

CSED 232 Object-Oriented Programming (Spring 2024)

Programming Assignment #5

Due date: 5월 17일 23시 59분 59초

담당 조교: 강병욱 (kbu417@postech.ac.kr)

[안내사항]

1. 모든 문제는 C++의 standard 입출력(i.e. `std::cin`, `std::cout`)을 기본으로 합니다.
2. 프로그램 기능 점수 기준은 채점용 testcase 통과 여부입니다.
 - 1) 기능별 점수는 채점기준을 참고해주세요.
 - 2) 채점용 testcase는 비공개입니다.

[감점]

1. 제출 기한이 지나면 얻은 총점의 20% 감점
2. 하루(24시간) 늦을 때마다 추가 20%씩 감점
 - a. 1일 이내: 20% 감점, 2일 이내: 40% 감점, 3일 이내: 60% 감점, 4일 이내 80% 감점
 - b. 4일 이상: 0점
3. 컴파일이 정상적으로 되지 않을 경우 최소 25% 감점

[제출방식]

본인 컴파일 환경에 따라 [채점환경]을 참고해서 제출해주세요.

파일을 업로드하실 때, **개발환경 파일의 "파일 제출"페이지에 써 있는 대로** 맞춰 올려 주시기 바랍니다. 폴더명은 학번(e.g. 20230000)으로 만들어 주시기 바랍니다. 또한 문제 폴더 안에 각 문제에 해당하는 Report(e.g. 20230000_report)도 첨부하여 zip파일로 압축한 후 제출해 주시기 바랍니다.

제출은 반드시 PLMS를 통해 제출해주시기 바랍니다. 이메일 제출은 받지 않습니다. 제출 기한을 기준으로 4일(5월 21일 23시 59분 59초)이 경과한 이후에는 0점이므로 PLMS에서도 과제 제출 받지 않습니다.

제출파일 예시) 20230000.zip

[채점기준]

1. 프로그램 기능 (설계 및 구현) – 85%

- 1) 프로그램이 요구 사항을 모두 만족하면서 올바르게 실행되는가?
- 2) 요구 사항을 만족하기 위한 프로그램(변수, 함수, 알고리즘 등) 설계가 적절한가?
- 3) 각 문제에서 제시한 세부 조건의 유의사항을 모두 만족하였는가?
- 4) 입력과 출력이 주어진 형식에 맞게 프로그램이 잘 작동하는가? (단순 오타 제외)
- 5) GGD, Goose, Bird, Goose, Duck 클래스 30점
- 6) BirdList 클래스 15점
- 7) MorticianGoose, DetectiveGoose, AssassinDuck 클래스 각 10점 (총 30)
- 8) Falcon, DodoBird 클래스 각 5점 (총 10)

2. 프로그램 가독성 – 5%

- 1) 변수 및 함수 명이 무엇을 의미하는지 파악하기 쉬운가?
- 2) 프로그램의 소스 코드를 이해하기 쉽도록 주석을 잘 작성하였는가?
- 3) GGD, BirdList 클래스 위주로 주석 작성해주세요.
- 4) 함수 선언시 함수의 역할 혹은 기능 등을 한 줄 정도로 요약
- 5) 함수 body에서는 주요 반복문 • branch문 위주로 주석 작성하시면 됩니다.

3. 보고서 구성 및 내용, 양식 – 10%

- 1) 보고서는 적절한 내용으로 이해하기 쉽고 보기 좋게 잘 작성되었는가?
- 2) 보고서의 양식을 잘 따랐는가?
- 3) 문제에서 제시한 질문이 있다면, 그에 대한 답변이 충분하고 적절한가?

[주의사항]

다른 사람의 프로그램이나 인터넷 등에 있는 프로그램을 단순히 복사(copy)하거나 수정해서 제출하면 부정행위로 간주됩니다. **부정행위 적발 시 'F' 학점을 받을 수 있으며, 학과에서 정한 기준에 따라 추가적인 불이익이 있을 수 있습니다.**

[채점환경]

1. Windows Visual Studio 2022

- 1) 그림 1 참고하여 제출
- 2) 채점 편의를 위해 프로젝트 이름은 "CSED232_Assignment5"로 통일 부탁드립니다.
- 3) x64폴더(2개 있음) 및 .vs폴더(숨김 폴더) 삭제해주세요.

2. g++ 13.1.0

- 1) 그림 2 참고하여 제출
- 2) Makefile 없을 시 visual studio 2022 환경에서 채점됩니다.

