**2023년 2학기**

**프로그래밍과 문제해결**

**Assignment #2**

**담당교수: 윤은영**

**학번: 20230563**

**학과: 무은재학부**

**이름: 김홍근**

**POVIS ID: hongsimi7**

**명예서약(Honor code)**

**“나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.”**

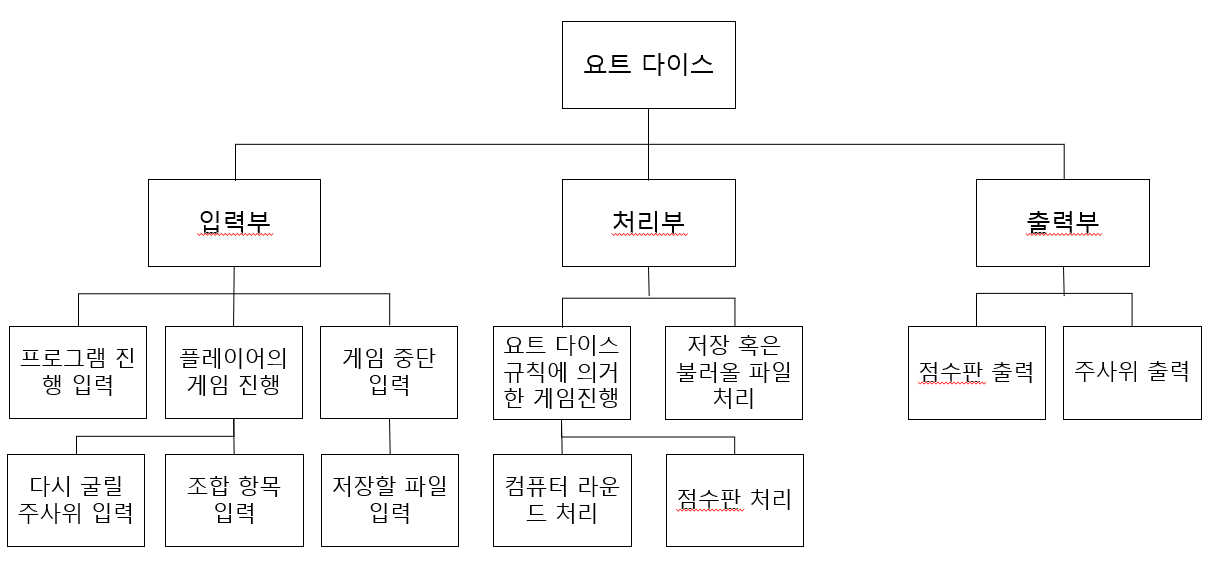
**Problem 2: 요트 다이스 (Yacht dice)**

**1. 문제의 개요**

본 프로그램을 간략히 설명하면 다음과 같다.

* 주사위 5개를 사용하여 미리 지정된 주사위 조합으로 최고 점수를 얻어야 한다.
* 주사위는 한 턴에 총 3번까지 굴릴 수 있으며, 굴리지 않거나 부분적으로 굴리는 것이 가능하다.
* 이미 채워진 점수판에는 점수 입력이 불가능하며, 조합과 맞는 항목에 점수를 넣을 경우, 0점으로 기록한다.
* 두 명의 플레이어(컴퓨터, 사용자)가 게임을 진행하게 되며, 총 라운드는 12라운드이고, 순서는 번갈아 가면서 진행된다.
* 두 플레이어가 모든 점수판을 채운다면, 점수를 합산하여 더 높은 점수를 가진 플레이어가 승리하게 된다.

이 때 사용되는 구상 가능한 구조 차트(structure chart)는 아래와 같이 표현될 수 있다.



* 입력부 : 플레이어가 새로 게임을 진행할 것인지, 게임을 불러올 것인지, 종료할 것인지

입력 받는다.

플레이어가 어떤 주사위를 다시 굴릴 것인지 입력 받는다.

플레이어가 어떤 조합 항목을 선택할 것인지 입력 받는다.

게임 중단을 입력 받았다면, 저장할 파일 이름을 입력 받는다.

* 처리부 : 요트 다이스 규칙에 의거해 게임을 진행한다.

플레이어 혹은 컴퓨터가 지정한 주사위 조합과 점수 조합 항목으로 점수판에 알맞게 처리한다.

컴퓨터는 주사위를 굴릴 때 가장 높은 점수의 조합 항목을 기준으로 주사위를 다시 굴리도록 처리한다.

