

### Compte rendu exercice 3

```
#include <iostream>
#include <cstring>

class Weapon{
private:
    char name[20];
    int damage;
```

1)

```
class RPCharacter{
private:
    char name[15];
    int level;
    int xp_points;
    int hp;
    Weapon weapon_list[10];
    int weapon_quantity;
    Weapon weapon_used;
```

2)

```
Weapon() : name("None"), damage(0) {}
```

3)

On

initialise une arme par défaut qui est vide.

```
Weapon(const char newName[], int newDamage) : damage(newDamage) {
    strncpy(name, newName, sizeof(name) - 1);
    name[sizeof(name) - 1] = '\0';
}
```

4)

Le constructeur avec deux arguments nous permet d'ajouter une nouvelle

arme en choisissant son nom et des dégâts. On utilise strncpy car le nouveau nom est une chaîne de caractère.

```
const char* getName() const { return name; }  
int getDamage() const { return damage; }
```

5)

Voici les deux getters.

```
public:  
  
    RPCharacter(const char newName[]) {  
        strncpy(name, newName, sizeof(name) - 1);  
        name[sizeof(name) - 1] = '\\0';  
  
        level = 1;  
        xp_points = 0;  
        hp = 100;  
        weapon_quantity = 0;  
        weapon_list.push_back(Weapon());  
        weapon_used = Weapon();  
    }
```

6)

Le constructeur prend un nom en paramètre et met tous les autres attributs par défaut ( pas d'armes, niveau 1 etc.)

```

class RPCharacter{
    char name[15];
    int level;
    int xp_points;
    int hp;
    std::vector<Weapon> weapon_list;
    int weapon_quantity;
    Weapon weapon_used;
    bool is_dead;

public:
    RPCharacter(const char newName[]) {
        strncpy(name, newName, sizeof(name) - 1);
        name[sizeof(name) - 1] = '\0';

        level = 1;
        xp_points = 0;
        hp = 100;
        weapon_quantity = 0;
        weapon_list.push_back(Weapon());
        weapon_used = Weapon();
        is_dead = false;
    }
}

```

7)

On ajoute le paramètre is\_dead qui sera à False par défaut.

```

void apply_damage(int damage_value){
    hp -= damage_value;
    if (hp <= 0){
        is_dead = true;
        hp = 0;
    };
};

```

8)

```

Weapon get_weapon(){
    if (weapon_quantity > 0){
        return weapon_used;
    }else if (weapon_quantity == 0){
        weapon_quantity += 1;
        weapon_list.clear();
        if (level = 1){
            weapon_list.push_back(Weapon("Hand", 1));
        }else{
            weapon_list.push_back(Weapon("Hand", level/2));
        };
        return Weapon();
    };
};

```

9)

La méthode get\_weapon renvoie l'arme utilisée et ajoute les poings si le joueur ne possède pas d'arme. Les dégâts des poings varient en fonction du niveau du joueur.

```

void Attack(RPCharacter& targetCharacter) {
    int damage = get_weapon().getDamage();
    targetCharacter.apply_damage(damage);
    xp_points += damage;
};

```

10)

La fonction Attack applique des dommages à un autre joueur. Le joueur récupère un nombre d'xp égal aux dégâts infligés.

```

void store_weapon(Weapon weapon){
    if (weapon_quantity == 10){
        weapon_list.pop_back();
    };
    weapon_list.push_back(weapon);
};

```

11)

```

void switch_weapon(int indice){
    if ( indice < 0 || indice > 9){
        std::cout << "Veuillez rentrer un emplacement valide" << std::endl;
        return;
    };

    if (weapon_list[indice].getName() == "None") {
        int last_weapon_index = -1;
        for (int i = 0; i < 10; ++i) {
            if (weapon_list[i].getName() != "None") {
                last_weapon_index = i;
            }
        }

        if (last_weapon_index != -1) {
            weapon_used = weapon_list[last_weapon_index];
            std::cout << "Arme switchée à l'emplacement " << last_weapon_index << std::endl;
        } else {
            std::cout << "La liste des armes est vide" << std::endl;
        }
    } else {
        weapon_used = weapon_list[indice];
        std::cout << "Arme switchée à l'emplacement " << indice << std::endl;
    }
};

```

12)

13)

```

Weapon get_weapon(int indice){
    if (weapon_quantity == 0){
        weapon_quantity += 1;
        weapon_list.clear();
        if (level == 1){
            weapon_list.push_back(Weapon("Hand", 1));
        }else{
            weapon_list.push_back(Weapon("Hand", level/2));
        };
        return weapon_list[1];
    };

    if ( indice > -1 || indice < 10){
        if (weapon_list[indice].getName() == "None") {
            int last_weapon_index = -1;
            for (int i = 0; i < 10; ++i) {
                if (weapon_list[i].getName() != "None") {
                    last_weapon_index = i;
                }
            }
            if (last_weapon_index != -1) {return weapon_list[last_weapon_index];}

        } else {
            return weapon_list[indice];
        }
    }
}
};