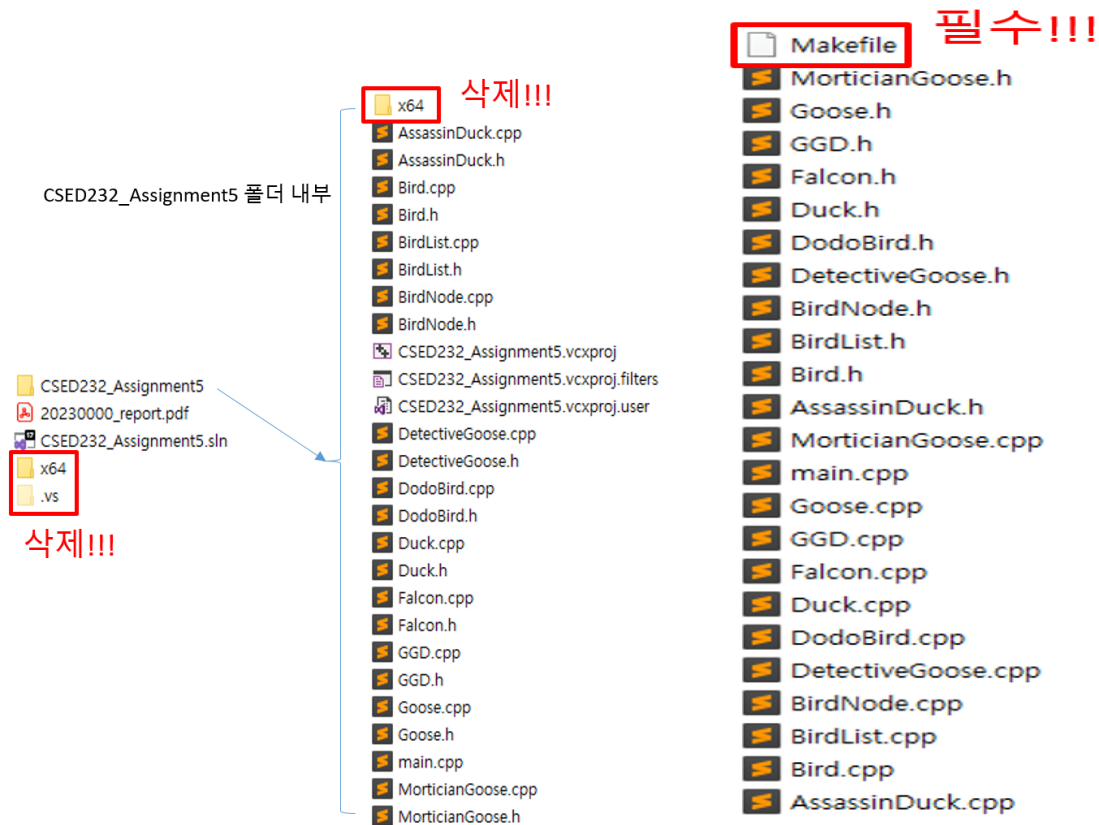


그림 1 Windows Visual Studio 2022 제출 방식

그림 2 g++ 13.1.0 제출 방식

문제: Goose Goose Duck

[문제 설명]

조교 강병욱은 최근에 연구실 동료들과 Goose Goose Duck 을 감명깊게 플레이하여, 이번 과제를 Goose Goose Duck 을 주제로하여 출제하기로 마음먹었다. GUI 과제가 아니므로, 텍스트 기반 게임으로 각색하여 과제를 출제하였다.

[Goose Goose Duck POSTECH-CSED232 Edition 게임 설명]

1. 게임 개요

Goose Goose Duck 은 온라인 멀티플레이 게임으로, 온라인 게임 '어몽어스'와 파티 게임인 '마피아'를 모티브로 하여 제작된 사회적 추론 게임이다. 서로 정체를 알고 있는 오리 진영과 서로 정체를 모르는 거위 진영간의 싸움으로, 각 진영마다 승리 조건을 달성하기 위해 라운드마다 상대 진영 구성원을 제거해 나가는 게임이다. 각 진영별 승리 목표와 라운드마다의 진행 방식은 후술.

2. 라운드 진행 방식

2.1. 역할별 능력 사용 Phase

2.1.1. 생존한 플레이어들은 순서대로 능력을 사용한다. 능력 사용 역할 순서는 다음과 같다. ⑥송골매(Falcon) ①암살자 오리(Assassin Duck) ②오리(Duck) ③탐정 거위(Detective Goose) ④장의사 거위(Mortician Goose) ⑤거위(Goose) ⑥도도새(Dodo Bird). 역할이 같은 플레이어간에는 플레이어 추가 순서를 따른다. 즉, 같은 역할간에는 먼저 추가된 플레이어일수록 먼저 능력을 사용한다.

2.1.2. 역할별 능력은 역할 설명항목을 참고.

2.2. 투표 Phase

2.2.1. 생존한 플레이어들은 순서대로 투표를 한다. 투표 순서는 플레이어 추가 순서를 따른다. 즉, 먼저 추가된 플레이어일수록 먼저 투표를 한다.

2.2.2. 각 플레이어는 자기 자신을 포함한 사망하지 않은 모든 플레이어에 대해 한 표를 행사할 수 있다. 아무 플레이어에게도 투표하지 않는 무효표를 행사할 수도 있다.

2.2.3. 가장 많은 투표를 받은 플레이어는 추방(사망처리) 당한다. 추방당한 플레이어가 오리인지 아닌지 모두에게 공개된다. 이 때, 무효표를 포함하여 가장 많은 투표를 받은 플레이어가 둘 이상일 경우 아무도 추방 당하지 않는다.

