**콘솔 게임 설명서**

|  |  |  |  |
| --- | --- | --- | --- |
| **팀원1** | (1413)박태훈 | **팀원2** |  |
| **게임 이름** | 폭설 속 제설하기 | | |
| **게임 개요** | | | |
| ※ 300자 넘게 작성할 것! 게임 방법에 대한 설명도 포함할 것!  추위가 우리를 감싸고, 곧 눈이 올 시기가 되었다. 디미고에서 폭설이 오면 제설을 한다는 소문에 “폭설 속 제설”을 만들었다. 무수히 쌓이는 눈을 치우며 참된 노동을 해보자. 게임의 배경은 폭설이 내리는 공간이다. 플레이어는 ‘>’ 혹은 ‘<’ 의 모양으로 표시되어 방향키 혹은 WASD(위왼쪽아래오른쪽)를 통해 조작이 가능하다. 눈의 곁으로 가서 스페이스바를 누르면 제설을 할 수 있다. 이동할 때마다 스페이스바를 눌러야 해서 사용자의 노동 강도를 높였다. 제설이라는 게임의 특수한 소재 덕분이다. 왼쪽에 있는 눈과 오른쪽에 있는 눈만 제설할 수 있도록 했다.  (화면설명)   1. 초기 화면   게임을 시작하기 전 대기하는 화면이다. 아이템에 대한 설명을 한다.   1. 난이도 선택 및 게임 설명 화면   게임을 시작하기 직전 눈의 속도를 통한 난이도 설정과 간단한 게임 방법에 대해 설명하는 화면이다.   1. 메인 게임 화면   게임이 진행되는 화면이다. 눈을 치울 때 마다 점수가 쌓이고, 점수가 올라가면 점차 스테이지가 증가한다. P인 지나다니는 사람이 눈에 부딪히면 체력이 감소한다. 다음 스테이지로 넘어가면 눈의 속도가 빨라지며 게임의 난이도가 증가한다. 체력이 줄어들어 0이 되면 게임이 종료되고, 점수 안내와 함께 다시 시작된다.  (각 오브젝트 설명)  ‘\*’ : 눈은 위에서 내리다가 아래에 들어와서 쌓인다. 사람이 지나가다가 가로막히면 체력이 줄지만 지나갈 수 있다. 플레이어가 앞에서 스페이스바를 눌러 눈을 치울 수 있다.  아이템 – ‘Q’, ‘H’, ‘X’ : 앞에서 스페이스바를 눌러 사용이 가능하다.  “Q” : 아이템이 위치한 줄의 눈을 모두 제설해준다.  “H” : 플레이어의 체력 5를 회복한다.  “X” : 아이템 근처 블럭( 7\*5 )를 제설해준다.  사람 – ‘P’ : 불규칙한 속도로 맵을 일직선으로 지나다닌다. 만약 진행방향에 눈이 있으면 체력이 줄고 눈을 치우고 지나간다. | | | |
| **새롭게 알게 된 내용** | | | |
| **필요했던 계기 및 발생한 문제** | | | |
| **이미 쌓여 있는 오브젝트 위에 오브젝트를 겹쳐지는 경우가 발생했다.** | | | |
| **해결 방법** | | | |
| **그래서 MAP을 블록별 구조체를 통해 체계화했다. InitBlock 함수에서 사용될 블록을 모두 설정해두었다. UI 역시 오브젝트로 처리하여 구조체 Rect를 통해 MAP에 오브젝트가 있는지 비교해가며 해결하였다. 체계화를 시킨 후에 추가적인 작업을 하니 작업의 속도가 빨라졌고, 오류와 버그가 매우 적어졌다.** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **사용한 조건 설명하기(조건4는 2인 프로젝트인 경우)** | | | |
| **조건1** | 스테이지 3단계 이상 구현하기/화면 구성하기  초기화면/난이도 선택화면/메인게임화면으로 구성하여 게임을 진행했다. (DisplayStart/CloseScene 함수에서 사용)  게임 중간에 스코어에 따라 스테이지를 진행시켜 3회 이상 스테이지 진행이 가능하도록 했다. (addScore함수에서 난이도를 변경하였다) | | |
| **조건2** | 글자색 또는 배경색을 바꾸기 / 키보드 입력 처리하기  키보드 입력을 활용해 방향키와 스페이스바를 통한 조작을 가능하게 했다. (Move함수) 글자색을 SetColor, DrawObj, DrawUI함수 등에서 교체하여 사용했다. | | |
| **조건3** | 구조체로 새로운 타입 정의하기  Gblock, Rect, Obj를 선언하여 Map에 블록을 지정할 때 Gblock 구조체를 사용했고 Obj는 오브젝트의 좌표와 그 형태를 담은 Gblock을 변수로 사용하여 UI와 눈/아이템 등의 오브젝트에서 사용했다. Rect는 UI를 담는 사각형을 만들어 사용하였다. | | |
| **조건4** | 수업시간에 배우지 않은 C언어의 새로운 부분 사용하기  Time.h에서 clock함수를 사용하여 게임을 시작하고나서 걸린 시간을 출력하였다. | | |
| **실행 화면** | | | |
| **화면1** | | **화면2** | **화면3** |
| 게임을 시작하기 전 대기하는 화면이다  초기화면 | | 게임을 시작하기 직전 눈의 속도를 통한 난이도 설정과 간단한 게임 방법에 대해 설명하는 화면이다.    난이도 선택 및 게임 설명 화면 |  |
| **프로젝트 진행하면서 느낀점** | | C언어로 게임을 만드는 경험은 이전의 문법을 교육 받은 것보다 훨씬 흥미롭고 새로웠다. 나의 시간을 투자해서 게임을 만들고, 혼자서 코드를 최대한 체계적으로 짜려고 노력하다 보니 프로그램의 흐름을 스스로 구성할 능력을 갖추어 가는 과정처럼 느껴져서 좋았다. 이번엔 혼자서 프로젝트를 진행하여 플레이어 이벤트 처리, 각 블럭별 핸들러 적용, 충돌 처리 등의 과정이 복잡하게 느껴졌다. 마침내 콘솔게임을 목표치 이상 완성한 점이 기쁘다. 하지만 이번에 프로그래밍언어가 C언어로 한정되어 아쉽기도 했다. 다음에도 기회가 있다면 프로젝트를 만들어 나가고 싶다. | |