게임 이어하기를 입력 받고 불러올 파일을 입력 받았다면, 그것이 올바른 파일인지 확인한다.

저장할 파일을 입력 받았다면, 올바르게 파일을 저장한다.

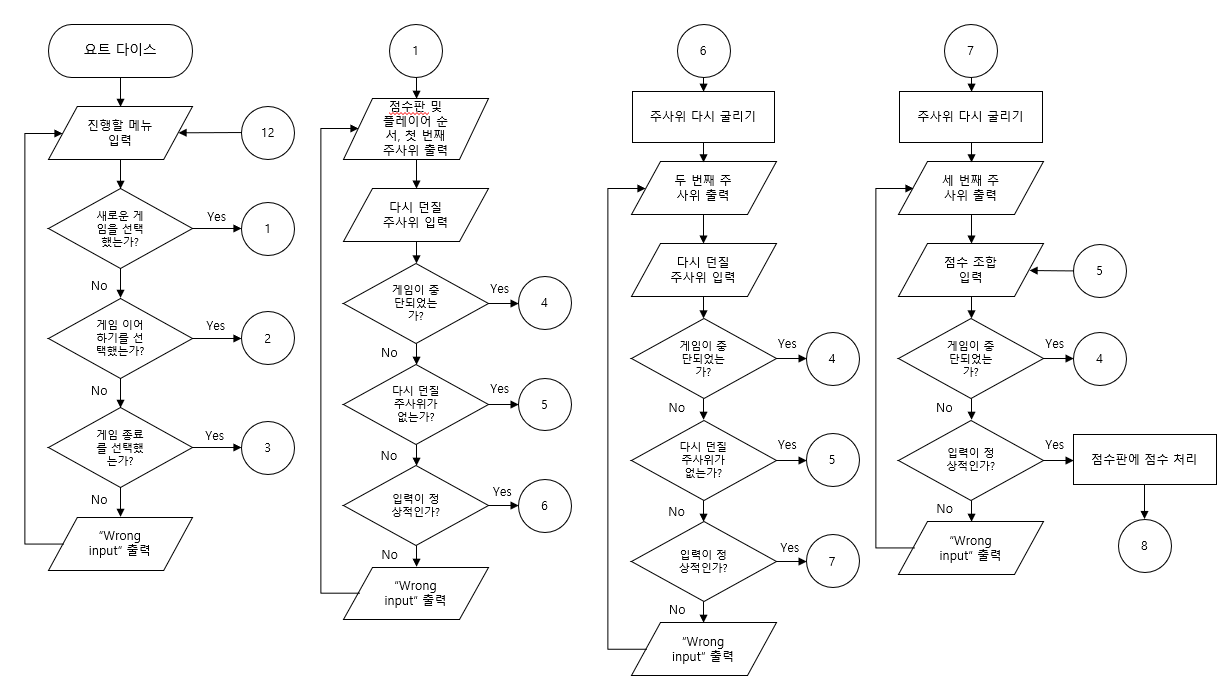
* 출력부 : 현재까지 진행된 게임의 점수판을 출력한다.

