bryant’ | Error |

## Integration test

|  |  |
| --- | --- |
| Integration | condition |
| Powerset-Data memo  Strength – data memo | A cache memo set as expected  Strength function can get the value from cache data |
| Data memo- Team Infor | The value calculate from team infor can be stored into Data memo |
| Team\_infor-whole rating | In team infor function the player rating can be obtained by using whole rating function |
| Team infor – Team elo rating | In team infor function each team Elo rating can be got from Team elo function |

## System test

|  |  |
| --- | --- |
| Case | Condition |
| Player select teams normally | A wins as expected |
| Players choose players worth more than their budget | Can not choose, force to choose another one |
| Players enter the team name | The team name will be shown on the screen |
|  |  |
|  |  |

# 7.evaluation

This section will contain self-reflection and some quantitative analysis of the calculated scores to assess the accuracy of the scores. In addition to this, there will be a comparison of this scoring system with other scoring systems that exist. These were chosen over user testing because the project is more focused on the back-end functionality of the system rather than the end-user experience.

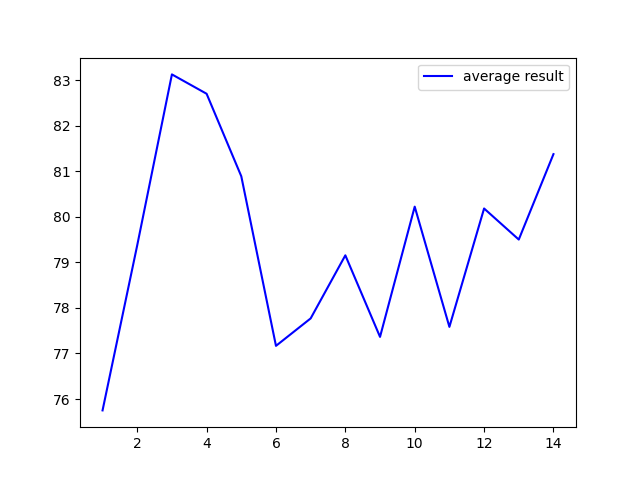
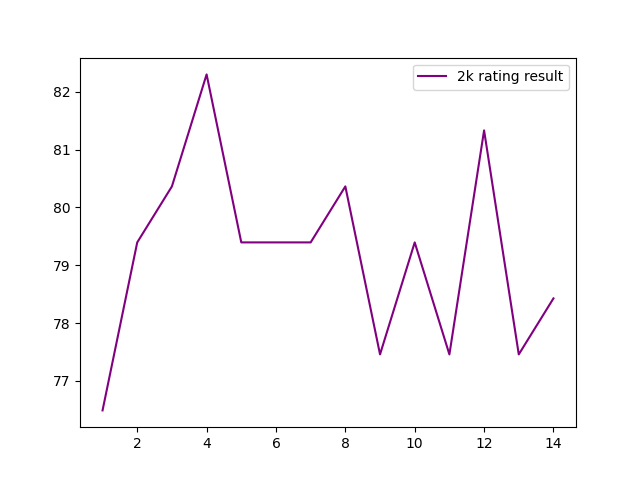


-confusion matrix graph

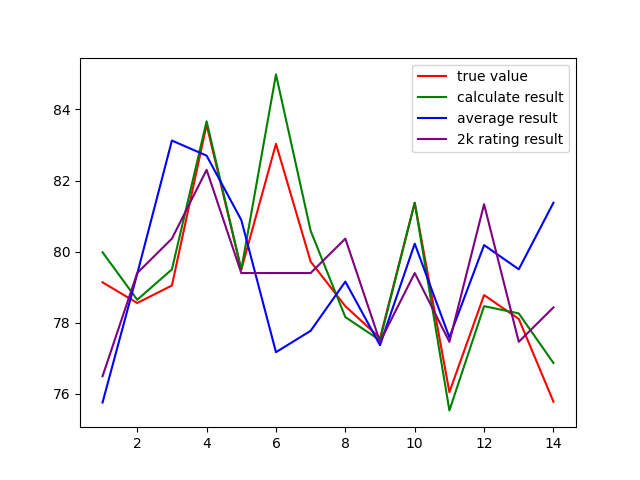
Above is the confusion matrix graph, The horizontal and vertical rows show the real data and the calculated data respectively, the middle row shows the difference between the two data and is colored to represent the size of the difference, with lighter colors indicating smaller differences and darker colors indicating larger differences

** **

-True value -calculate value

-rating calculate from average -the rating from 2krating website



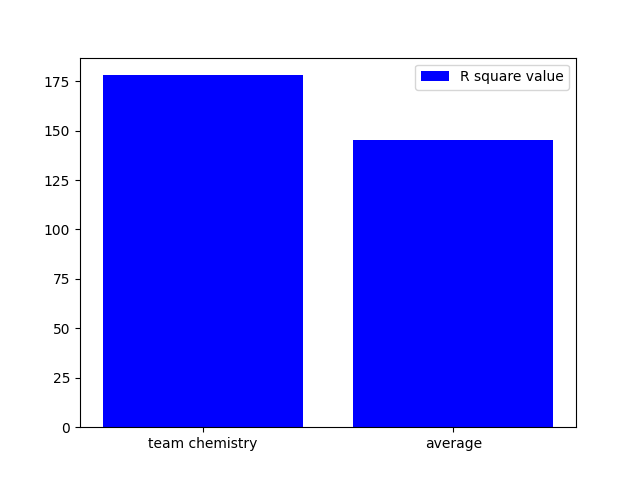
-Figure 7

Four different results are shown in figure 7, the true value is set for each team Elo rating divide by the ratio(average Elo/average Player rating),and the 2k rating is that the rating record from the other rating system ,according to the graph we could find most of the value are similar to each other, except the team 6 and team 14,and our calculate value follow a similar path to the true value.

