

### 1. 알고리즘의 기본 목표:

컴퓨터는 그냥 무작위로 가위, 바위, 보를 내는 게 아니라, 실행자가 직전에 낸 수를 분석해서 그 수를 이길 수 있는 수를 내려고 노력합니다. 하지만 실행자가 이 패턴을 눈치챌 수도 있으니, 예측이 힘들도록 일정 확률로 무작위 수를 섞어 냅니다.

### 2. 필요한 정보:

- 실행자의 마지막 수: 실행자가 바로 직전에 냈던 수 (가위, 바위, 보 중 하나). 게임 시작 전에는 '없음'으로 초기화됩니다.
- 현재 진행 중인 라운드 수: 몇 번째 가위바위보 대결인지 기록합니다. 첫 번째 라운드인지 판단하는 데 사용됩니다.

### 3. 알고리즘의 단계별 진행:

게임을 시작하고 각 라운드가 시작될 때마다 컴퓨터는 다음 단계를 따라 움직입니다.

#### 단계 1: 라운드 시작 및 실행자 입력 대기

- 현재 라운드 수를 1 증가시킵니다.
- 실행자에게 가위, 바위, 보 중 하나를 내도록 요청합니다.
- 실행자가 낸 수를 입력받아 저장합니다. (이 수는 다음 라운드에서 '실행자의 마지막 수'가 됩니다.)

#### 단계 2: 컴퓨터의 수 결정

여기서 컴퓨터가 어떤 수를 낼지 결정합니다.

##### 만약 첫 번째 라운드라면 (실행자의 마지막 수가 '없음'이라면):

- 컴퓨터는 아직 실행자에 대한 정보가 전혀 없기 때문에 그냥 완전히 무작위로 가위, 바위, 보 중에 하나를 선택합니다. (각각 1/3 확률)

##### 만약 첫 번째 라운드가 아니라면 (실행자의 마지막 수가 있다면):

컴퓨터는 실행자가 바로 직전에 냈던 수를 기억합니다 ('실행자의 마지막 수').

그 '실행자의 마지막 수'를 이길 수 있는 '카운터 수'를 결정합니다.

- 실행자의 마지막 수가 '가위'였다면, '카운터 수'는 '바위'.
- 실행자의 마지막 수가 '바위'였다면, '카운터 수'는 '보'.
- 실행자의 마지막 수가 '보'였다면, '카운터 수'는 '가위'.

자, 이제 컴퓨터는 '카운터 수'를 알고 있어요. 근데 이걸 무조건 내면 실행자에게 패턴을 들켜므로 랜덤 확률을 사용합니다

컴퓨터는 0부터 1 사이의 임의의 숫자 하나를 뽑습니다.

##### 만약 뽑은 숫자가 0.7보다 작다면 (약 70% 확률):

- 컴퓨터는 방금 결정한 '카운터 수'를 냅니다.

##### 만약 뽑은 숫자가 0.7보다 크거나 같다면 (약 30% 확률):

- 컴퓨터는 '카운터 수'와 상관없이 완전히 무작위로 가위, 바위, 보 중에 하나를 선택

합니다.

이렇게 결정된 컴퓨터의 수를 저장해 둡니다.

### 단계 3: 승패 판정

- 실행자가 낸 수와 컴퓨터가 결정한 수를 비교하여 이겼는지, 졌는지, 비겼는지를 판정합니다.
- 일반적인 가위바위보 규칙을 따릅니다:
  - 바위 & 가위
  - 가위 & 보
  - 보 & 바위
  - 같은 수를 내면 무승부.

### 단계 4: 결과 표시 및 다음 라운드 준비

- 실행자가 낸 수, 컴퓨터가 낸 수, 그리고 판정 결과를 화면에 보여줍니다.
- **가장 중요!** 이번 라운드에서 플레이어가 낸 수를 '**실행자의 마지막 수**'로 업데이트합니다. 이 정보가 다음 라운드에서 컴퓨터의 결정에 사용됩니다.
- 실행자에게 게임을 더 할 것인지 묻습니다. 계속한다면 단계 1로 돌아가고, 그만한다면 게임을 종료합니다