2.3. 승리 조건 달성 여부 판단

능력 사용 Phase 와 투표 Phase 가 종료되고나면 진영별 승리 조건을 판단하고, 승리한 진영이 있다면 게임 결과를 출력한 뒤 게임을 종료한다. 승리한 진영이 없다면 다음 라운드를 진행한다. 승리 조건은 진영별 승리 조건항목을 참고하라.

3. 진영별 승리 조건

3.1. 거위 (Goose)

거위 진영에 속한 역할은 다음과 같다: 거위(Goose), 탐정 거위(Detective Goose) 그리고 장의사 거위(Mortician Goose). 거위 진영에 속한 플레이어가 한 명이라도 생존해 있고, 오리 진영에 속한 모든 플레이어와 송골매가 사망한 경우에 거위가 승리한다. (도도새는 살상 능력이 없어 생존해 있어도 상관없다.)

3.2. 오리 (Duck)

오리 진영에 속한 역할은 다음과 같다: 오리(Duck) 그리고 암살자 오리(Assassin Duck). 오리 진영에 속한 플레이어가 한 명이라도 생존해 있고, 살아남은 오리 진영 플레이어의 수가 살아남은 다른 플레이어의 수보다 많거나 동일한 경우 오리가 승리한다. 단, 생존한 플레이어가 2 명 일 때, 생존한 조류가 오리 진영 조류가 한마리이고 송골매가 한마리인 경우에는, 오리의 승리가 아닌 송골매의 승리다.

3.3. 중립 (Neutral)





중립 진영은 거위와 오리 양측 모두와 승리 조건을 공유하지 않는다. 중립 진영에 속한 송골매와 도도새도 서로 승리 조건을 공유하지 않는다. 송골매와 도도새는 각각 한 마리씩만 존재할 수 있다.




3.3.1. 송골매는 자기 자신이 살아있고, 다른 플레이어가 한명 이하일 때 승리한다.

3.3.2. 도도새는 가장 많은 투표를 받아 추방당했을 때 남은 역할의 수와 종류에 관계없이 승리한다. 마지막에 도도새 혼자만 살아남는 상황은 일어나지 않는다고 가정한다. (도도새와 다른 플레이어 둘만 남은 상황에서 다른 플레이어는 무효표만 던져도 본인이 이길 수 있으니 도도새에게 투표하여 고의 패배를 하는 상황은 없다고 가정하자.)

4. 역할 설명

아래의 표를 참고하라. 번호는 능력 사용 Phase 에서의 순서와 사용자가 플레이어 추가시 입력할 때 사용하는 번호와 같다.

| 번호 | 이름 | 그림 | 설명 |
|----|-----------------------------------|---|---|
| 5 | 거위 (Goose) |  | 능력이 없는 역할이다. |
| 3 | 탐정 거 (Detective Goose) |  | 라운드마다 '조사' 능력을 1 회 사용하거나 사용하지 않을 수도 있다. 조사: 생존한 플레이어 중 한 명을 선택하여, 해당 플레이어가 이번 라운드에서 다른 플레이어를 죽였는지 죽이지 않았는지 확인할 수 있다. |
| 4 | 장의사 거위 (Mortician Goose) |  | 라운드마다 '염습' 능력을 1 회 사용하거나 사용하지 않을 수도 있다. 염습: 죽은 플레이어 중 한 명을 선택하여, 해당 플레이어의 역할을 알아낼 수 있다. |
| 2 | 오리 (Duck) |  | 라운드마다 '살조' 능력을 1 회 사용하거나 사용하지 않을 수도 있다. 살조(殺鳥): 자신을 제외한 다른 플레이어 한 명을 죽일 수 있다. 다른 오리를 대상으로도 같은 진영이지만 대의를 위해 죽일 수 있다. 단, 해당 라운드에 이미 다른 오리들이 '라운드당 오리 살조 제한 횟수'만큼 죽였다면, 해당 라운드에서는 다른 플레이어를 죽일 수 없다. |

| | | | |
|---|---------------------------|---|--|
| 1 | 암살자 오리 (Assassin Duck) |  | <p>라운드마다 '암살' 능력과 '살조' 능력을 각각 1 회씩 사용하거나 사용하지 않을 수도 있다. 라운드마다 '암살' 능력을 사용하고 싶으면 '살조' 능력보다 먼저 사용해야만 한다. 즉, 해당 라운드에서 '살조' 능력을 먼저 사용했으면 해당 라운드에서는 '암살' 능력을 사용할 수 없다.</p> <p>암살: 한 플레이어를 선택하여 역할을 맞추면 해당 플레이어를 죽일 수 있다. 단, 선택한 플레이어의 역할을 맞추지 못했을 경우, 프라이드에 치명적인 상처를 입게되어 선택한 플레이어를 죽이지 못한채 비관하여 자살한다. 암살 능력으로 다른 플레이어를 죽였을 경우, 라운드당 오리 살조 제한 수에 포함되지 않는다. 게임이 진행되는 모든 라운드 동안 암살 능력은 2 회만 사용할 수 있다.</p> <p>살조: 오리(Duck)의 살조 능력과 동일.</p> |
| 0 | 송골매 (Falcon) |  | <p>게임당 최대 한 마리만 존재 가능하다. 다른 조류와는 다르게 투표는 무효표만 가능하다. 라운드마다 '살조' 능력을 1 회 사용하거나 사용하지 않을 수도 있다.</p> <p>'살조': 자신을 제외한 다른 조류 한 마리를 죽일 수 있다.</p> |
| 6 | 도도새 (Dodo Bird) |  | <p>게임당 최대 한 마리만 존재 가능하다. 능력이 없는 역할이다.</p> |

