|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 32주차 | **기간** | 01.30~02.05 | **지도교수** | 이형구 |
| 이번주 한일 요약 | 빨간색 열쇠용 차량 제작, 좀비 걷기 구현, 중점연구분야 n번 슬라이스 연구 | | | | |

<상세 수행내용>

- 걷기/뛰기에 필요한 플래그 값인 targetType 동기화

- 좀비 걷기/뛰기 애니메이션 배속 맞춤

- 플레이어 슬라이스 높이조절 오프셋 살짝 보정

- 좀비 BT 직렬적 구조에서 병렬적 구조로 수정

- 좀비 숨고르기 애니메이션 적용

- 좀비 속도 설정 살짝 수정 (발소리, 마지막으로 탐지한 플레이어 위치로 이동할 때도 그냥 뛰도록)

- 좀비 발소리를 탐지할 때 거리에 따라 탐지 실수를 하는 기능 구현

<좀비 AI 관련>

- 좀비 병렬구조로 만들고 detect selector 아래 여러 sequence들의 상태를 실시간으로 추적하고해당 작업들의 우선순위를 주어 작동하게 하려고 생각해보니, 그럼 좀비가 너무 똑똑하게 움직일 것 같아서 일단 다시 이전과 같이 작동하도록 하였습니다. (지금은 직렬적으로 여러 작업중 한가지 task가 선택되면 해당 task를 수행하다가 우선순위가 더 높은 작업이 포착되면 해당 작업을 수행하고 해당 작업이 끝나면 블랙보드(detect의 플래그 값들)를 완전히 초기화하고 다시 detect를 수행하는 방식인데, 만약 병렬적으로 구현 완성하면 실시간으로 계속 하위 sequence task들을 계속 수집하고 있고 우선순위가 높은 작업이 끝나더라도 기억해둔 다른 작업을 바로 또 작업하게 됩니다. (ex 플레이어를 쫓아가다가 플레이어를 놓치고 해당 플레이어를 마지막으로 본 위치로 이동하는게 아닌, 미리 기억하고 있던 더 가까운 쪽의 다른 플레이어의 발소리가 들린 쪽을 먼저 탐색하고 그러고 나서 다시 플레이어를 마지막으로 본 위치로 이동하는 등 모든 상황을 실시간으로 포착하고 기억하고 순회하게 됩니다.) 이러면 너무 좀비치고는 너무 똑똑해 보일 뿐만 아니라 최적화에도 그리 좋지 않은 방식인 것 같아 해당 아이디어는 폐기하고 다시 이전 직렬적으로 작동하는 detect selector로 회귀했습니다.

- 보스 좀비는 필요한 모델과 애니메이션 에셋은 미리 찾아봤으나 보스 좀비를 구현하려면 새로운 BT 구현와 애니메이션 연결 작업, 서버와 클라 동기화 작업들이 필요하게 되어, 해당 작업을 통해 BT 구현을 시연해보이는 것은 시간상 사실상 힘들 것 같다 판단되어, 차라리 현재 BT에서 가지수를 늘릴 수 있는 방법들을 모색하였습니다.

스케치, 도표, 평면도, 그림이(가) 표시된 사진

자동 생성된 설명

* 기존 BT

도표, 텍스트, 평면도, 스크린샷이(가) 표시된 사진

자동 생성된 설명

* 새로 구색한 BT

- 따라서 현재 BT에서 “체력이 낮으면 도망가기 기능”과 “발소리를 탐지할 때 거리에 따라 실수하기”, “다른 좀비가 플레이어를 탐지하면 소리를 내서 근처 다른 좀비들이 같이 반응하여 같이 무리지어 플레이어를 따라 가도록 하는 기능”들을 추가로 구현할 생각입니다.

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** |  | | |
| **해결방안** |  | | |
| **다음주차** | 33주차 | **다음기간** | 02.06 ~ 02.12 |
| **다음주 할 일** | 남은 중점 연구분야   * n번 절단 구현하기 * 부활하기 구현 (절단 부위 다시 결합) * 좀비 BT 추가 기능 구현 (가지 노드들 늘리기) | | |
| **지도 교수**  **Comment** |  | | |