Todo

**Unit**

-Inheritance and polymorphism should be actively utilized.

Let’s think of inheritance tree.

Enemies and enemy\_bullets disappear when they reach the end of the wall.

All units pass through each other can exist in the same cell

* Can print anything if there’s more than one object on one cell.
* My plane (‘M’) must be printed. -> print share() 이용

Unit Data Sheet

Bullets

|  |  |  |
| --- | --- | --- |
| Name | Func | Speed |
| Bullet(‘,^,!) |  | 1 |
| Powerup(P) | My\_plane generates 3 bullets at once. |  |
| Levelup(L) | Increases the level of the bullet. |  |
| Enemy\_bullet(\*) | Damage 1 (can be buffed.) | 1 |

Bullet

* The symbol changes depending on the level.
* The damage from the bullet is equal to the level.

Planes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Hp | Speed | Score | Shot Frame | Feature |
| My\_plane(M) | 5 |  |  | 1 | vector<Bullet> |
| Enemy\_1n(n) | 10 | 0 | 1 |  |  |
| Enemy\_2r(r) | 5 | 3(cell/0.3s) | 2 |  |  |
| Enemy\_3s(s) | 4 | 9 | 3 |  | Enemy\_bullet() after moves |
| Enemy\_4d(d) | 5 | 3 | 4 |  | Enemy\_bullet() diagonally |
| Enemy\_5a(a) | 8 | 0 | 5 |  | Givin buff() (Every 0.6s) |

Enemy implementation

public

* Moving
* Int HP (that can be changed, public?)
* Int score (only getter needed.)
* Int speed
* Shot frame if it is needed.

How to implement movement of enemy plane?

If (speed!=0)

{movement implementation}

How about implementing all features of enemy in header file…. NO usage of case:

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

From this, we should place enemies on the right position.

**Interaction**

When objects are in the same cell.

1. Enemy\_1~5 <->my\_plane  
   my\_plane gets damaged. (my\_plane.hp--)
2. Enemy\_bullet <-> my\_plane  
   my\_palne gets damaged. (my\_plane.hp-=enemy\_bullet.damage)
3. Bullet <-> Enemy1~5  
   Enemy gets damaged. (enemy.hp -= bullet.level

**Overall System**

Input file로부터 사건 발생시키기

일시정지

종료

종료 후 출력

* Create events from the input file.
* If my\_plane’s hp is less than or equal to 0, the game ends.
* If all enemies are dead, and all the events received from the input file are generated, the game ends.
* Press the ‘p’ key to pause the game. You can press any key to resume.
* When the game is over, you must print the total score and the number of kills by enemy unit type.