

```
Microsoft Visual Studio 디버거 콘솔

**** 갠블링 게임을 시작합니다. ****
첫번째 선수 이름>>길동
두번째 선수 이름>>춘향
길동: <Enter>

길동의 점수: 0 1 0
아쉽군요!
춘향: <Enter>

춘향의 점수: 2 1 1
아쉽군요!
길동: <Enter>
모
잘못된 입력입니다. 다시 입력해주세요.
길동: <Enter>

길동의 점수: 1 2 0
아쉽군요!
춘향: <Enter>
모
잘못된 입력입니다. 다시 입력해주세요.
춘향: <Enter>

춘향의 점수: 2 0 2
아쉽군요!
길동: <Enter>

길동의 점수: 0 1 0
아쉽군요!
춘향: <Enter>

춘향의 점수: 1 0 1
아쉽군요!
길동: <Enter>

길동의 점수: 0 0 0
길동님이 승리!!

C:\vc++vs_workspace\C++\HomeWork#1\hw\hw#2\64\Debug\hw#2.exe(프로세스 31988)이(가) 0 코드(0x0)와 함께 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요...
```

소스 구현 설명

-문제 정의 : 이름을 입력받고 랜덤값 3개 생성, 빈값인 엔터를 클릭시 본인의 숫자를 보여주고 본인의 숫자가 0 0 0 처럼 연속적으로 동일한 숫자면 이긴다. 서로 번갈아가며 하는거며 2명의 선수는 배열로 구성하고 랜덤 숫자는 0~2로 제한한다. 선수는 Player클래스, 게임은 GameblingGame 클래스로 작성한다. 엔터 클릭시만 넘어가야 되므로 엔터값이 아니면 오류문구나 다시 하라는 문구로 돌아가서 엔터값을치게 만든다(개인사항).

- 문제 해결방법:
1. Player 클래스를 정의하며 score를 배열로선언 string name값을 선언한후 생성자를 만들어 score[]배열의 값을 0으로 초기화(혹시모를 오류대비)
 2. makeScore 함수를 만들어 rand()%3을 통해 0,1,2의 랜덤정수를 배열에 저장한다.
 3. GamblingGame 클래스를 정의하며 Player를 배열로 선언하고 string input을 선언한다 input은 "" 빈값인지 아닌지 확인할 변수이다.
 4. play함수를 정의하며 선수이름을 입력받는다 입력받은후 cin.ignore()를 쓰는데 버퍼에 남아 있는 개행문자를 제거하는 것으로 이게 없으면 전에 입력했던 엔터값이 버퍼에 남아 있기 때문에 다음 입력값에서 아무것도 안해도 "" 빈값이 넘어간다(문제 푸는도중 자주 넘어가서 따로 알아보아서 추가하였습니다.)
 5. 얼마나 반복할지 모르기에 while문으로 전체를 묶었으며 for문을 이용하여 첫번째 순서 사람일때 , 두번째 순서 사람일때를 정의하였으며 이 또한 사용자가 얼마나 엔터값이 아닌 다른 값을 넣을줄 몰라 while문으로 조건식을 묶었으며 3개 동일 숫자일 경우 return;으로 함수를 종료하고 3개가 동일 숫자가 아닐경우 break로 반복문 하나를 나간뒤 다음 for문을 통해 2번째 선수로 넘어간다. 빈값이 아닐시 continue문으로 다시 while을 반복하여 빈값을 넣을 수 있도록 한다.
 6. 메인 함수에서 srand(static_cast<unsigned int>(time(0))); 를 사용하여 랜덤 시드 초기화를 통해 여러번 게임을 실행해도 각각 다른 숫자가 나오도록 한다.(문제 푸는 도중 계속 같은 값만 나오기에 따로 알아보았습니다.)
 7. GamblingGame game; 선언후 game.play()를 통해 게임을 시작하고 모든것이 완료되면 return 0;을 통해 시스템에게 종료 되었다고 알려줍니다.

-아이디어 평가: 처음에는 여러개의 while문을 중첩시키며 복잡하게 만들었지만 코드가 너무 복잡하고 난잡하여 수정하다 보니 for문을 통해 만들면 좀더 코드를 간결하게 짤 수 있다는 것을 생각해 내었으며 cin.ignore() , srand(static_cast<unsigned int>(time(0))); 를 통하여 개행문자 제거, 시드 초기화를 통해 오류를 해결하여 이상적인 겜블링 코드를 만들었다고 생각 합니다.

-문제를 해결한 키 아이디어 또는 알고리즘 설명
while문에서 for과 다시 while문의 중첩이 중요 알고리즘 이며 srand, cin.ignore이 중요 합니다.
게임을 순서를 지키며 반복 시키는 코드가 짜기가 헛갈렸으며 특히 for문 없이 작성한다면 코드가 너무 길어지기에 for문이 중요 하였으며 srand를 통해 여러번 반복해도 같은 값이 나오지 않게 만들어야 했으며 cin.ignore을 사용하여 버퍼에 남은 개행문자 제거를 통해 오류를 통제하였습니다.]