****

-r square value bar chart

The r square value is used to calculate the accuracy of our project, as the graph shown if the value of r square is very low that means the result is more accurate. However, the data of our result is to close to the true data, so I assume that maybe there cause a overfitting condition. In order to check whether is it overfitting the 2krating value are set to the true value, and this is the diagram show below:



-the R square value graph base on 2krating

The graph indicate that, our result’s R square value is a little bit higher than the average data.

# 8.conclusion

In conclusion, this report shows the overall process of a simple game based on a scoring system. It shows what this scoring system entails through background research, it covers the design methods used for development as well as the design of the UI and the rules of the game. The implementation phase explains some of the model or methodological changes adopted, some of the problems encountered during implementation and finally includes the evaluation and testing of the data.

The main aim of this project was to first, design a reasonable model to describe the relationship between players. Secondly, generate a team rating system based on exploring player-player relationships. In the evaluation section, there are two images of the r-squared based on different real data. The first one is based on the average Elo rating and the average level of the players as the real data. The other one takes the results of another rating system and normalizes them and compares them with our ratings as real data. It was found that some of the 14 famous combinations had a positive team chemistry value and some had a negative relationship value. It was found that most of the combinations with a negative value were those where the individual players were very good, but the team did not end up playing very well that season. The data used for this project is not large enough at the moment, so there is a risk of inaccuracy. There are also some role players whose details were not available when the player information was entered. So there is a good chance that inaccuracies will occur. Of course, the final data will have to be compared and improved by setting different real data. However, based on the ratings and data obtained so far I believe that the objectives and requirements defined in this article have been achieved.

## Future work

1. Expand data

It is possible that the final results of this project are not very accurate because the data stored is too small, which only has information on 14 teams and over 140 players. In order to ensure the accuracy of the data, it is necessary to store more data. It is also crucial to devise a method to calculate individual player ability values, as they are currently calculated by player efficiency.

1. Against AI mode

For the moment for this project, the mode only supports confrontations between two players, and players will be added to play against Ai later. This might be done by learning TensorFlow and using TensorFlow to implement machine learning to set up how the Ai works. The AI is currently set up to work in two ways, the first being brute force hacking and the second being using tensor flow to train the machine and then pit the player against the AI. The player can choose the mode and difficulty according to their needs.

1. Optimize the UI

The current UI of this project is still very plain and the layout is not yet well set up. Players can only see a very limited amount of information about the player when selecting a player and are not yet able to see the player's ability stats such as three points, two points, defense etc. It would be helpful if this feature was added to the UI, especially for those who don't know the NBA, so that they can have a better understanding of each player's characteristics and outstanding abilities. In addition, the structure of the UI could be adjusted and made more colorful, which would greatly improve the user experience.

1. Expand the model

This project can now give a rating based on the player's choice of player, With more information and time, it is possible not only to display each team's rating on the final page, but also to set a prediction of the number of points a player will receive during this simulated rating. For example, how many points Player A has scored, how many goals he has stolen.

# 9.appendix

ProBasketballTalk | NBC Sports. 2021. *Daryl Morey says 3-pointers worth too much: 'It skews the game'*. [online] Available at: <https://nba.nbcsports.com/2021/04/06/daryl-morey-says-3-pointers-worth-too-much-it-skews-the-game/>

Harrington, W., 2021. *What team chemistry really means — The Basketball Movement*. [online] The Basketball Movement. Available at: <https://thebasketballmovement.com/blog/2019/4/27/what-team-chemistry-means>

FiveThirtyEight. 2021. *The Complete History Of The NBA*. [online] Available at: <https://projects.fivethirtyeight.com/complete-history-of-the-nba/#lakers>

livescience.com. 2021. *What Is the Fibonacci Sequence?*. [online] Available at: <https://www.livescience.com/37470-fibonacci-sequence.html>

Koroleva, L., 2018. *Memoization. Improving Recursive Solution for Fibonacci Sequence Problem.*. [online] Medium. Available at: <https://medium.com/@nothingisfunny/memoization-improving-recursive-solution-for-fibonacci-sequence-problem-c02dab7a74e5> [Accessed 11 April 2021].

Basketball-Reference.com. n.d. Calculating PER | Basketball-Reference.Com. [online] Available at: https://www.basketball-reference.com/about/per.html [Accessed 7 November 2020].

Fein, Z., 2009. Cracking The Code: How To Calculate Hollinger's PER Without All The Mess. [online] Bleacher Report. Available at: https://bleacherreport.com/articles/113144- cracking-the-code-how-to-calculate-hollingers-per-without-all-the-mess [Accessed 10 November 2020]

Geeks for Geeks. 2021. *Python Tkinter Tutorial - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/python-tkinter-tutorial/#:~:text=Tkinter%20is%20the%20most%20commonly%20used%20library%20for,applications%20with%20Tkinter%20becomes%20the%20fastest%20and%20easiest.> [Accessed 11 April 2021].

OpenGenus IQ: Learn Computer Science. 2021. *Graph Representation: Adjacency Matrix and Adjacency List*. [online] Available at: <https://iq.opengenus.org/graph-representation-adjacency-matrix-adjacency-list/> [Accessed 11 April 2021].

BCS The Chartered Institute for IT, “BCS Code of Conduct,” 5 June 2019. [Online].Available: https://www.bcs.org/membership/become-a-member/bcs-code-of- conduct/. [Accessed 1 November 2020].

School of Engineering and Informatics, The University of Sussex, “Information for Students,” 2020. [Online]. Available: http://www.sussex.ac.uk/ei/internal/forstudents/informatics/undergraduate/finalyearprojects/i nformationforstudents. [Accessed 2 November 2020].