피카츄배구 분석하기



README.md

발표자 : 오지호

목차

뭐 하나요?

DLL파일이 뭔가요?

DLL injection

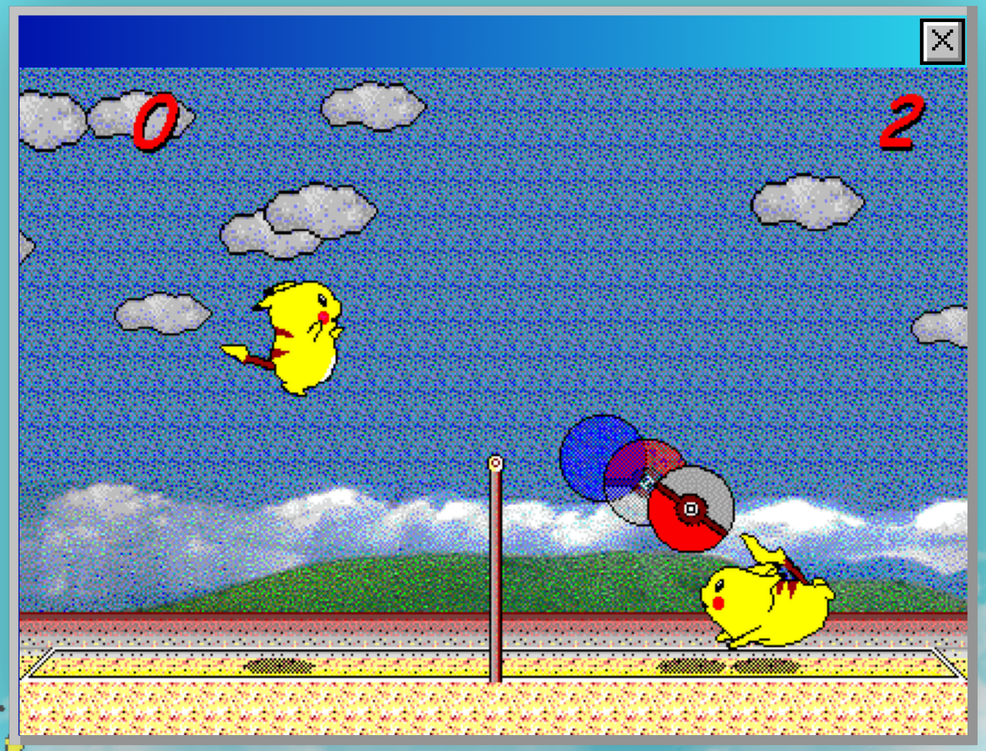
피카츄 배구가 뭔가요?

