C++ Inheritance

Lab 10

Objectives

- Get used to C++ project structure. (*.cpp, *.h)
- Practice OOP with C++:
  - Inheritance
    - Class extending
    - Method overriding
  - Encapsulation
    - Access modifiers
  - Polymorphism
    - Operator overriding

Po***c***ke***t***mon Game Application

- Some of you (Older students...) may have some memories of Pokemon games.
- We will make 2-player pocketmon game.

Enemy MILTANK
used ROLLOUT!

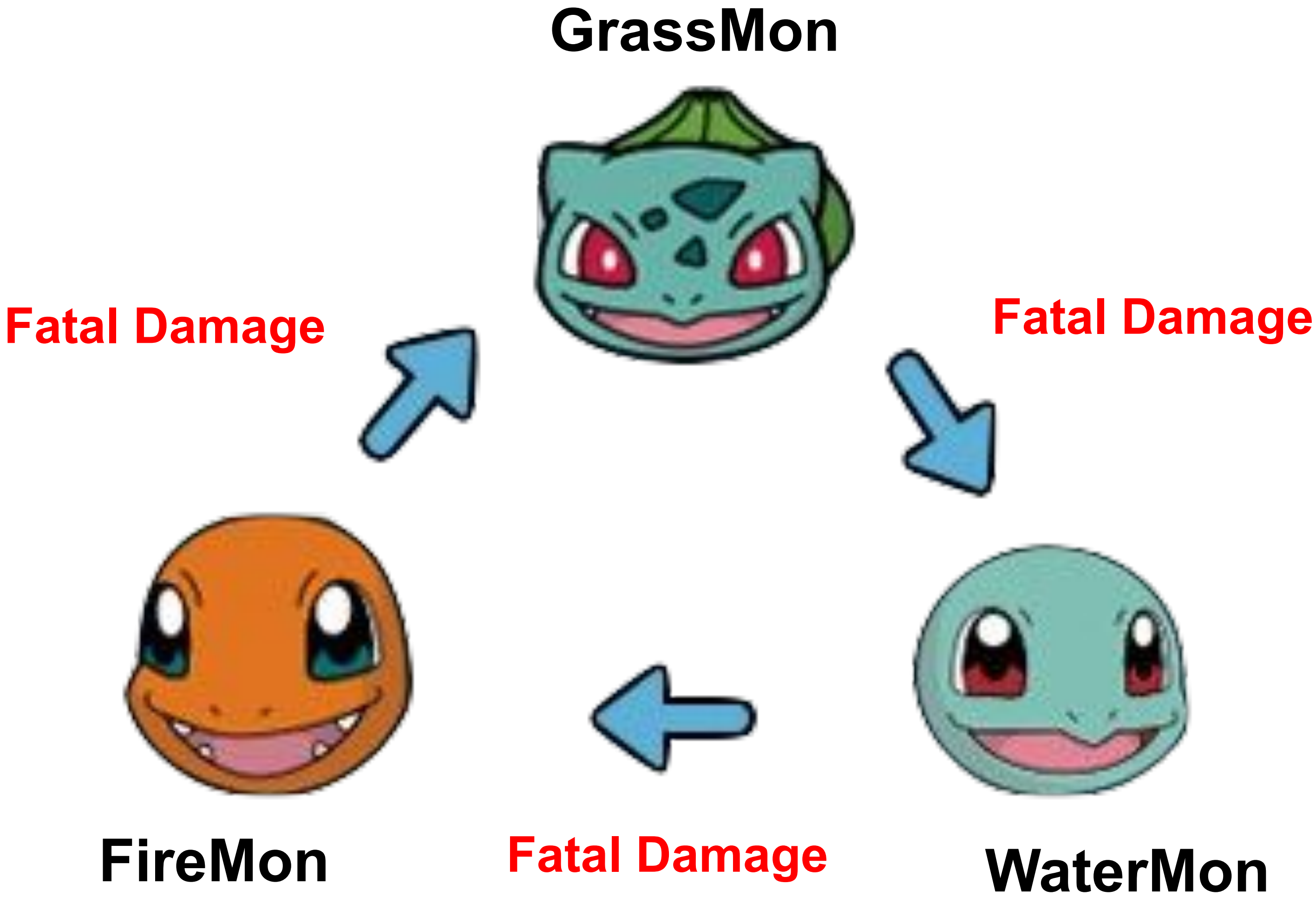
Po***c***ke***t***mon Game Application

- There are two players in the game.
- Each player can have at most 6 monsters.
- There are three types of monster, WaterMon, FireMon, and GrassMon.
- At each round,
  a. each player chooses one of the monster.
  b. The 1st player’s chosen monster attacks the 2nd player’s monster, and then the 2nd player’s monster attacks back.

Normal & Counter Damage

- The default attack damage is given for each type of monster.
- At each attack, the health of the target is decreased by the amount of the damage.
- Each type of monster has a counter monster type. WaterMon >> FireMon >> GrassMon >> WaterMon …
  - For example a WaterMon does a huge damage (“fatal damage”) to a FireMon.
- Each type of monster has its own “fatal damage”.

Normal & Counter Damage



GrassMon

Fatal Damage

Fatal Damage

FireMon

Fatal Damage

WaterMon

Specification

- Implement methods in pocketmon_game.h.
  - `add_monster_for_player1/2(string monster_type)`
    - Add a monster to the player.
    - monster_type: “WaterMon”, “FireMon”, “GrassMon”
  - `attack1(i, j)`
    - i_{th} monster of the player 2 attacks j_{th} monster of the player 1.
  - `attack2(i, j)`
    - i_{th} monster of the player 1 attacks j_{th} monster of the player 2.
- Add any files, classes, or functions if you need.
- Feel free to change any trivial specifications.

Result of test.cpp

```
===== Current Status =====
player 1: [8WaterMon 1, health=100][7FireMon 2, health=100]
player 2: [8GrassMon 3, health=100][7FireMon 4, health=100]
===== Current Status =====
player 1: [8WaterMon 1, health=100][7FireMon 2, health=100]
player 2: [8GrassMon 3, health=100][7FireMon 4, health=75]
===== Current Status =====
player 1: [8WaterMon 1, health=90][7FireMon 2, health=100]
player 2: [8GrassMon 3, health=100][7FireMon 4, health=75]
```

Advanced (Optional) 1/2

- Print a message if a monster is beaten(health is below 0). Also the beaten monster can’t attack back.

Advanced (Optional) 2/2

- Make an interactive UI wrapping the game app.

```
===== Choose monsters for player 1 =====
the number of the monsters: 2
monster 1 type: WaterMon
monster 2 type: FireMon
...
===== Choose monsters for player 2 =====
…
===== Current Status =====
player 1: [WaterMon 1, health=100][FireMon 1, health=100]
player 2: [GrassMon 1, health=100][FireMon 1, health=100]

===== Match 1 =====
player 1: choose monster: 1
player 2: choose monster: 2
        player 1 (WaterMon 1) => player 2 (FireMon 1) : [Attack] damage: 10
        player 2 (WaterMon 1) => player 1 (WaterMon 1) : [Fatal Attack] damage: 25
...
===== Current Status =====
...
===== Game Over ===== # Choose any policy to end the game
```