

[Toy Ground(토이 그라운드)]

12 주	2020. 3. 14 ~ 2021. 3. 20	작성자	김동석
이번 주 한 일	<p>[0] 공동 03.15 주간 회의 - 일주일간 계획공유 - 지금까지 진행 상황 공유 - 3, 4월 월간 목표 설정</p> <p>[1] 김영준(서버) 주간목표: - 공유자원 개념 이해 - 게임 서버 프로그래밍 강의 수강</p> <p>*사진은 모두 notion을 통해 스스로 공부한 내용을 정리한 것을 캡처했습니다.</p> <p>03.15~17 스레드 동기화 스레드간 동기화가 이루어져야 하는 부분(충돌처리, 캐릭터 이동처리, 총알 이동처리, 맵 구조 처리 등)에 대한 설계가 부족하여 정리 및 설계를 진행 (system call이 아닌 공유자원으로 동기화 진행) data race가 발생함에 따라 동기화에 대한 생각을 먼저 했어야 했는데 저번주 개발 내용과 순서가 잘못되었다고 생각함</p> <p>03.18 게임엔진1 팀 프로젝트로 인한 개발 지연/공유 메모리에 대한 공부</p>		

PC에서의 공유 메모리

- 다른 코어에서 보았을 때 업데이트 순서가 틀릴 수 있음
- 메모리의 내용이 한 순간에 업데이트 되지 않을 때도 있음
- 언젠가는 메모리에 대한 쓰기가 실행됨
- 자기 자신의 프로그램 순서는 지켜짐
- 캐시의 일관성은 지켜짐(한번 지워진 값이 살아나진 않음, 언젠가 모든 코어가 동일한 값을 봄)
- 캐시라인 내부의 쓰기는 중간값을 만들지 않음

해결책

- 공유 메모리, 메모리 일관성에 대한 상황을 고려하면서 프로그래밍 한다(불가능)
- lock을 통해 순서의 어긋남을 제거함(성능저하, lock의 구현?)
- 문제가 되는 어긋남을 제거(atomic_thread_fence)사용→어렵다, 남발 시 성능 저하)
- 기존의 멀티쓰레드용 라이브러리를 사용(C++11 <atomic>)

03.19 보간이 필요한 게임 요소에 대한 설계

총알의 이동, 보물의 생성(물리 요소 有)에 대해서 물리 이동에 대한 값을 매번 연산하고 클라이언트에 넘겨주는 것에서 불필요한(우선순위가 낮은) 연산이 많이 일어난다고 생각함

총알, 보물 생성, 오브젝트 좌표, 각도 변경(나무가 쓰러짐)에 대한 물리 연산은 보간을 통해 패킷을 절약할 수 있도록 설계함

03.20 volatile, FAR* 등 게임 서버 수업 의문점 공부

volatile: 항상 메모리에 접근하여 변수를 확인

(volatile이 아니면 레지스터에 등록하여 항상 확인하지 않을 수 있음)

```
volatile int * a; //volatile int를 가르키는 "포인터"
*a = 1; //최적화로 인한 오동작 없음
a = b; //최적화로 인한 메모리 접근 생략, 순서 변경(오동작 가능)
int * volatile a; //int를 가르키는 "volatile 포인터"
*a = 1; //최적화로 인한 메모리 접근 생략, 순서 변경(오동작 가능)
a = b; //최적화로 인한 오동작 없음
```

디스어셈블리를 통해 확인한 메모리에 반드시 정해진 순서로 읽고 쓰는 volatile 키워드를 통해 멀티 스레드에서의 안정성에 대해서 공부 (결국 atomic을 사용)

C++ 코드를 읽으며 가끔 등장하는 FAR*을 명시적으로 작성하는 코드가 있는데 검색을 해봐도 이전 16bit OS에서 1MB 메모리를 할당하기 위해 사용된 오래된 키워드로 나오는데 구조체 선언에 등장하는 FAR*

의 의미를 찾지 못해 궁금하다.