피카츄 배구 분석

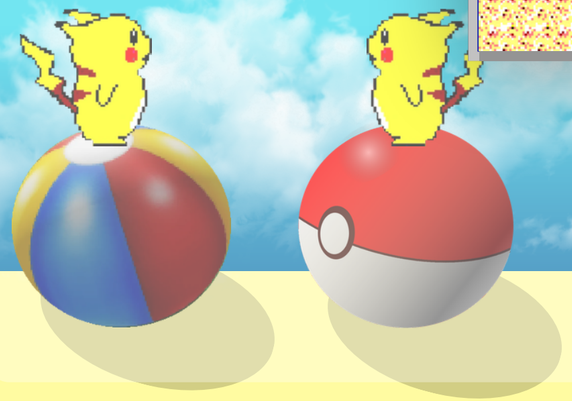
마무리



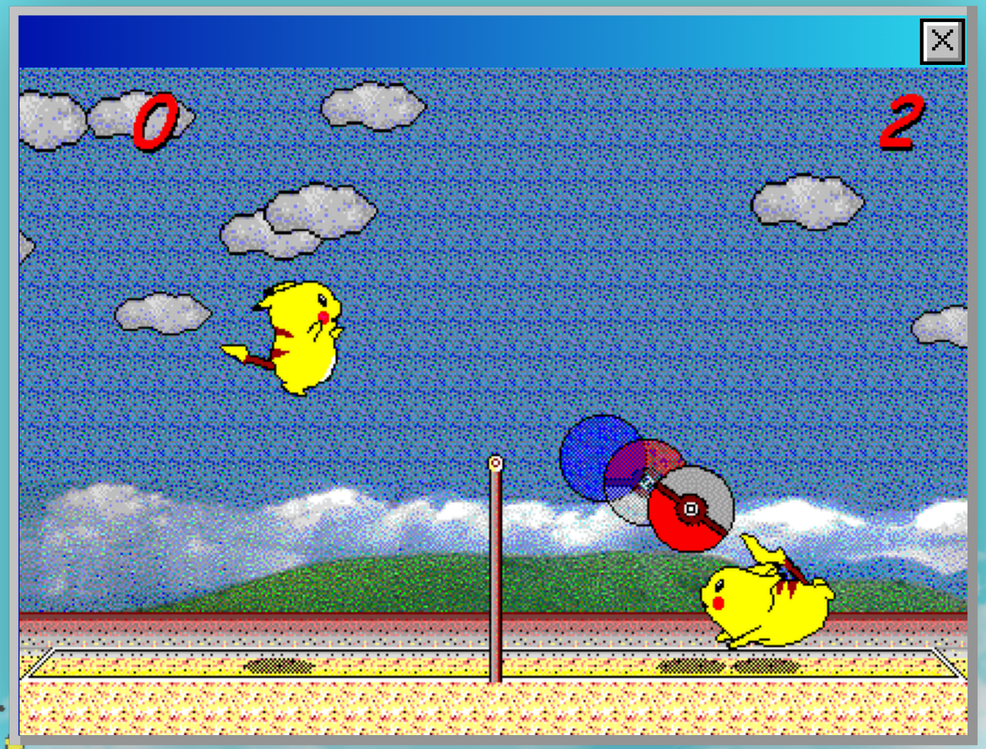
뭐 하나요?



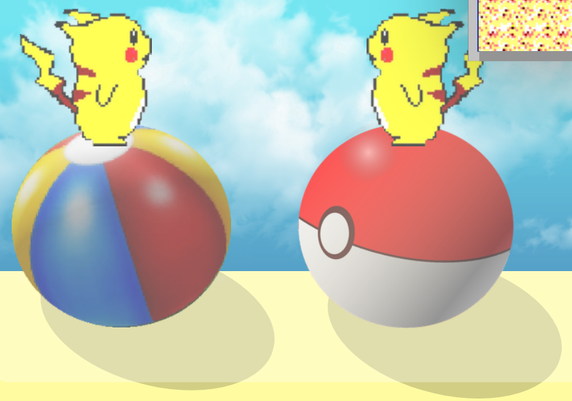
<< 이거 분석



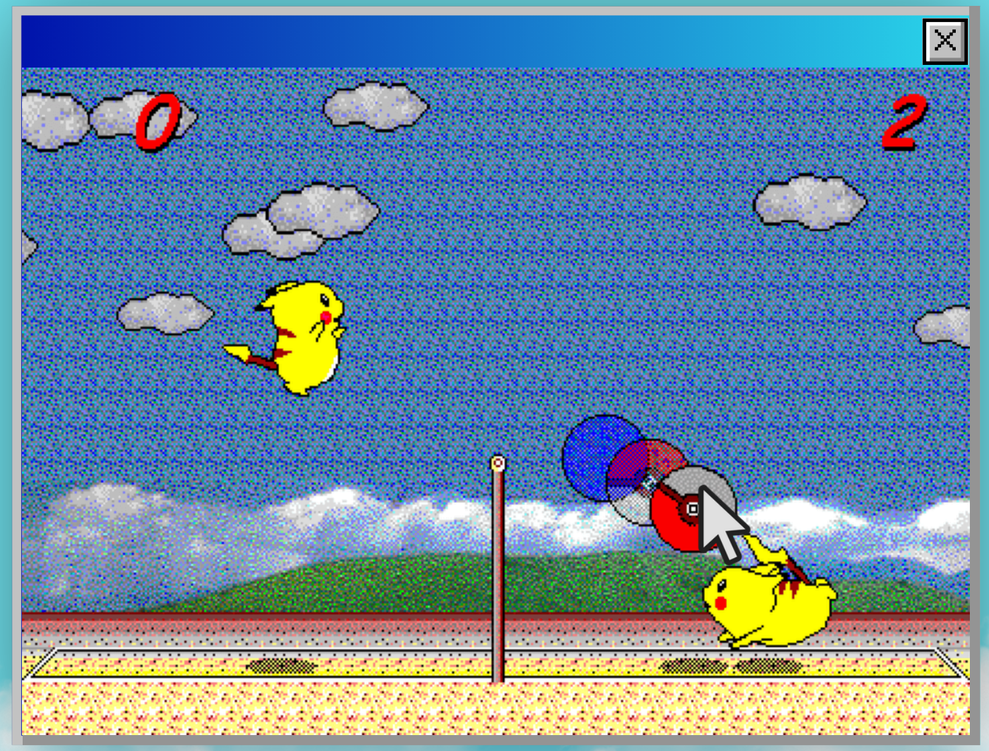
뭐 하나요?