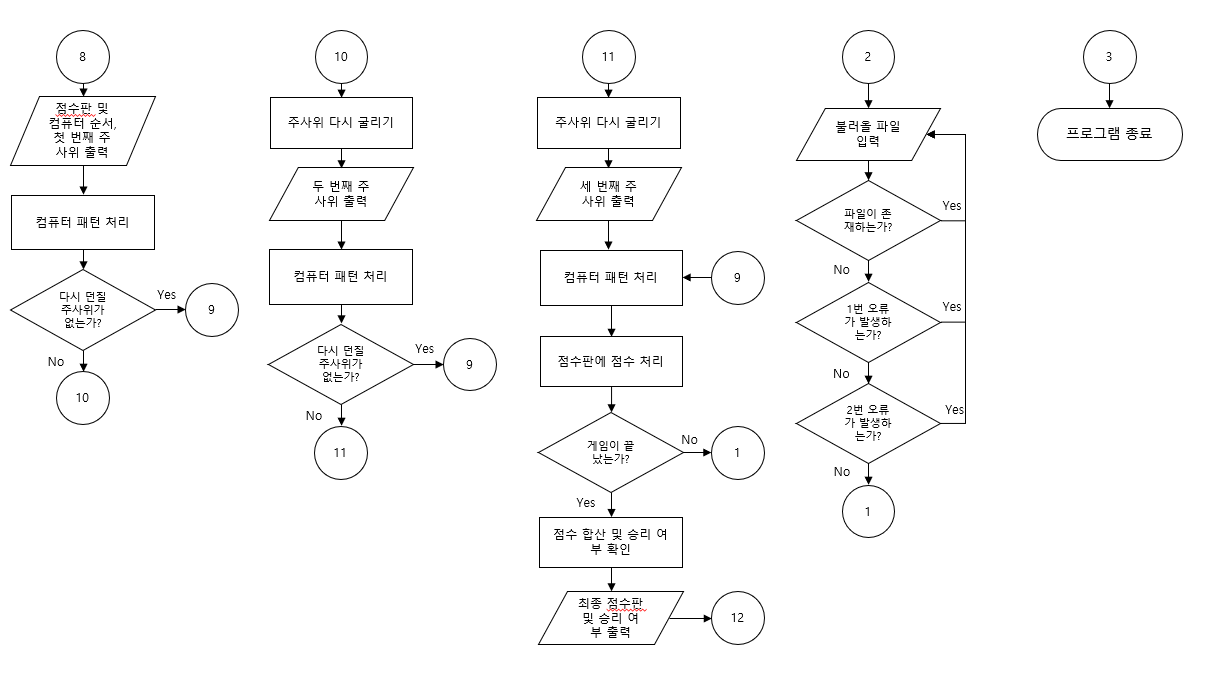
컴퓨터가 선택한 주사위 조합 및 점수 조합 항목을 출력한다.

플레이어가 굴린 주사위 조합을 출력한다.

**2. 알고리즘**

본 프로그램 작성을 위한 알고리즘을 Flowchart 형태로 나태내면 다음과 같다.





**3. 프로그램의 구조 및 설명**

a) 점수판 출력

- print\_score\_board 함수를 통해 현재까지 진행된 게임의 점수판을 출력한다.

b) 플레이어의 순서

- roll\_dice 함수를 이용하여 주사위 조합을 반환한다.

- 어떤 주사위를 다시 굴릴지 아님 게임을 중단할 것인지 입력을 받는다.

- calc\_score 함수를 이용하여 주사위 조합과 플레이어가 선택한 점수 조합, 점수 리스트, 플레이어 턴을 매개변수로 받아 점수 리스트를 반환한다.

- 게임이 중단되었다면, 저장할 파일이 무엇인지 입력 받고 저장한다.

c) 컴퓨터의 순서

- roll\_dice 함수를 이용하여 주사위 조합을 반환한다.

- computer\_pattern 함수를 이용하여 주사위 조합과 점수판을 매개 변수로 전달받아 상황에 가장 적합한 점수 조합과 그에 맞는 다시 굴릴 주사위의 인덱스 리스트를 반환한다.

- calc\_score 함수를 이용하여 주사위 조합과 컴퓨터가 선택한 점수 조합, 점수 리스트, 컴퓨터 턴을 매개 변수로 받아 점수 리스트를 반환한다.

d) 게임 이어 하기

- 먼저 불러올 파일을 입력 받는다.

- os 모듈의 path.exists 함수을 이용하여 파일이 존재한지 확인한다.

- load\_file2list 함수를 이용하여 파일을 2차원 리스트로 변환한다.

- check\_error 함수를 이용하여 파일을 변환한 2차원 리스트가 정상적인지 확인한다.

e) 함수 설명

- print\_score\_board 함수는 점수 리스트를 매개 변수로 받아, 점수판을 출력하는 함수이다.

- load\_file2list 함수는 게임 이어하기를 선택할 시, 받은 파일명을 매개 변수로 받아 점수 리스트로 생성하여 반환하는 함수이다.

- check\_error 함수는 점수 리스트를 매개 변수로 받아 만약 오류 1이면 1을, 오류 2이면 2를, 정상적인 파일이면 그대로 점수 리스트를 변환하는 함수이다. 리턴 값을 False, True가 아닌 위와 같이 바뀐 이유는 오류를 함수에서 구분을 하고 더 효율적으로 코드를 짜기 위해서이다.

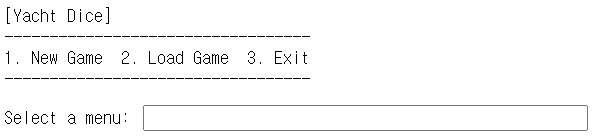
- roll\_dice 함수는 주사위 조합과 다시 던질 주사위의 인덱스 리스트를 매개변수로 받는다. 이 때, 두 매개변수는 디폴트 인수로 빈 리스트([])로 설정한다. 첫 번째 주사위를 굴릴 때에는 전 주사위 조합과 다시 던질 주사위의 인덱스 리스트가 없기 때문이다. 그리고 업데이트 된 주사위 리스트를 반환하는 함수이다.