[프로그램 기능]

1. 게임 설정 기능

프로그램을 실행하면 그림 3 과 같은 게임 설정 메뉴가 출력되며 사용자로부터 메뉴 번호를 입력 받는다. 이때 선택 가능한 메뉴는 3 개이며, 각 메뉴에 대한 설명은 아래에서 설명된다. 1,2 번 메뉴의 기능 수행이 끝난 뒤에는 메뉴를 다시 출력하고 사용자가 다른 메뉴를 선택할 수 있도록 한다. 사용자가 잘못된 메뉴 번호를 입력 하지 않는다고 가정한다.

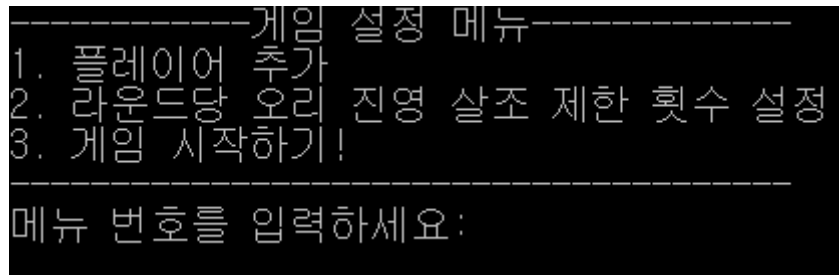


그림 3 게임 설정 메뉴 메인 화면

1.1. 플레이어 추가

“플레이어의 이름을 입력해주세요: ”를 출력하고 추가할 플레이어의 이름을 입력받는다. 플레이어의 이름은 공백이 포함되지 않은 10 자 이내의 문자열로 입력된다. 빈 문자열이나 이전에 입력되었던 문자열이 입력되지 않는다고 가정한다.

“역할 번호를 입력해주세요: ”를 출력하고 추가할 플레이어의 역할 번호를 입력받는다. 플레이어의 역할 번호는 [0,6]에 속하는 정수만 입력된다. 사용자가 송골매나 도도새를 각각 두마리 이상 추가하는 경우는 없다고 가정한다. 그림 4 참고.

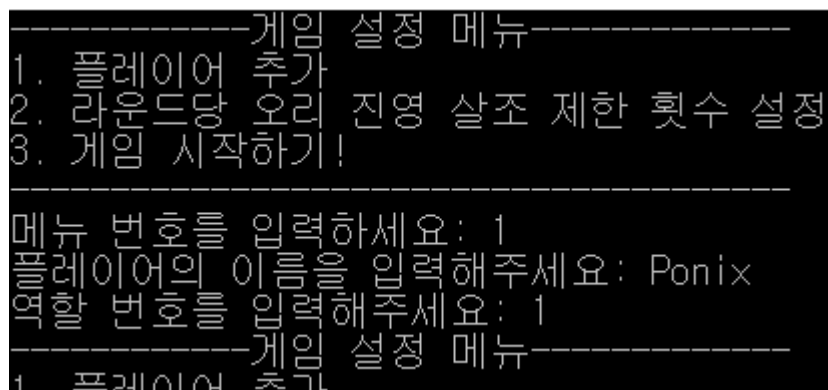


그림 4 플레이어 추가 기능

1.2. 라운드당 오리 살조 제한 횟수 변경

라운드당 오리 살조 제한 횟수의 디폴트 값은 1이다.

“값을 입력하세요: ”를 출력하고 값을 입력받는다. 입력 받은 값으로 라운드당 오리 살조 제한 횟수를 변경한다. 사용자가 양의 정수만 입력한다고 가정한다.

1.3. 게임 시작하기!

어느 진영의 승리 조건이 이미 만족되어 게임을 정상적으로 시작할 수 없는 경우와 플레이어를 한 명도 추가하지 않은 경우에는 "게임을 시작할 수 없습니다!!"를 출력한다. 그 외의 정상적인 게임 시작 가능 경우에는 "게임 시작!!"을 출력하고 게임 설정 기능을 종료한다.