이거만 하면
좀 그러니까



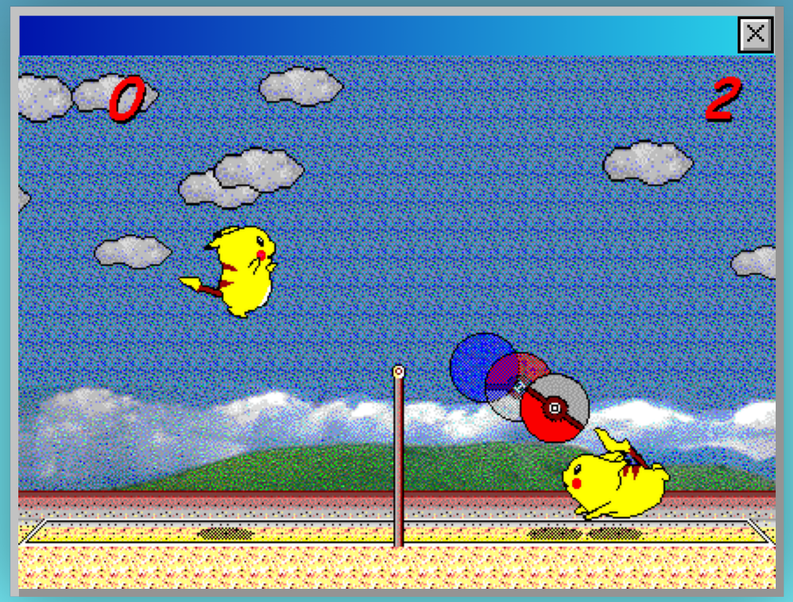
뭐 하나요?



마우스 포인터로
공 움직이기



피카츄 배구



피카츄가 배구함



DLL 파일이 뭔가요?

windows 운영체제 내에서
쓰이는 코드 묶음

동적 라이브러리



DLL 파일이 뭔가요?

필요할때 불러와서 쓸수있는





DLL injection

ㅇㅇ 필요하니까 오세요



네



일반적으로 프로세스에서 불러옴



DLL injection

네



얘도 써주세요



injector



injector가 끼워 넣음



피카츄 배구 분석



피카츄 배구 분석

main함수 → 이벤트 처리 → 게임 이벤트 처리 → 공 물리엔진



피카츄 배구 분석

```
int RunMessageLoop()
{
    MSG message;

    for (;;) {
        while (!PeekMessageA(&message, nullptr, 0, 0, PM_NOREMOVE)) {
            if (g_game_app != nullptr)
                GameApp_Tick(g_game_app);
        }

        if (!GetMessageA(&message, nullptr, 0, 0))
            return 0;

        TranslateMessage(&message);
        DispatchMessageA(&message);
    }
}
```

이벤트 처리 함수

피카츄 배구 분석

```
int Ball_UpdateWorldCollision(Ball* ball)
{
    ball->spinAccumulator += ball->vx / 2;
    if (ball->spinAccumulator < 0)       ball->spinAccumulator += 50;
    else if (ball->spinAccumulator > 50) ball->spinAccumulator -= 50;
    ball->animationFrame = ball->spinAccumulator / 10;

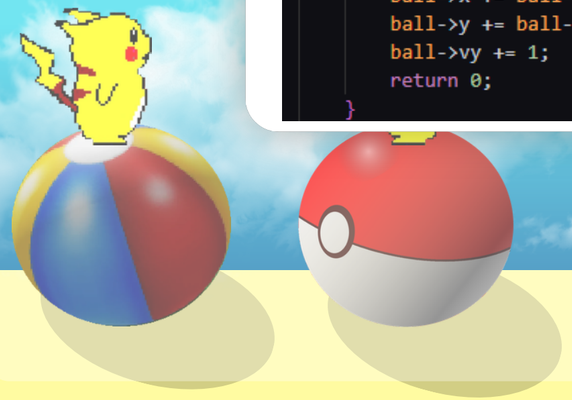
    if (static_cast<unsigned>(ball->x + ball->vx - 20) >= 413)
        ball->vx = -ball->vx;

    if (ball->y + ball->vy < 0)
        ball->vy = 1;

    if (std::abs(ball->x - 216) <= 24 && ball->y > 176) {
        if (ball->y <= 192) {
            if (ball->vy > 0)
                ball->vy = -ball->vy;
        } else {
            ball->vx = ball->x < 216 ? -std::abs(ball->vx)
                                     :  std::abs(ball->vx);
        }
    }

    if (ball->y + ball->vy <= 252) {
        ball->x += ball->vx;
        ball->y += ball->vy;
        ball->vy += 1;
        return 0;
    }
```

