Programming Assignment – Rush Hour

Requirement Checklist

이름 & 학번

**~~요구사항1~~**~~. start를 하기 전에 left, right, up, down 등의 명령을 내릴 경우, invalid command로 처리한다.~~

**~~처리방법1~~**~~. start가 되지 않았는지 유무는 cars 값이 0x0 인지를 통해 확인할 수 있다. main 함수의 left, right, up, down을 처리하는 case 부분에서 cars 값을 확인하여, cars가 0x0인 경우에는 invalid command로 처리하였다.~~

**~~요구사항2~~**~~. start에서 입력 받은 파일명에 해당하는 파일이 없는 경우, 에러 메세지를 표시한다.~~

**~~처리방법2~~**~~. fopen()의 return value가 0x0인 경우, load\_file()가 return 1을 하고, main에서 에러 메시지를 출력하게 하였다.~~

…

**~~요구사항3~~**~~. Start에서 입력 받은 파일 내 첫 줄에 첫 줄에 차의 개수가 적히지 않은 경우, 차의 개수가 2< 거나 >36인 경우 에러 메시지를 출력~~

**처리방법3**. ~~첫 줄에 적힌 문자를 atoi를 사용하여 정수인지 확인, atoi return 값이 0인 경우 “invalid data” 출력 정수라면 2<, >36인지 확인함 이 사이에 통과하지 못하면 return 1~~

**~~요구사항4~~**~~. Start에서 입력 받은 파일 내 첫 줄에 적힌 차의 개수가 2> 거나 >36인 경우~~

**~~처리방법4~~**~~. 2<, >36인지 확인함 이 사이에 통과하지 못하면 “invalid data” 출력 ,return 1~~

**~~요구사항5~~**~~. Start에서 입력 받은 파일 내 첫 줄에 적힌 차의 개수와 실제 라인 수 일치~~

**~~처리방법5~~**~~. 첫 줄에 적힌 정수와 라인 수 비교하고 맞지 않을 시 “invalid data” 출력 ,return 1~~

**요구사항6**. Start에서 입력 받은 파일 내 차가 board범위를 넘은 경우,

**처리방법6**. ~~‘:’을 기준으로 strtok했을 때 ‘A1 ~ F6’인지 확인하는 함수를 사용하여 return 값이 1인 경우 에러로 처리, 메세지 출력~~ , ~~현재 차의 위치를 update후 x1, x2, y1, y2의 값의 범위를 확인하여 board를 벗어난 경우 에러 처리~~

**~~요구사항7~~**~~. 입력받은 차의 span이 1과 6사이가 아닌 경우 에러 메세지 표시~~

**~~처리방법7~~**~~. Load\_game에서 한 줄을 getline으로 받은 후 strtok를 사용하여 하나씩 판단할 때 그 정수가 1과 6사이인지 판단하고 아닐 경우 return~~ 1

**요구사항8**. ~~처음 입력된 좌표를 토대로 span만큼, 그리고 방향에 따라 차를 배치시켜야 함~~

**~~처리방법8~~**~~. 처음 배정받은 위치 ex ‘C6’, ‘vertical’ 인 경우 c6 = cell[5][2] 인 부분을 오른쪽 아래 방향에 맞게 car의 변수들을 할당한다.~~

**~~요구사항9~~**~~. Start에서 입력 받은 파일 내 한 cell에 두 개 이상의 차가 배치된 경우~~

**~~처리방법9~~**~~. Cell 값이 0이 아닌 경우에는 이미 배정된 차가 있다는 의미, Load\_game에서 retrun 1~~

* ~~처음에는 차를 받을 때마다 바로 cell에 넣는 걸로 하자!~~

**~~요구사항10~~**~~. 1번째 차는 horizontal하게 4번째 줄에 있어야 함~~

**~~처리방법10~~** ~~1번째 차의 direction은 horizontal, 4번째 줄에 있는지 확인 아닌 경우 load 에서 return, “invalid data”출력~~

**~~요구사항11~~**~~. 입력받은 것 ‘:’을 기준으로~~

**~~처리방법11.~~**

**~~요구사항 12.~~** ~~start에서 입력 받은 파일을 읽을 때 각각 car의 direction이 유효한 값이어야 한다~~**~~.~~**

**~~처리방법 12.~~** ~~각각 car의 direction이 vertical 또는 horizontal을 제외한 기타 값일 경우 에러 메시지를 출력하고 종료한다.~~

//load game 완료

Move 시작

**~~요구사항13~~**~~. Command로 주어진 차의 번호 (parameter id)가 n\_car보다 크거나 0보다 작은 경우~~

**~~처리방법13~~**~~. Move 함수에서 판단하여 return 1~~

**~~요구사항14.~~** ~~이동명령은 한번에 한칸 이동하도록 동작한다.~~

**~~처리방법14.~~** ~~Horizontal, vertical에 맞는 방향으로 한칸씩 이동하도록 처리한다.~~

**~~요구사항15~~**~~. Right, Left, up, down로 차를 움직이려는 경우 그 차의 direction과 맞지 않은 명령어일 경우~~

**~~처리방법15.~~** ~~움직이려는 차의 평행, 수직과 움직이려는 명령어가 같은지 확인 (cars.[id].dir 이 vertical이라면 up, down만, horizontal이라면 left, right만)하고 맞지 않은 경우에는 “impossible” 출력 후 return 1~~

**~~요구사항16~~**~~. 움직일 때board를 벗어난 경우~~

**~~처리방법16.~~** ~~확인해야하는 부분 check, up: y2, down:y1, right x2, left: x1 이 0<, >7인지 확인~~

**~~요구사항17~~**~~. 차를 움직였을 때 다른 차와 겹치는 경우~~

**~~처리방법17.~~** ~~확인해야하는 부분 check, up: y2, down:y1, right x2, left: x1, 이 부분 값들을 적절하게+-1 해준 후 cell[]에 값이 0이 아닌 경우에는 겹친 것으로 판단, move 함수에서 “impossible”출력 후 return 1~~

//main

**~~요구사항18~~**~~. 주어진 command 외에 다른 입력어가 들어온 경우~~

**~~처리방법18.~~** ~~switch문에서 검사하여 “invalid command” 출력~~

//cell

**~~요구사항19~~**~~. 게임이 끝난 경우를 알아야 함~~

**~~처리방법19.~~** ~~update\_cell할 때 cars[0].x2가 5면 게임 끝~~

잘못된 걸 입력시 getchar, flush로 비우기 ??