

문제 정의

두 명의 플레이어가 번갈아 가며 무작위 숫자 3개를 생성해 세 개의 숫자가 모두 동일하면 해당 플레이어가 승리하는 게임을 만들면 됩니다

이 게임의 규칙은

- 두 명의 플레이어가 차례대로 세 개의 무작위 숫자를 생성

- 숫자의 범위는 0부터 2까지

- 세 숫자가 모두 동일할 경우 그 플레이어가 승리

- 승자가 나올 때까지 게임이 계속 반복

을 명심하고 코드를 짜면 되겠습니다.

문제 해결 방법

위 기능들을 구현하기 위해 무엇이 필요한가 생각해보면

각 플레이어의 이름과 숫자 생성 기능을 관리하는 클래스인 Player 클래스

0~2 사이의 무작위 숫자 3개를 생성하고, 그 숫자들이 모두 동일한지 판단하는
Player::playGambling() 함수

두 명의 플레이어가 번갈아 가며 게임을 진행하고 승리 여부를 판정하는 GamblingGame 클래스

명의 플레이어 이름을 입력받아 게임을 초기화하는 GamblingGame::GamblingGame() 생성자

플레이어가 번갈아가며 차례를 진행하고, 승리 여부를 판단해 게임을 종료하는
GamblingGame::play() 함수

게임 객체를 생성하고 게임을 시작하는 역할을 담당하는main() 함수

들을 생성해 구동하면 될듯합니다

아이디어 평가

위 코드들을 구현하고 실행하던 중 엔터 키를 누르지 않아도 결과가 자동으로 출력되어 게임이 원활하게 진행되지 않는 상황이 발생해 살펴보니 플레이어 이름을 입력받는

GamblingGame::GamblingGame() 생성자와, 플레이어가 번갈아가며 차례를 진행하는

GamblingGame::play() 함수의 입력 처리 과정에서 getchar() 함수를 사용하지 않았습니다. 이로 인해 사용자가 엔터 키를 누르지 않아도 결과가 자동으로 출력되어 게임이 원활하게 진행되지 않

있던 것을 알 수 있었으며, `getchar()` 함수를 추가하였더니 정상적으로 작동하여 이러한 오류를 해결했습니다.

문제를 해결한 키 아이디어 또는 알고리즘 설명

이문제를 해결하기 위해 사용된 키 아이디어 또는 알고리즘은 크게 세가지가 있었습니다.

-객체지향

`Player` 클래스와 `GamblingGame` 클래스같이 각 플레이어와 게임 자체를 독립적인 클래스로 설계하여, 각 요소의 책임을 분명히 했습니다

-무작위 숫자 생성 및 승리 조건 확인

`rand() % 3`을 통해 0, 1, 2의 숫자를 생성하고, 배열의 첫 번째 숫자와 두 번째 숫자, 두 번째 숫자와 세 번째 숫자를 차례대로 비교해 승리 여부를 판정하는 방식의 알고리즘을 사용했습니다.

-순서와 게임 종료 조건

두 명의 플레이어가 번갈아 가며 차례를 진행하기 위해 $i \% 2$ 연산을 사용하여 현재 차례가 누구인지 결정합니다. i 가 짝수이면 첫 번째 플레이어, 홀수이면 두 번째 플레이어의 순서가 되도록 설계했습니다.

승리자가 나오면 `isGameCompleted` 변수를 `true`로 설정하여 게임을 종료시켰습니다.