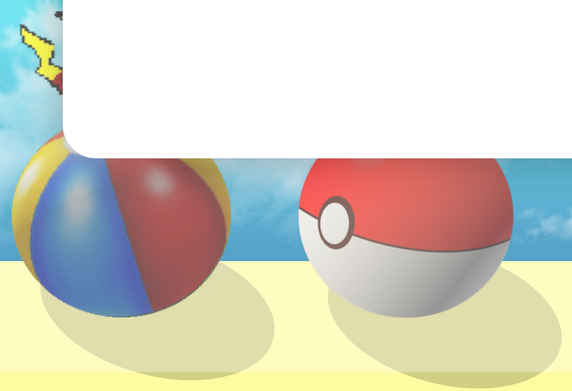
잘 타고 내려와서
공 이랑 뭐 물리엔진처리 부분 까지
온거 같음



피카츄 배구 분석

```
if (ball->y + ball->vy <= 252) {
    ball->x += ball->vx;
    ball->y += ball->vy;
    ball->vy += 1;
    return 0;
}
```

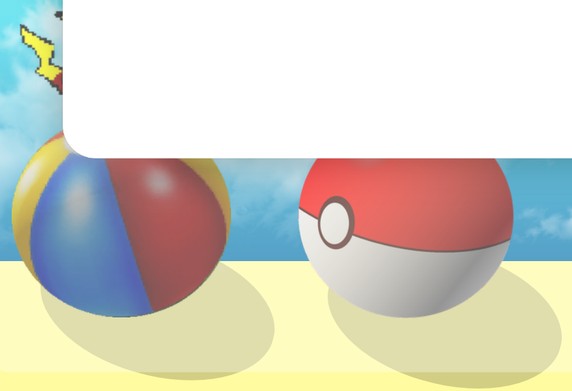
코드를 보고 걸러내보니
이 부분만 보면 될거같음



피카츄 배구 분석

```
if (ball->y + ball->vy <= 252) {
    ball->x += ball->vx;
    ball->y += ball->vy;
    ball->vy += 1;
    return 0;
}
```

잘 분석을 해보니
바닥을 넘는지 확인
가로 위치에 가로 속도 더하기
세로 위치에 세로 속도 더하기
중력 표현
이런것들임



피카츄 배구 분석

방금 전에 분석한 자료들을 가지고
DLL injection으로 하려고 했던거 하기



피카츄 배구 분석

```cpp
bool InjectLoadLibrary(HANDLE process, const std::wstring& dll_path) {
    auto load_library = reinterpret_cast<LPTHREAD_START_ROUTINE>(
        GetProcAddress(GetModuleHandleW(L"kernel32.dll"), "LoadLibraryW"));
    const SIZE_T size = (dll_path.size() + 1) * sizeof(wchar_t);
    void* path = VirtualAllocEx(process, nullptr, size,
                                MEM_COMMIT | MEM_RESERVE, PAGE_READWRITE);
    if (load_library == nullptr || path == nullptr)
        return false;
    if (!WriteProcessMemory(process, path, dll_path.c_str(), size, nullptr)) {
        VirtualFreeEx(process, path, 0, MEM_RELEASE);
        return false;
    }
    ScopedHandle thread(CreateRemoteThread(process, nullptr, 0, load_library, path, 0, nullptr));
    if (thread.get() == nullptr) {
        VirtualFreeEx(process, path, 0, MEM_RELEASE);
        return false;
    }
    WaitForSingleObject(thread.get(), INFINITE);
    DWORD module = 0;
    const bool loaded = GetExitCodeThread(thread.get(), &module) != FALSE && module != 0;
    VirtualFreeEx(process, path, 0, MEM_RELEASE);
    return loaded;
}
```

