
	M5 Entornos de desarrollo		
	UF2-Optimización de Software		
PR4-Refactorización de código			
Apellidos: Farga i Expósito	Nombre: Sergi	Fecha:	15/01/2023

Para realizar esta práctica, he modificado las siguientes funciones:

-bool ataqueEnemigo1(), bool ataqueEnemigo2(), bool ataqueEnemigo3() y bool ataqueEnemigo4() pasan a ser bool ataqueEnemigo(int enemyDamage, int& heroHP, string enemyName, bool& enemyIsAlive).

Antes:

```
bool ataqueEnemigo1() {
    if (enemy1IsAlive)
    {
        enemy1Damage = rand() % 20 + 1;
        heroHP = heroHP - enemy1Damage;
        cout << "El " << enemy1Name << " te ha atacado!" << endl;
        cout << "Te quedan " << heroHP << " puntos de vida" << endl << endl;

        return true;
    }
    else
        return false;
}
```

Sergi Farga

Después:

```
bool ataqueEnemigo(int enemyDamage, int& heroHP, string enemyName, bool& enemyIsAlive) {
    if (enemyIsAlive)
    {
        enemyDamage = rand() % 20 + 1;
        heroHP = heroHP - enemyDamage;
        cout << "El " << enemyName << " te ha atacado!" << endl;
        cout << "Te quedan " << heroHP << " puntos de vida" << endl << endl;

        return true;
    }
    else
        return false;
}
```

Sergi Farga

El código resultante tras convertir las 4 funciones de ataqueEnemigo en una sola al llamar dicha función queda como se muestra a continuación:

Antes:

```
//Enemigo 1 ataca
heroIsAlive = ataqueEnemigo1();
//Enemigo 2 ataca
heroIsAlive = ataqueEnemigo2();
```

Sergi Farga

Después:

```
//Enemigo 1 ataca
heroIsAlive = ataqueEnemigo(enemy1Damage, heroHP, enemy1Name, enemy1IsAlive);
//Enemigo 2 ataca
heroIsAlive = ataqueEnemigo(enemy2Damage, heroHP, enemy2Name, enemy2IsAlive);
```

Sergi Farga

-int heroChoosesEnemyToAttackPath1() y int heroChoosesEnemyToAttackPath2() pasan a ser int heroChoosesEnemyToAttack(string heroName, string& enemy1Name, string& enemy2Name).

Antes:

```
int heroChoosesEnemyToAttackPath1() {
    int selectorEnemigo;
    cout << heroName << ", a quien quieres atacar?" << endl;
    cout << "[1] " << enemy1Name << endl << "[2] " << enemy2Name << endl;
    cin >> selectorEnemigo;
    if (selectorEnemigo == 1)
    {
        return 1;
    }
    else if (selectorEnemigo == 2)
    {
        return 2;
    }
}
```

Sergi Farga

Después:

```
int heroChoosesEnemyToAttack(string heroName, string& enemy1Name, string& enemy2Name) {
    int selectorEnemigo;
    cout << heroName << ", a quien quieres atacar?" << endl;
    cout << "[1] " << enemy1Name << endl << "[2] " << enemy2Name << endl;
    cin >> selectorEnemigo;
    if (selectorEnemigo == 1)
    {
        return 1;
    }
    else if (selectorEnemigo == 2)
    {
        return 2;
    }
}
```

Sergi Farga

El código resultante tras convertir las 2 funciones de heroChoosesEnemyToAttack en una sola al llamar dicha función queda como se muestra a continuación:

Antes:

```
atacarUnEnemigo = heroChoosesEnemyToAttackPath1();
```

Sergi Farga

Después:

```
atacarUnEnemigo = heroChoosesEnemyToAttack(heroName, enemy1Name, enemy2Name);
```

Sergi Farga

Solo con realizar las mencionadas modificaciones, el código total del programa pasa de ocupar 533 líneas a ocupar 464.