2. 게임 진행 기능

매 라운드마다 "-----Round [현재 라운드(1 부터 시작)]-----"를 출력한다. 그림 5,6 참고.

| 추가 순서 | 1 | 2 | 3 |
|-------|---|---|---|
| 역할 |  |  |  |
| 이름 | Ponix | Goose1 | Goose2 |

그림 5 그림 6에 대한 게임 설정 메뉴 입력

```

-----게임 설정 메뉴-----
1. 플레이어 추가
2. 라운드당 오리 진영 살조 제한 횟수 설정
3. 게임 시작하기!
-----
메뉴 번호를 입력하세요 : 3
게임 시작!!
-----Round 1-----
Ponix님 당신은 송골매입니다.
살조를 사용하시겠습니까? [Y/N]: Y
살해하고 싶은 플레이어의 이름을 입력하세요 : Goose1
전체 메시지: Goose1이(가) 무참히 살해당하였습니다..
Goose2님 당신은 거위입니다.
당신은 능력이 없는 조류입니다.
Ponix님은 송골매이므로 무효표에 자동 투표됩니다.
Goose2님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요 : Ponix
전체 메시지: 이번 투표에서는 아무 조류도 당첨되지 않았습니다.
전체 메시지: 송골매의 승리입니다!

```

그림 6 게임 진행 기능 예시 (그림 5 입력 후 게임 진행)

2.1. 능력 사용 Phase

각 플레이어마다 자신의 차례가 오면 “[현재 차례 플레이어 이름]님 당신은 [현재 차례 플레이어 역할 이름]입니다.”를 출력한다.

능력이 있는 조류의 차례라면 “[능력 이름]을(를) 사용하시겠습니까? [Y/N]: ”를 출력하여 능력 사용 여부를 입력받는다. 사용자가 Y 를 입력하면 능력을 사용하고, N 을 입력하면 능력을 사용하지 않는다. 능력이 2 개인 암살자 오리는 암살 능력 사용 여부를 입력받고 암살 능력에 대한 처리를 모두 마치고 살조 능력 사용 여부를 입력받는다. 능력이 없는 조류의 차례라면 “당신은 능력이 없는 조류입니다.”를 출력하고 다음 차례로 넘어간다. 조사,살조,암살 능력 사용시 플레이어가 자신의 이름을 입력하거나 없는 플레이어의 이름, 사망한 플레이어의 이름을 입력하지 않는다고 가정한다.

2.1.1. 조사 (그림 7,8 참고)

“조사하고 싶은 플레이어의 이름을 입력하세요: ”를 출력하고 조사 대상 플레이어의 이름을 입력받는다. 조사 대상이 이번 라운드에 다른 플레이어를 죽였다면 “[조사 대상 플레이어의 이름]은(는) 이번 라운드에 누군가를 무참히 살해하였습니다..!”를 출력하고, 이번 라운드에 다른 플레이어를 죽이지 않았다면 “[조사 대상 플레이어의 이름]은(는) 이번 라운드에 아무도 해치지 않았습니다.”를 출력한다. 그림 7,8 참고.

2.1.2. 연습 (그림 9,10 참고)

연습은 사망한 플레이어를 지목해야하므로, 플레이어가 자신의 이름을 입력하거나 없는 플레이어의 이름, 생존한 플레이어의 이름을 입력하지 않는다고 가정한다. 사망한 플레이어가 아무도 없다면 “연습 가능 대상이 없습니다.”를 출력하고 종료한다. 사망한 플레이어가 있다면 “연습하고 싶은 플레이어의 이름을 입력하세요: ”를 출력하고 연습 대상 플레이어의 이름을 입력받는다. “[연습 대상 플레이어의 이름]의 역할은 [역할 이름]입니다.”를 출력하고 종료한다. 그림 9,10,11 참고.

2.1.3. 살조(오리)

라운드당 오리 진영의 살조 제한 횟수에 이미 도달하였다면, “라운드당 오리 진영의 살조 제한 횟수에 도달하였습니다.”를 출력하고 종료한다. 제한 횟수에 도달하지 않았다면 “살해하고 싶은 플레이어의 이름을 입력하세요: ”를 출력하고 살해 대상 플레이어의 이름을 입력받는다. “전체 메시지: [살해 대상 플레이어의 이름]이(가) 무참히 살해당하였습니다..”를 출력하고 종료한다.

2.1.4. 암살 (그림 11,12,13 참고)

해당 암살자 오리의 게임당 암살 가능 횟수에 이미 도달하였다면, “더 이상 암살 능력을 사용할 수 없습니다!”를 출력하고 종료한다. 게임당 암살 가능 횟수에 도달하였다면 “암살하고 싶은 플레이어의 이름을 입력하세요: ”를 출력하고 암살 대상 플레이어의 이름을 입력받는다. “암살하고 싶은 플레이어의 역할 번호를 입력하세요: ”를 출력하고 암살 대상 플레이어의 예상 역할 번호를 입력받는다. 역할

번호가 맞는 경우 "전체 메시지: [살해 대상 플레이어의 이름]이(가) 무참히 살해당하였습니다.."를 출력하고 암살 대상 플레이어를 사망 처리하고 종료한다. 역할 번호가 틀릴 경우 "전체 메시지: [현재 차례 암살자 오리 플레이어의 이름]은(는) 극단적 선택을 하였습니다."를 출력하고 현재 차례인 암살자 오리 플레이어를 사망 처리하고 종료한다. 그림 12,13,14 참고.