injector코드
피카츄 배구 메모리에 DLL 경로 적기
하고 DLL파일 불러오기



DLL 파일 만들기



피카츄 배구 분석

```cpp
std::uintptr_t FindBall() {
    const auto base = reinterpret_cast<std::uintptr_t>(GetModuleHandleW(nullptr));
    std::uintptr_t pointer = Read<std::uintptr_t>(base + kGameRoot);
    if (!pointer) return 0;

    pointer = Read<std::uintptr_t>(pointer + 0x28);
    if (!pointer) return 0;

    pointer = Read<std::uintptr_t>(pointer + 0x10);
    if (!pointer) return 0;

    return Read<std::uintptr_t>(pointer + 0x14);
}
```

공의 위치 확인
기준 잡기



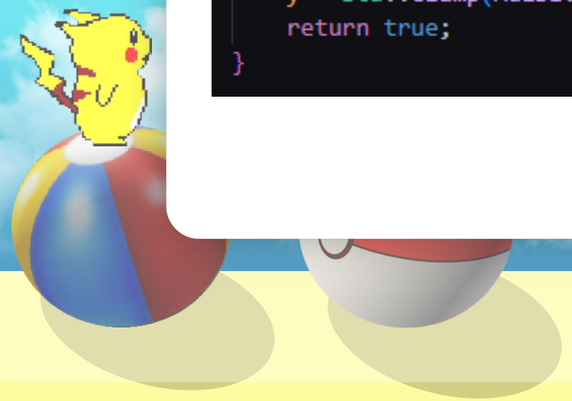
피카츄 배구 분석

```cpp
bool GetMouseInGame(HWND window, int& x, int& y) {
    POINT mouse{};
    RECT client{};
    if (!GetCursorPos(&mouse) || !ScreenToClient(window, &mouse) ||
        !GetClientRect(window, &client)) {
        return false;
    }

    const int width = client.right - client.left;
    const int height = client.bottom - client.top;
    if (width <= 0 || height <= 0 || mouse.x < 0 || mouse.y < 0 ||
        mouse.x >= width || mouse.y >= height) {
        return false;
    }

    x = std::clamp(MulDiv(mouse.x, kGameWidth, width), 20, 412);
    y = std::clamp(MulDiv(mouse.y, kGameHeight, height), 0, 232);
    return true;
}
```

기준 위치 잡은걸로 계산하고
마우스 위치는 게임 좌표로 변환



피카츄 배구 분석

```
void MoveBall(std::uintptr_t ball, int x, int y) {
    Write<int>(ball + 0x30, x);
    Write<int>(ball + 0x34, y);
    Write<int>(ball + 0x38, 0);
    Write<int>(ball + 0x3C, 0);
    Write<int>(ball + 0x58, x);
    Write<int>(ball + 0x5C, x);
    Write<int>(ball + 0x60, y);
    Write<int>(ball + 0x64, y);
}
```

전달 받은 마우스 좌표로 공 위치 변경

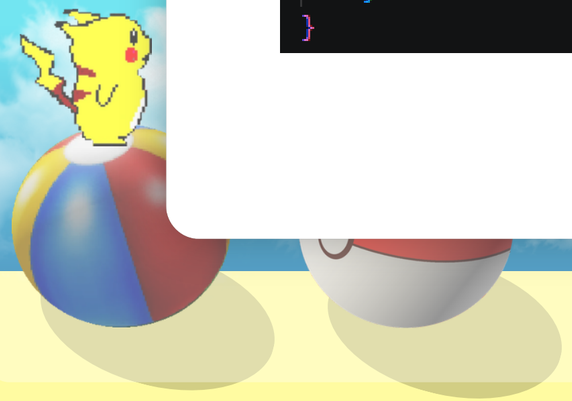


피카츄 배구 분석

```
DWORD WINAPI MouseControl(void*) {
    while (true) {
        HWND window = GetForegroundWindow();
        DWORD processId = 0;
        GetWindowThreadProcessId(window, &processId);

        int mouseX = 0;
        int mouseY = 0;
        if (processId == GetCurrentProcessId() &&
            (GetAsyncKeyState(VK_LBUTTON) & 0x8000) &&
            GetMouseInGame(window, mouseX, mouseY)) {
            const std::uintptr_t ball = FindBall();
            if (ball) MoveBall(ball, mouseX, mouseY);
        }
        Sleep(8);
    }
}
```

