

# 객체지향프로그래밍 5주차 과제 소스구현 설명

ICT융합학부 소프트웨어학과  
202284002 고범주

## 1.문제 정의:

이 프로그램은 두 명의 플레이어가 번갈아 가며 0부터 2 사이의 랜덤 숫자를 뽑고, 이 숫자들이 모두 동일하면 승리하는 간단한 갬블링 게임을 구현하는 것입니다. 각 플레이어는 자신의 차례에 Enter 키를 눌러 세 개의 숫자를 생성하며, 이 숫자들이 모두 동일할 경우 그 플레이어가 승리합니다. 목표는 두 플레이어 간의 대결을 통해 누가 승리하는지를 결정하는 것입니다.

## 2.문제 해결 방법:

### (1)플레이어 클래스 생성

각 플레이어를 나타내는 Player 클래스를 생성합니다. 이 클래스는 플레이어의 이름과 랜덤 숫자(카드)를 관리합니다.

### (2)갬블링 클래스 생성

게임의 진행을 관리하는 GamblingGame 클래스를 구현합니다. 이 클래스는 두 명의 플레이어 객체를 포함하며, 플레이어의 차례를 관리하고 게임의 흐름을 제어합니다.

### (3)입력 및 출력 처리

플레이어의 이름을 입력받고, 각 플레이어의 차례에 따라 랜덤 숫자를 출력합니다. 결과에 따라 승리를 판단하고 최종 결과를 출력합니다.

## 3.아이디어 평가

### (1)플레이어 클래스를 통한 구조화

Player 클래스를 사용하여 각 플레이어의 정보를 구조화함으로써 코드의 가독성과 유지보수성을 높였습니다. 각 플레이어에 대해 별도의 객체를 생성함으로써 상태를 독립적으로 관리할 수 있습니다.

### (2)랜덤 숫자 생성

rand() 함수를 사용하여 랜덤 숫자를 생성하는 방식은 간단하고 효율적입니다. 0부터 2까지의 숫자를 생성하는 것은 간단한 논리로 구현되었습니다.

### (3)게임의 흐름 관리

GamblingGame 클래스 내에서 플레이어의 차례를 관리하고 승리 조건을 확인하는 로직을 통해 게임을 원활하게 진행할 수 있었습니다. 각 차례에 대해 반복문을 사용하여 무한 루프를 만들고, 승리 여부에 따라 루프를 종료했습니다.

## 4.문제를 해결한 키 아이디어 또는 알고리즘 설명

### (1)Player 클래스

int card[3]: 플레이어가 차례에 생성하는 세 개의 랜덤 숫자를 저장하는 배열입니다. 숫자는 0~2 사이에서 랜덤하게 결정됩니다.

char name[100]: 플레이어의 이름을 저장하는 문자열입니다. strcpy() 함수를 사용하여 초기화합니다.

Player(): 기본 생성자, 이름을 "플레이어"로 초기화합니다.

Player(const char\* name): 이름을 사용자로부터 입력받아 초기화하는 생성자입니다.

const char\* getName(): 플레이어의 이름을 반환하는 함수입니다.

bool playGambling(): 랜덤 숫자를 생성하고 승리 조건을 검사하는 함수입니다. 3개의 카드 숫자를 생성하고 모두 같으면 true, 다르면 false를 반환합니다.

## (2)GamblingGame 클래스

Player player[2]: 두 명의 플레이어 객체를 배열로 저장합니다. 각 플레이어의 이름과 게임 상태를 관리합니다.

bool isGameEnd: 게임이 종료되었는지를 나타내는 플래그 변수입니다.

GamblingGame(): 두 명의 플레이어 이름을 입력받아 초기화하는 생성자입니다.

void play(): 게임의 주 로직을 처리합니다. 플레이어의 차례에 따라 숫자를 생성하고 승리 조건을 체크합니다.

## (3)게임 진행

GamblingGame::play() 함수에서 무한 루프를 통해 플레이어의 차례를 반복적으로 진행합니다. 각 플레이어가 <Enter>를 눌러 랜덤 숫자를 생성하고, 승리 조건을 검사하여 게임을 종료합니다. 게임의 종료 여부는 isGameEnd 변수를 통해 관리합니다.