2.1.5. 살조(송골매)

"살해하고 싶은 플레이어의 이름을 입력하세요: "를 출력하고 살해 대상 플레이어의 이름을 입력받는다. "전체 메시지: [살해 대상 플레이어의 이름]이(가) 무참히 살해당하였습니다.."를 출력하고 종료한다.

2.2. 투표 Phase

순서대로 투표를 진행한다. 플레이어가 없는 플레이어의 이름, 사망한 플레이어의 이름을 입력하지 않는다고 가정한다.

2.2.1. 송골매 제외

"[현재 차례 플레이어 이름]님 투표를 하시겠습니까? [Y/N]: "를 출력하여 투표 참여 여부를 입력받는다. 사용자가 N 을 입력하면 무효표로 처리하고, Y 를 입력하면 "투표하고 싶은 플레이어의 이름을 입력하세요: "를 출력하여 투표 대상 플레이어 이름을 입력받는다.

2.2.2. 송골매

"[현재 차례 플레이어 이름]님은 송골매이므로 무효표에 자동 투표됩니다."를 출력한다.

2.2.3. 추방당하는 플레이어가 있을 경우에는 "전체 메시지: [추방당한 플레이어의 이름]은(는) 더 좋은 곳을 갔습니다."를 출력하고 해당 플레이어를 사망 처리한다. 추방당한 플레이어가 오리일 경우 "전체 메시지: 만세 [추방당한 플레이어의 이름]은(는) 오리입니다!!"를 출력하고, 오리가 아닌 경우 "전체 메시지: 맙소사 [추방당한 플레이어의 이름]은(는) 오리가 아닙니다!!"를 출력한다.

2.2.4. 추방당하는 플레이어가 없을 경우에는 "전체 메시지: 이번 투표에서는 아무 조류도 당첨되지 않았습니다."를 출력한다.

2.3. 승리 조건 달성 여부 판단

라운드 종료마다 승리한 진영이 있을 경우 게임 결과를 출력하고 게임을 종료한다. 승리한 진영이 없을 경우 다음 라운드를 진행한다.

2.3.1. 거위 승리: "전체 메시지: 거위의 승리입니다!"를 출력하고 게임을 종료한다.

2.3.2. 오리 승리: "전체 메시지: 오리의 승리입니다!"를 출력하고 게임을 종료한다.

2.3.3. 송골매 승리: "전체 메시지: 송골매의 승리입니다!"를 출력하고 게임을 종료한다.

2.3.4. 도도새 승리: "전체 메시지: 도도새의 승리입니다!"를 출력하고 게임을 종료한다.

추가 예시

2.4. 조사 예시 (그림 7,8)

| 추가 순서 | 1 | 2 | 3 |
|-------|---|---|---|
| 역할 |  |  |  |
| 이름 | Goose | Duck | DetectiveG |

그림 7 그림 8에 대한 게임 설정 메뉴 입력

```

-----Round 1-----
Duck님 당신은 오리입니다.
살조를 사용하시겠습니까? [Y/N]: N
DetectiveG님 당신은 탐정 거위입니다.
조사를 사용하시겠습니까? [Y/N]: Y
조사하고 싶은 플레이어의 이름을 입력하세요 : Duck
Duck은(는) 이번 라운드에 아무도 해치지 않았습니다.
Goose님 당신은 거위입니다.
당신은 능력이 없는 조류입니다.
Goose님 투표를 하시겠습니까? [Y/N]: N
Duck님 투표를 하시겠습니까? [Y/N]: N
DetectiveG님 투표를 하시겠습니까? [Y/N]: N
전체 메시지: 이번 투표에서는 아무 조류도 당첨되지 않았습니다.
-----Round 2-----
Duck님 당신은 오리입니다.
살조를 사용하시겠습니까? [Y/N]: Y
살해하고 싶은 플레이어의 이름을 입력하세요 : Goose
전체 메시지: Goose이(가) 무참히 살해당하였습니다..
DetectiveG님 당신은 탐정 거위입니다.
조사를 사용하시겠습니까? [Y/N]: Y
조사하고 싶은 플레이어의 이름을 입력하세요 : Duck
Duck은(는) 이번 라운드에 누군가를 무참히 살해하였습니다..!
Duck님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요 : DetectiveG
DetectiveG님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요 : Duck
전체 메시지: 이번 투표에서는 아무 조류도 당첨되지 않았습니다.
전체 메시지: 오리의 승리입니다!
  
