The Class Enemy

1 Specification:

1) Characters have the following properties:

- Attack (A) – indicates the attack-force of the character
- Attack Chance – indicates the probability of a successful attack.
- Defence (D) – indicates the defensive abilities of the character
- Defence Chance – indicates the probability to defend against a successful attack.
- Health (H) – indicates the health status of a character. Once the health level has reached 0 the character is defeated.
- Strength (S) – indicates the weight a character can carry.

2) Whenever a player attacks an enemy, the enemy will immediately afterwards attack the player (unless the enemy has been defeated on the last attack).

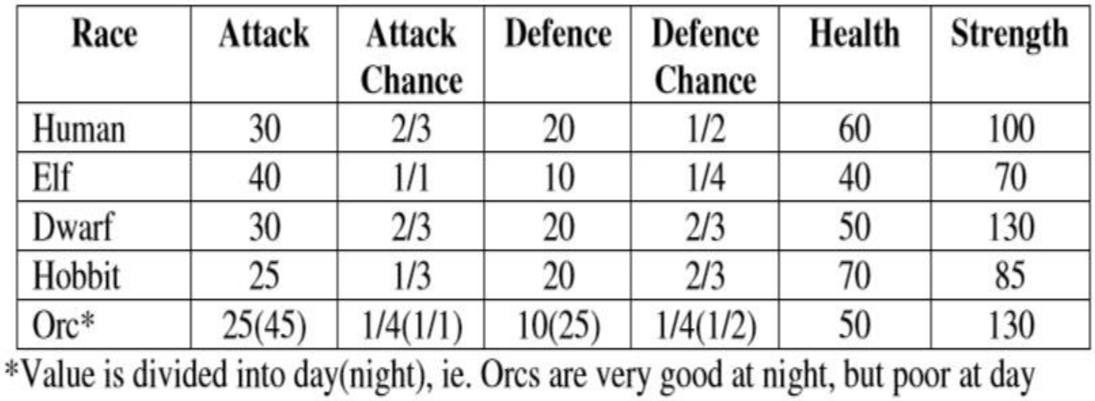
3) The attacker resolve the attack success determined by the attack chance of the attacker's race. On successful attack, the defender resolves the defence success determined by the defence chance of the defender's race.

4) Value is divided into day(night), ie. Orcs are very good at night, but poor at day.

5) Enemies have the same characters/types as players.

6) The player can see the information of enemy when they meet or fighgting with them.

7) The race defines the abilities of a character(Human Elf Dwarf Hobbit (Halfling) Orc). The player will meet any tpye enemy in the following chart. The following races are available:



Race	Attack	Attack Chance	Defence	Defence Chance	Health	Strength
Human	30	2/3	20	1/2	60	100
Elf	40	1/1	10	1/4	40	70
Dwarf	30	2/3	20	2/3	50	130
Hobbit	25	1/3	20	2/3	70	85
Orc*	25(45)	1/4(1/1)	10(25)	1/4(1/2)	50	130

*Value is divided into day(night), ie. Orcs are very good at night, but poor at day

8) The game starts on daytime. Daytime and Night-time alternate every 5 moves.

1.1 Specification of enemy Generation

1. **enemy ()**: constructor.

- *Pre: none.*
- *Post: constructor was created.*

2. **enemy(character choose)**: set the enemy class with choose.

- *Pre: none*
- *Post: set the enemy use the choose.*

3. ~**enemy():** destructor.

- *Pre: existed constructor.*
- *Post: delete the enemy class*

4. **int Attack():** attack the player.

- *Pre: meet the enemy and start defeat.*
- *Post: attack the player according to the attack chance.*

5. **void Defence(int &Harm Vaule)**: defense for the attack according to the defence chance.

- *Pre: successful attack by enemy.*
- *Post: defence the attack according to the defence chance.*

6. **void OrcAlter(int &steps)**: change the attributes of Orc every 5 times.

- *Pre: meet Orc as enemy.*
- *Post: change the value of Orc's attributes every 5 times(daytime and nightime).*

7. **void print () const**: print the current information of enemy.

- *Pre: meet the enemy.*
- *Post: print the current attribute information in the screen.*

8. **character getEnemy()**: get the type of the enemy.

- *Pre: there has been an enemy.*
- *Post: get the enemy attributes*

9. void setEnemy(character & NewEnemy):change the enemy type.

- *Pre: there has been an enemy.*
- *Post: set the type of the enemy.*

2 User's View of Services

- enemy();
- enemy(character choose);
- virtual ~enemy();
- int Attack() ;
- int Defence(int &HarmVaule);
- void OrcAlter(int &steps);
- void print() const;
- inline character getEnemy();
- inline void setEnemy(character & NewEnemy);

3 Internal Data Representation

Variable	Type	Access
EnemyTypeg	character	private

4 Remaining Definitions

5 Coding