- calc\_score 함수는 주사위 리스트, 선택한 점수 조합, 점수 리스트, 순서를 매개변수로 받아 업데이트된 점수 리스트를 반환하는 함수이다. 여기서 점수 리스트와 순서를 매개변수로 추가한 이유는 점수만 반환하는 것이 아닌 점수 리스트를 업데이트하는 것까지 구현하기 위해서이다.

- computer\_pattern 함수는 주사위 조합과 점수 리스트를 매개 변수로 받아, 컴퓨터 룰에 따라 선택된 점수 조합과 다시 던질 주사위의 인덱스 리스트를 반환하는 함수이다.

**4. 프로그램 실행 방법 및 예제**

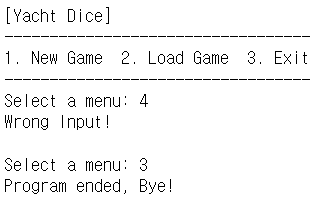
* Anaconda prompt의 Jupyter notebook 환경에서 프로그램을 실행한다.
* 프로그램을 실행했을 때, 시작 화면은 다음과 같다.



- 1을 선택하면 새 게임을, 2를 선택하면 게임 이어하기를, 3을 선택하면 게임을 종료한

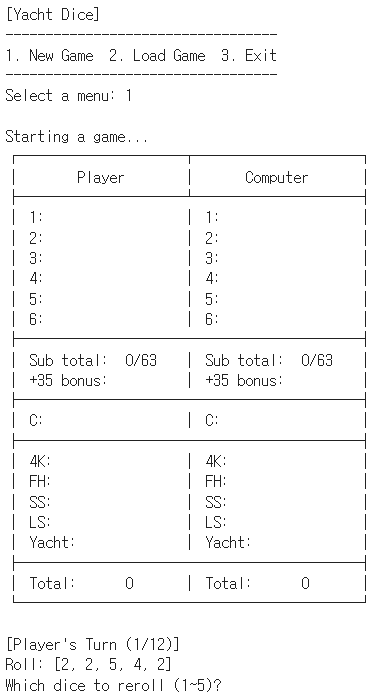
다. 이때, 1, 2, 3 이외의 입력을 받았다면, Wrong Input! 메시지를 출력한 다음, 다시 입

력 받는다. 예시는 다음과 같다.



- 만약 1을 선택한다면, 다음과 같이 각 플레이어의 점수판을 출력한 후, 플레이어부터

주사위를 던져 게임을 시작한다.



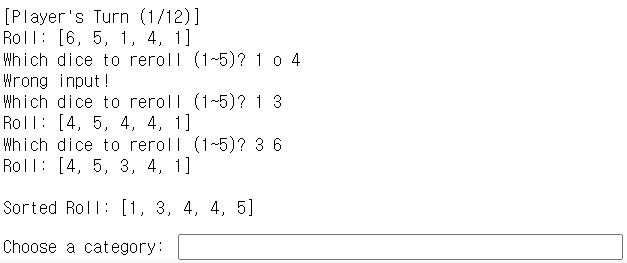
- 플레이어는 5개 주사위로, 최대 3회까지 주사위를 던질 수 있다. 원하는 조합을 맞추기

위해 최대 2회까지 주사위를 다시 던질 수 있다. 다음은 o 입력으로 인한 에러 메시지

출력과 6 위치 무시 후 3번 주사위만 reroll한 예시다. 또한, 주사위가 첫 번째 주사위

에서 두 번째 주사위로 바뀔 때 1, 3번 주사위가, 두 번째 주사위에서 세 번째 주사위

로 바뀔 때, 3번 주사위가 바뀐 것을 확인할 수 있다.