```

그림 8 게임 진행 예시 (조사)

2.5. 연습 예시 (그림 9,10)

| 추가 순서 | 1 | 2 | 3 |
|-------|---|--|---|
| 역할 |  |  |  |
| 이름 | Goose | Duck | MorticianG |

그림 9 그림 10에 대한 게임 설정 메뉴 입력

```

-----Round 1-----
Duck님 당신은 오리입니다.
살조를 사용하시겠습니까? [Y/N]: N
MorticianG님 당신은 장의사 거위입니다.
연습을 사용하시겠습니까? [Y/N]: Y
연습 가능 대상이 없습니다.
Goose님 당신은 거위입니다.
당신은 능력이 없는 조류입니다.
Goose님 투표를 하시겠습니까? [Y/N]: N
Duck님 투표를 하시겠습니까? [Y/N]: N
MorticianG님 투표를 하시겠습니까? [Y/N]: N
전체 메시지: 이번 투표에서는 아무 조류도 당첨되지 않았습니다.
-----Round 2-----
Duck님 당신은 오리입니다.
살조를 사용하시겠습니까? [Y/N]: Y
살해하고 싶은 플레이어의 이름을 입력하세요: Goose
전체 메시지: Goose이(가) 무참히 살해당하였습니다..
MorticianG님 당신은 장의사 거위입니다.
연습을 사용하시겠습니까? [Y/N]: Y
연습하고 싶은 플레이어의 이름을 입력하세요: Goose
Goose의 역할은 거위입니다.
Duck님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: MorticianG
MorticianG님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: Duck
전체 메시지: 이번 투표에서는 아무 조류도 당첨되지 않았습니다.
전체 메시지: 오리의 승리입니다!

```

그림 10 게임 진행 예시 (연습)

2.6. 암살 예시

| 추가 순서 | 1 | 2 | 3 | 4 |
|-------|---|---|--|---|
| 역할 |  |  |  |  |
| 이름 | Goose1 | Dodo | AssassinD | MorticianG |

그림 11 그림 12와 그림 13에 대한 게임 설정 메뉴 입력

2.6.1. 암살 성공 (그림 11,12)

```

-----Round 1-----
AssassinD님 당신은 암살자 오리입니다.
암살을 하시겠습니까? [Y/N]: Y
암살하고 싶은 플레이어의 이름을 입력하세요: MorticianG
암살하고 싶은 플레이어의 역할 번호를 입력하세요: 4
전체 메시지: MorticianG이(가) 무참히 살해당하였습니다..
살조를 하시겠습니까? [Y/N]: N
Goose님 당신은 거위입니다.
당신은 능력이 없는 조류입니다.
Dodo님 당신은 도도새입니다.
당신은 능력이 없는 조류입니다.
Goose님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: AssassinD
Dodo님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: AssassinD
AssassinD님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: Goose1
전체 메시지: AssassinD은(는) 더 좋은 곳으로 갔습니다.
전체 메시지: 만세 AssassinD은(는) 오리입니다!!
전체 메시지: 거위의 승리입니다!
  
```

그림 12 게임 진행 예시 (암살 성공)

2.6.2. 암살 실패 (그림 11,13)

```
-----Round 1-----
AssassinD님 당신은 암살자 오리입니다.
암살을 사용하시겠습니까? [Y/N]: Y
암살하고 싶은 플레이어의 이름을 입력하세요: MorticianG
암살하고 싶은 플레이어의 역할 번호를 입력하세요: 5
전체 메시지: AssassinD은(는) 극단적 선택을 하였습니다.
MorticianG님 당신은 장의사 거위입니다.
염습을 사용하시겠습니까? [Y/N]: Y
염습하고 싶은 플레이어의 이름을 입력하세요: AssassinD
AssassinD의 역할은 암살자 오리입니다.
Goose님 당신은 거위입니다.
당신은 능력이 없는 조류입니다.
Dodo님 당신은 도도새입니다.
당신은 능력이 없는 조류입니다.
Goose님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: Dodo
Dodo님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: Dodo
MorticianG님 투표를 하시겠습니까? [Y/N]: Y
투표하고 싶은 플레이어의 이름을 입력하세요: Dodo
전체 메시지: Dodo은(는) 더 좋은 곳으로 갔습니다.
전체 메시지: 맙소사 Dodo은(는) 오리가 아닙니다!!
전체 메시지: 도도새의 승리입니다!
```

그림 13 게임 진행 예시 (암살 실패)

[Class 및 기타 파일 설명]

1. main.cpp (main 함수 파일)

첨부파일로 제공된 main 함수를 사용하면 됩니다. **첨부파일로 제공된 main 함수는 수정하지 마세요. 다른 파일들은 기존 내용들을 수정 및 삭제하셔도 괜찮습니다.**

2. Macro.h 파일

사용하고 싶은 매크로나 전역 상수는 이 파일에만 정의해주세요. 전역 변수는 이번 과제에서 사용하지 않습니다. 이번 과제에서 사용자 입력 예외 처리는 최대한 제외해드렸으니 전역 변수를 사용하지 않고 코딩하는 연습을 해보도록 합니다.

3. Bird 클래스 및 그외 조류(역할) 클래스

그림 14 를 참고하여 class inheritance 를 이용하여 조류 클래스들을 구현해봅시다. 클래스 및 멤버 함수 선언은 *.h 파일에, 정의는 *.cpp 파일에 해주세요.

3.1. Bird 클래스

abstract 클래스입니다. **최소 한 개의 순수 추상 함수(pure virtual function)를 포함하고 있어야 합니다.** (ex) virtual void func() = 0;) 순수 추상 함수가 없이 구현된 제출물은 0 점입니다.