```

14)

```

void drink_potion(int value){
    hp += value;
    if ( hp > 100 ){
        hp = 100;
    }
};

```

15)

- 16) En modifiant un peu les scripts et en ajoutant l'affichage des actions des joueurs, on peut créer un scénario dans le Main :

```
int main(void){

    // Création des deux joueurs
    RPCharacter Noamus = RPCharacter("Noamus");
    RPCharacter Antoinus = RPCharacter("Antoinus");

    // Les deux joueurs sortent leurs poings
    Noamus.get_weapon(0);
    Antoinus.get_weapon(0);

    // Noamus est en possession d'une sarbacane malicieuse ainsi que d'une baguette magique
    Noamus.store_weapon(Weapon("Sarbacane", 10));
    Noamus.store_weapon(Weapon("Baguette magique", 20));

    // Les deux joueurs se donnent quelques coups
    Noamus.Attack(Antoinus);
    Antoinus.Attack(Noamus);
    Noamus.Attack(Antoinus);
    Antoinus.Attack(Noamus);
```

```
// Noamus brandit sa sarbacane et use de sa malice en tirant deux épines aux pieds d'Antoinus
Noamus.switch_weapon(1);
Noamus.Attack(Antoinus);
Noamus.Attack(Antoinus);

// Antoinus est pris au dépourvu mais trouve un fléaux géant à ses pieds qu'il ramasse
Antoinus.store_weapon(Weapon("Fléaux Géant", 30));

// Se trouvant à distance, Antoinus ne peut pas porter de coup à Noamus.
// Ce dernier en profite pour sortir sa baguette magique et lancer une attaque remplie de fourberie
Noamus.switch_weapon(2);
Noamus.Attack(Antoinus);

// Après avoir encaissé cette attaque farfelue, Antoinus court en direction de Noamus et le frappe avec
Antoinus.switch_weapon(1);
Antoinus.Attack(Noamus);
```

```
// Noamus recule et brandit une potion de shield bleue tah fortnite prime qu'il engloutit
Noamus.drink_potion(50);

// Cela poussa Antoinus à user de la malveillance max
// La malice ne suffit pas à Noamus pour stopper la série d'attaque d'Antoinus
std::cout << "MALVEILLANCE MAX !!!!!!!!!!!!!!" << std::endl;
Antoinus.Attack(Noamus);
Antoinus.Attack(Noamus);
Antoinus.Attack(Noamus);
Antoinus.Attack(Noamus);

// Antoinus crie de rage après avoir éteint le petit lutin espiègle Noamus
```

```
PS C:\Users\leolo\OneDrive\Bureau\P00_Project\output> cd 'c:\Users\leolo\OneDrive\Bureau\P00_Project\output'
PS C:\Users\leolo\OneDrive\Bureau\P00_Project\output> & .\'exercice3.exe'
Le Joueur Noamus apparait
Le Joueur Antoinus apparait
Le Joueur Noamus se muni de ses poings
Le Joueur Antoinus se muni de ses poings
Le Joueur Noamus ajoute l'arme Sarbacane á son inventaire !
Le Joueur Noamus ajoute l'arme Baguette magique á son inventaire !
Le Joueur Noamus attaque le joueur Antoinus
Le Joueur Antoinus perd 1 hp !
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 1 hp !
Le Joueur Noamus attaque le joueur Antoinus
Le Joueur Antoinus perd 1 hp !
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 1 hp !
Arme switchée á l'emplacement 1
Le Joueur Noamus attaque le joueur Antoinus
Le Joueur Antoinus perd 10 hp !
Le Joueur Noamus attaque le joueur Antoinus
Le Joueur Antoinus perd 10 hp !
Le Joueur Antoinus ajoute l'arme Flaux Gant á son inventaire !
Arme switchée á l'emplacement 2
Le Joueur Noamus attaque le joueur Antoinus
Le Joueur Antoinus perd 20 hp !
```

```
Arme switchée á l'emplacement 1
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 30 hp !
Le Joueur Noamus se régénère de 50 hp !
Le Joueur Noamus est full vie
MALVEILLANCE MAX !!!!!!!!!!!!!
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 30 hp !
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 30 hp !
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 30 hp !
Le Joueur Antoinus attaque le joueur Noamus
Le Joueur Noamus perd 30 hp !
Le Joueur Noamus est mort !
```