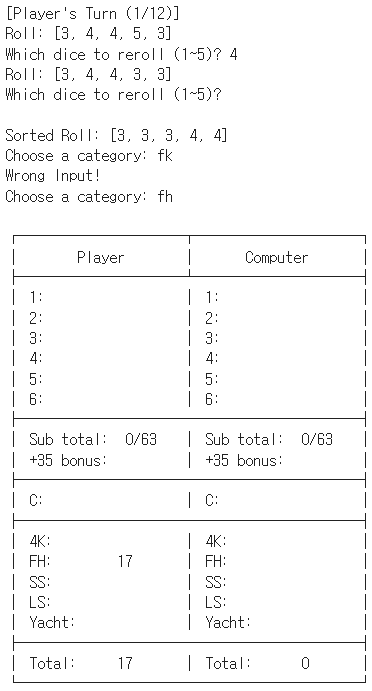


- 주사위 던지기가 끝났다면, Player의 점수판에서 자신이 원하는 항목을 채워 넣는다. 이

때, 주사위 조합은 오름차순으로 정렬되어 출력된다. 다음 예시는 올바르지 않은 입력

을 받았을 때, Wrong input을 출력하고 다시 입력을 받는다. 또한, FH를 입력하여 점수

판에 기록된 것을 확인할 수 있다.



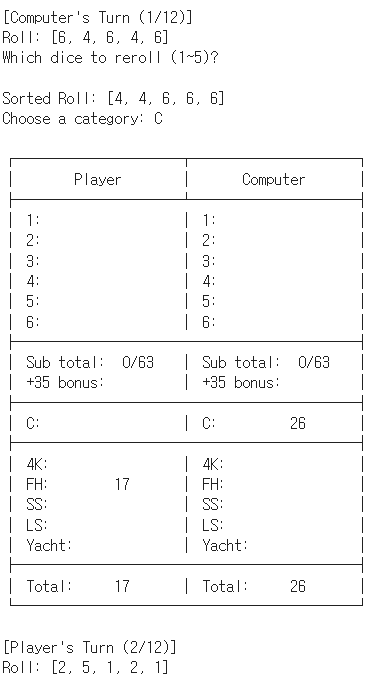
* Player의 턴이 끝나면, computer의 순서가 된다. 컴퓨터는 일정한 룰에 따라 행동을 한

다. 아래의 예시에서는 모든 점수 조합에서 C와 4K가 가장 높다. 이 때, 가장 높은 점수

칸 중 어떤 칸을 선택해도 무방하기에 C를 선택한 모습을 볼 수 있다. 또한, C에서는 3

이하의 주사위를 다시 던지도록 되어 있는데, 3이하의 주사위가 없으므로 던지지 않는

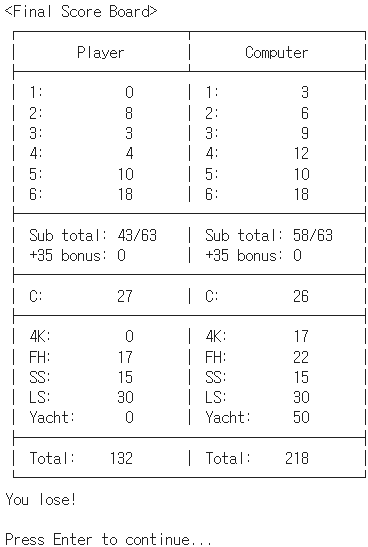
모습을 볼 수 있다. 그리고 C칸에 점수가 입력된 모습을 볼 수 있다.



- 모든 플레이어가 모든 점수판을 다 채운다면, 최종 점수판이 출력되고, 최종 점수를 합

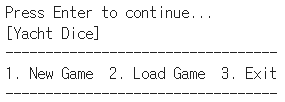
산하여 점수를 비교한 다음, 승패를 결정한다. 다음 예시는 컴퓨터의 점수가 높으므로

플레이어가 진 모습일 볼 수 있다.



- 다음 예시는 엔터키를 누른 다음, 초기 화면으로 돌아가는 모습이다

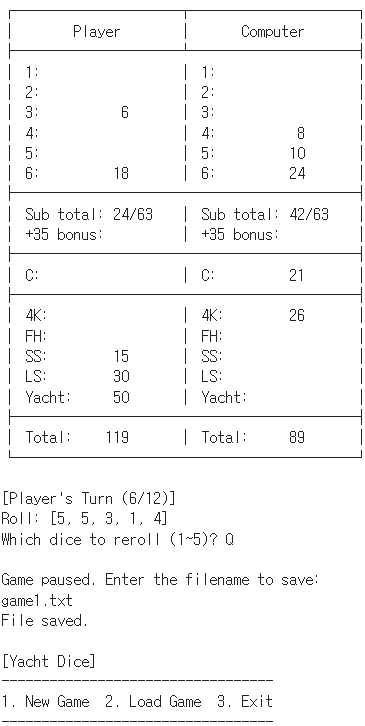
.