3.2. 그외 조류(역할) 클래스

각 역할마다 별개의 파일로 구현해주세요. class inheritance 를 활용하여 구현해보세요.

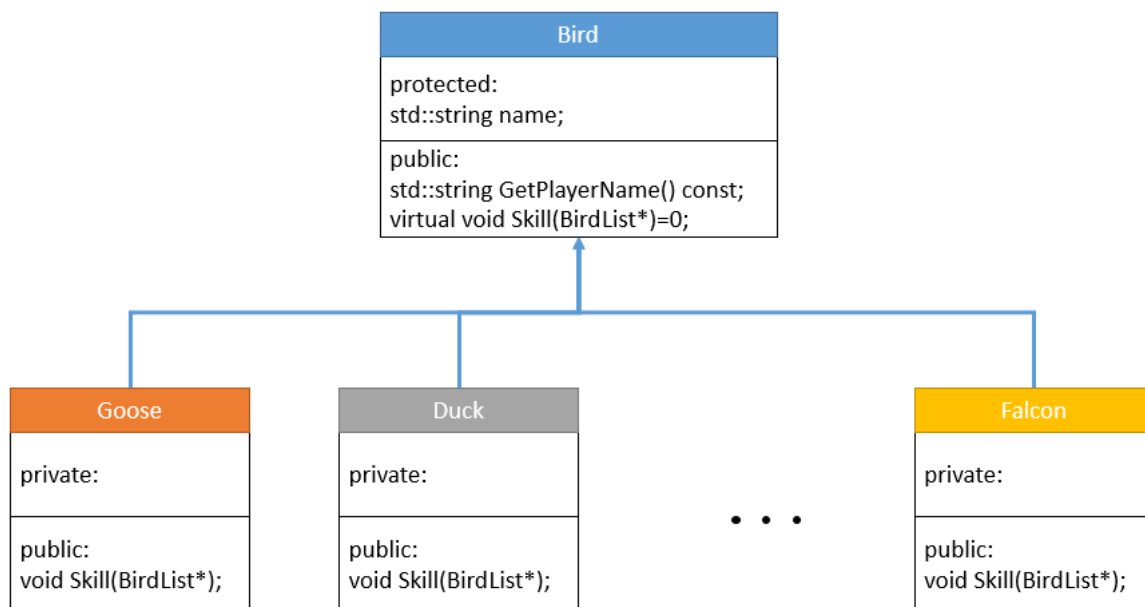


그림 14 Bird 및 조류(역할) 클래스 관계도 예시

4. GGD 클래스

메뉴 화면을 포함하여 게임을 제어하는 클래스입니다. main 함수에서 하나의 Object 만 사용됩니다. 아래 함수들은 제공된 파일에서 함수 선언부를 수정하시면 안됩니다.

4.1. void GameStart()

게임 설정 기능을 구현하시면 됩니다.

4.2. bool IsGameOver()

함수 호출 시점에서 승리한 진영이 있을 경우 true 를 반환하고 승리한 진영이 없을 경우 false 를 반환하면 됩니다.

4.3. void RoundProgress()

게임 진행 기능 중 능력 사용 Phase 와 투표 Phase 를 구현하면 됩니다. 모든 내용을 이 함수에 직접 구현하는 것이 아닌 BirdList 와 조류 클래스에 나눠 구현하고 그것들을 호출하는 방식으로 구현해보는 것이 출제 의도 입니다. GGD 클래스의 다른 멤버 함수들도 마찬가지입니다.

4.4. void PrintGameResult()

게임 최종 결과를 출력하는 함수입니다.

5. BirdList, BirdNode 클래스

5.1. BirdNode 는 완성된 헤더 파일과 cpp 파일이 제공됩니다. 기존 내용을 수정하시거나 본인 코드를 추가하셔서 사용하셔도 좋습니다.

5.2. STL 대신 Linked list 에 플레이어들을 저장하는 방식으로 구현해봅시다.

[세부조건]

Class inheritance 를 실습해보는 것이 목표인 문제입니다. 그림 14 를 참고하여 Report 에 class inheritance 에 대한 내용을 서술해주세요.

아래의 주의사항들을 꼼꼼히 읽어보시고 잘 지켜주세요.

1. class inheritance 로 구현하지 않은 조류 class 는 0 점입니다.
2. 전역 변수는 사용 금지입니다. 매크로나 전역 상수는 사용 가능합니다.
3. 사용가능 헤더 파일: <iostream>, <string>
4. 각 클래스마다 header file 과 cpp file 을 나누어 구현하세요. 클래스마다 추가로 사용하고 싶은 변수나 함수가 있다면 추가해도 됩니다. 다만, 멤버 변수는 protected 혹은 private 으로 한정해주세요.
5. 문제에 명시되어 있지 않더라도 각 클래스마다 생성자, 소멸자는 필수입니다. 기본 생성자의 경우, 별다른 언급이 없다면 멤버 변수에 int 는 0, string 은 "", pointer 는 NULL 로 초기화 해주세요.