마우스 드래그 반복 확인
인식 하는 부분



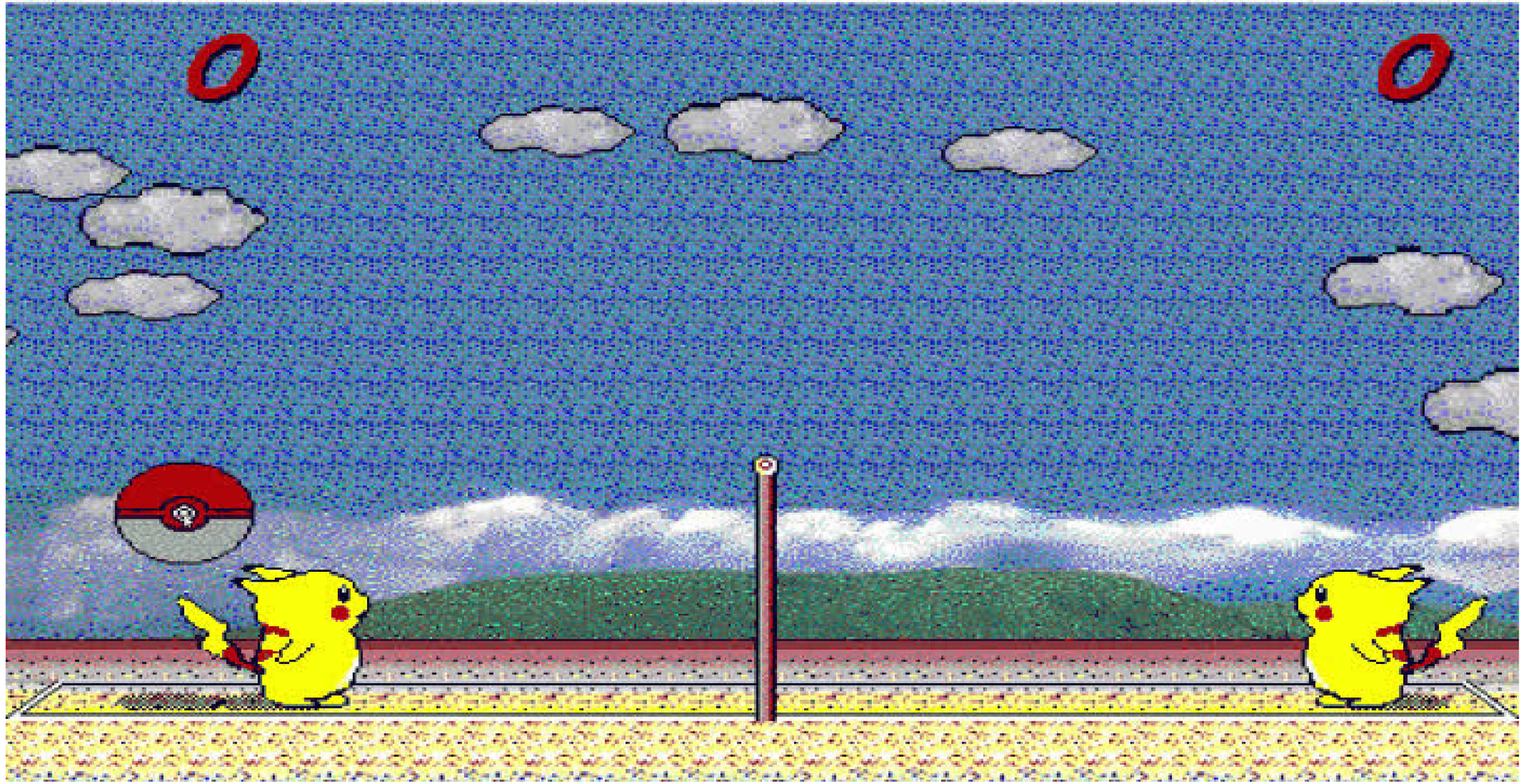
피카츄 배구 분석

```
BOOL WINAPI DllMain(HINSTANCE dll, DWORD reason, LPVOID) {
    if (reason == DLL_PROCESS_ATTACH) {
        DisableThreadLibraryCalls(dll);
        CloseHandle(CreateThread(nullptr, 0, MouseControl, nullptr, 0, nullptr));
    }
    return TRUE;
}
```

메인



마무리



발표자 : 오지호

감사합니다