* 만약 게임 중간에 사용자 입력이 있을 때, “Q” 입력을 받았다면, 게임을 중단한다. 아래

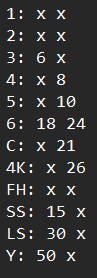
예시처럼 게임이 중단된 다음, 저장할 파일 이름을 입력 받는다. 그런 다음, 점수판을

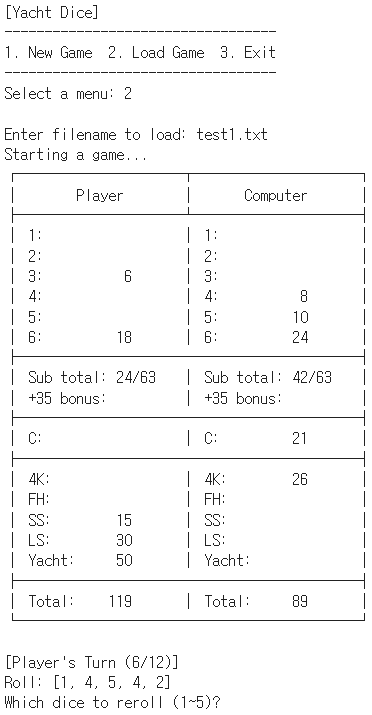
파일에 저장한 후, 초기 메뉴로 돌아간다.



* 초기 메뉴에서 만약 2를 선택했다면, 점수판이 저장된 파일을 읽어와 점수판을 채운 후,

게임을 진행한다. 다음 예시는 파일을 불러온 모습과 txt 파일이다.





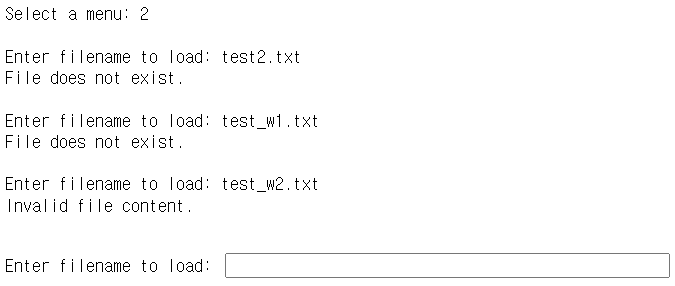
* 만약 파일을 입력 받았을 때, 파일 자체가 존재하지 않는다면, “File does not exist.” 메시

지를 출력하고 다시 파일을 받는다. 또한, 총 줄의 개수가 12가 아닌 경우, “File does

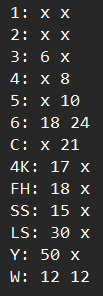
not exist를 출력하고 다시 파일을 받는다. 주사위 조합으로 불가능한 점수가 적혀 있는

경우, “Invalid file content.”를 출력한 다음, 다시 파일을 받는다. 다음은 잘못된 파일 예

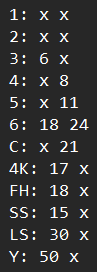
시와 프로그램 실행 시 출력된 문구들이다.



[test\_w1.txt]



[test\_w2.txt]



**5. 토론**

- 모듈을 사용하는 방법을 알았고, random 모듈을 불러와 주사위 조합을 랜덤으로 설정할 수 있었다.

- 2차원 리스트의 활용 방법을 알게 되었다.

- 리스트의 Mutable한 특성을 알게 되었다.

- sallow copy와 deep copy의 특징을 알고 이를 적용할 수 있었다.

- 파일을 불러오고 닫는 방법을 사용할 수 있었다.

**6. 결론**

- 본 과제에서는 if-else문, while문, 함수 지정, 파일을 열고 읽고 쓰기를 사용하는 방법을 익히는데 유용했으며, Python의 sallow copy와 deep copy를 적용해 볼 수 있었다. 또한, 알고리즘을 작성 및 디버깅하는 방법을 학습할 수 있었다.

**7. 개선방향**

- 본 과제를 할 때, 무분별한 복사 및 붙여넣기로 효율적인 코드를 짜지 못했다. 따라서 800줄이라는 양의 코딩을 만들어 버렸다. 새로운 게임을 시작할 때와 게임 이어하기를 진행할 때, 게임을 진행하는 것은 똑같으므로 게임 진행을 함수로 제작하면 더 깔끔한 코드를 짤 수 있을 것으로 예상했다.