[2] 김동석(클라이언트)

주간 목표 :

- 프레임워크 마무리(90%)
- 서버와 연동 준비(진행 중)
- Scene Controller

03.16 Character

- Character Move 수정

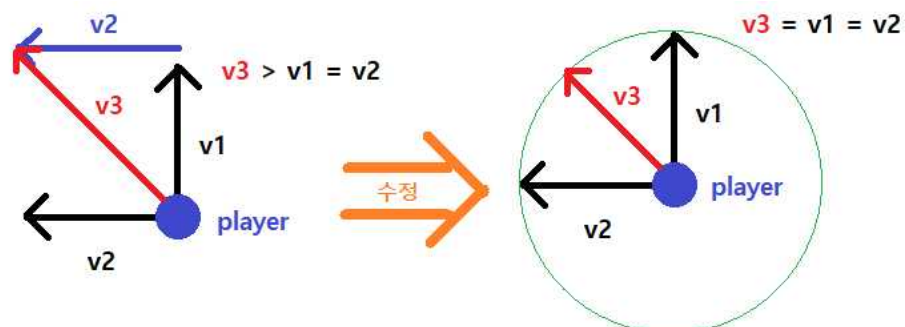
플레이어가 움직일 때 앞, 뒤, 왼쪽, 오른쪽으로 움직일 때는 지정해준 속도로 움직이지만 대각선으로 움직일 때 엄청나게 빠르게 움직이는 문제가 발생했다.

→원인

각 방향으로 속도를 더해주니까 벡터가 합쳐지면서 설정해준 속도보다 대각선으로 갈 때 속도가 훨씬 빨라진다.

→해결방법

키가 눌릴 때마다 해당 방향에 속도를 더해주는 것이 아닌 키 입력으로 방향이 정해진 다음 마지막에 속도를 더해주면 해당 방향으로 설정한 속도만큼 이동이 가능하다.



키보드 입력을 받을때마다 해당 방향에 속도를 더해준다.

→ 키보드 입력을 받아 방향을 정한 다음 마지막에 속도만큼 더해준다.

- 마우스 커서가 안보이도록 설정

마우스 커서가 안보이도록 설정하고 기존의 클릭하여 화면을 이동하는 방식이 아닌 그냥 마우스만 이동하면 화면이 움직이도록 설정

03.17 ~ 03.18 Controller

	<ul style="list-style-type: none"> - Controller 서버와의 연동을 대비하여 각 Scene과 플레이어 별로 마우스와 키보드를 관리할 Controller를 생성하였다. - Scene Controller 현재 LobbyScene과 GamePlayScene을 관리 GamePlayScene에서는 마우스와 키보드의 입력에 따라서 1인칭, 3인칭, 프리뷰 별로 카메라를 이동시킨다. - GamePlay Controller 플레이어의 이동 및 마우스 움직임에 따른 화면 이동 관리 <p>03.20 GamePlayScene</p> <ul style="list-style-type: none"> - GamePlayScene(30%) SceneController를 적용하여 GamePlayScene 생성 조명을 불러오고 3가지 카메라 시점을 적용할 수 있도록 수정 실행이 잘되는 것을 확인 후 오브젝트를 추가하는 부분 코딩 예정 - Shader 수정 GameScene을 생성하고 실행했을 때 오류 발생 셰이더 파일에 값이 제대로 전달되지 않는 것 같아서 수정 중이다.
다음 주 할 일	<p>[0] 공동 03.22 주간 회의 (13주차) 안건 : 일일계획 공유</p> <p>[1] 김영준 (서버) 주간 목표 - 인 게임 서버 구현 - 멀티 스레드 이해와 공유자원 관리</p> <p>[2] 김동석 (클라이언트) 주간 목표 - GameScene 마무리 - Object 관리자 만들기</p>
문제점	[1] 김영준 (서버)

	<ul style="list-style-type: none"> - 멀티스레드 코딩을 하면서 lock, unlock을 통해 공유자원을 관리하는 설계를 진행한 결과를 볼 때마다 싱글스레드 코딩과 다르지 않은 허술한 결과가 나와서 답답함, 공유자원을 사용하지 않아도 되는 부분에 대해서는 최대한 공유자원을 줄이는 방법으로 설계를 진행할 수 있도록 노력하겠습니다. <p>[2] 김동석 (클라이언트)</p> <ul style="list-style-type: none"> - 허리 상태가 많이 좋아졌다. - 지금까지 해온 것과 앞으로 해야 할 일들을 정리해봤는데 막막한 마음에 평소보다 나태한 한 주를 보냈다. 한 번에 모든 일을 하려 하지 않고 조금씩 나눠서 차근차근 계획을 세워서 해야겠다.
--	--

[추가]