

파이널 프로젝트

# 플젝의 목표

C++의 주요 기능들을 응용에서 게임 혹은 시뮬레이션을 만드는 것.

주요 조건:

- 게임 혹은 시뮬레이션 (다른 분야도 상의 하에 가능)
- 1~2명에서 개발 (파트너를 구해도 괜찮습니다)
- Github에 업로드
- 키워드: "무한"
  - Dynamic memory allocation을 최대한 활용할 수 있는 걸로
  - 예시) 무한 지뢰찾기, Game of Life, 입자 시뮬레이션 등등...

# 추천 플젝 규모

- 무한 지뢰찾기

꽤 좋은 플젝인 이유는

- 간단하게 하자면 적당히 쉽고,
- 효율적으로 돌아가게 하자면 꽤 어려워진다.
  - 메모리 관리를 어떻게 하느냐에 따라서...

무한 지뢰찾기의 스켈레톤 코드를 올릴테니 참고하실 분은 참고하시길...

# 키워드를 반드시 지킬 필요는 없고

개인 역량에 맞춰서 아무런 게임이나 만들어도 괜찮습니다. 최대한 많은 사람이 이수하도록 하고 싶습니다...

다른 게임 아이디어들:

- Snake
- 벽돌깨기
- Space Invaders
- Doodle Jump
- 2048

◦ 이 리스트에서 가장 쉬운 편이지만... 애니메이션을 넣는다면 아닐지도!

# SFML

Example: <https://www.sfml-dev.org/documentation/2.4.2/>

# SFML 사용 조언

- 인터넷에 사람들이 만들어 놓은 게임이나 엔진 코드를 참고해도 괜찮다
  - 아예 Ctrl-C Ctrl-V한거만 아니면...
- SFML 사용을 위한 자료들 참고:
  - SFML Tutorial: <https://www.sfml-dev.org/tutorials/2.4/>
  - SFML Documentation: <https://www.sfml-dev.org/documentation/2.4.2/>
  - SFML 관련된 유틸리티 코드 모음: <https://github.com/SFML/SFML/wiki>
  - Thor 유틸리티 라이브러리 <http://www.bromeon.ch/libraries/thor/index.html>

# SFML 사용 조언

게임 속 리소스들 (텍스처, 사운드, 폰트 등)을 관리하는 Resource Manager가 있으면 편하다. (다음은 대충 짠 예시 코드)

```
class ResourceManager {
public:
    shared_ptr<sf::Texture> loadTexture(const string& filename,
                                       const string& name) {
        auto texture = make_shared<sf::Texture>();
        texture->loadFromFile(filename);
        textures[name] = texture;
    }

    shared_ptr<sf::Texture> getTexture(const string& name) {
        auto foundIt = textures.find(name);
        if (foundIt == textures.end()) {
            cerr << "Cannot find texture " << name << "." << endl;
            return nullptr;
        }
        else {
            return *foundIt;
        }
    }

private:
    unordered_map<string, shared_ptr<sf::Texture>> textures;
}
```

# SFML 사용 조언

반복되는 텍스처를 여러번 렌더링 할 때에는 Sprite보다 Vertex Array를 쓰는 것이 훨씬 빠르다.

무한 지뢰찾기처럼 거대한 타일 기반 게임일 경우 참고.

<https://www.sfml-dev.org/tutorials/2.4/graphics-vertex-array.php>

만약 이 기능을 쓰고 싶는데 이해가 잘 안된다면 저에게 오세요



# 데드라인

주제 선정 및 Github repo 생성: 1월 6일까지

끄으을?