

문제 정의

여러 클래스로 이루어진 객체 지향 프로그래밍을하는 문제이다.

선수를 관리하는 Player 클래스와, 게임 진행을 담당하는 GamblingGame 클래스 2개를 만들어 랜덤으로 나오는 숫자 3개를 맞추는 슬롯머신과 같은 게임을 만드는 문제이다.

문제 해결 방법

Player클래스와 GamblingGame클래스 2개를 만든다.

Player클래스에는 이름을 설정하는 Player(string name), 이름을 불러오는 getName()을 추가한다.

GamblingGame클래스에는 두 선수를 포인터 배열인 Player* players[2], 이름을 배열에 저장하는 GamblingGame(string name1, string name2), 소멸자 ~GamblingGame, 게임을 진행하는 play(), 랜덤으로 숫자를 뽑아 같은지 아닌지를 판단하는 ranNum(string name)을 추가한다.

숫자는 랜덤으로 뽑는다. 각 차례 숫자는 엔터를 치면 뽑는다. 숫자의 범위는 0~2

게임은 승리자가 나올 때까지 반복한다.

승리자가 나오면 소멸자가 선수 배열을 초기화 시키고 코드를 종료한다.

아이디어 평가

일단 각 클래스의 구현부와 선언부를 헤더파일과 cpp파일로 나누어 작성하였다. 나누어서 작성을 계속하니 코드 관리하는 게 쉬워졌다.

main에서 선수 두 명의 이름을 입력받아 GamblingGame클래스의 GamblingGame함수에 넣어 포인터 배열에 두 선수의 이름을 저장했다.

그 후 ranNum함수에서 숫자를 뽑는다. 숫자는 랜덤으로 뽑기 위해 rand()를 이용하는데, 그냥 rand()를 사용하면 매번 같은 수만 나오기 때문에 srand((unsigned)time(NULL))을 이용해서 랜덤한 난수값을 생성해주는 seed값을 변경해서 매번 다른 수가 나오게 설정했다.

각 선수의 차례에 엔터키를 눌러야 숫자가 뽑히게 해야하기 때문에 while문을 이용해 엔터를 감지할 때까지 코드를 잡고 있다가 엔터를 치면 break해서 숫자를 뽑게 했다.

숫자가 뽑히면 num이라는 배열에 저장되고 모든 숫자가 같으면 true를 다르면 false를 반환하도록 했다.

반환값을 가지고 play함수에서 while문으로 계속해서 각 플레이어 마다 bool값을 확인해서 true값을 가지는 플레이어가 나오기 전까지 계속해서 반복문을 반복한다. true값을 가진 플레이어가 나오면 break해서 반복문을 탈출하고 소멸자가 배열을 초기화하면서 코드를 종료한다.