



# INFORME FUNCIONES JUEGO

ALEJANDRO GODOY CINTAS  
1DAW

Para empezar la primera función del videojuego es la del **gamestart()**

```
void gamestart() {
    cout << "////////////////////////////////////////\n";
    cout << "-----Bienvenidos a Street Fighter-----\n";
    cout << "////////////////////////////////////////\n\n\n";
    cout << "Presiona cualquier tecla para continuar...";
    cin.get();
    system("cls");

    cout << "Introduce el nombre del Jugador: ";
    cin >> playername;
    cout << "El nombre del jugador es " << playername << "\n";
}
```

En este caso tal y como se pide en la práctica, la asignación del nombre del héroe debe ser de tipo **void**, en la que simplemente se asignara el nombre de playername.

```
bool statusEnemy1() {
    if (HPenemy1 <= 0) {
        HPenemy1 = 0;
        enemy1IsAlive = false;
        return enemy1IsAlive;
    }
    else {
        return enemy1IsAlive;
    }
}
```

A continuación, se ha creado una función para ver el estado de los enemigos en la que si la vida es 0 nos retorna un enemyisalive = false, de lo contrario =true para todos los enemigos y para el heroe

Seguidamente se ha creado una función de tipo **void** en la que debemos seleccionar el tipo de ataque que queremos hacer al enemigo y dependiendo de la respuesta que introduzca el jugador nos dará el nombre del ataque y el daño que le hemos quitado al enemigo

```
void seleccionAtaqueEnemy1() {
    Haduken = rand() % 25;
    Shoryuken = rand() % 40;
    do {
        cout << "Que ataque quieres hacer?\n[1] Haduken\n[2] Shoryuken\n[3] Ultimate\n";
        cin >> selectAtaque;
    } while (selectAtaque != 1 && selectAtaque != 2 && selectAtaque != 3);
    if (selectAtaque == 1) {
        HPenemy1 -= Haduken;
        statusEnemy1();
        cout << "Le has hecho un Haduken a " << enemyName1 << " y le quedan " << HPenemy1
        << " puntos de vida.\n";
    }
    else if (selectAtaque == 2) {
        HPenemy1 -= Shoryuken;
        statusEnemy1();
        cout << "Le has hecho un Shoryuken a " << enemyName1 << " y le quedan " <<
        HPenemy1 << " puntos de vida.\n";
    }
    else if (selectAtaque == 3 && ultimateStatus) {
        HPenemy1 -= Ultimate;
        statusEnemy1();
        cout << "Le has tirado la Ultimate a " << enemyName1 << " y le quedan " <<
        HPenemy1 << " puntos de vida\n Ya no puedes usar mas la Ultimate durante la partida.\n";
        ultimateStatus = false;
    }
    else if (selectAtaque == 3 && !ultimateStatus) {
        cout << "Ya no te queda la Ultimate, has gastado un ataque.\n";
    }
}
```

También se ha realizado una función de tipo void en la que el enemigo ataca al héroe y le quita x puntos de vida

```
void ataqueEnemigo1AHero() {
    enemy1damage = rand() % 30 + 1;
    if (statusEnemy1()) {
        HPplayer -= enemy1damage;
        cout << enemyName1 << " te ha quitado " << enemy1damage << ". Te quedan " <<
        HPplayer << " puntos de vida\n";
    }
}
```

Tenemos también la función `bucleJuego()` en la que se agrupan las funciones para darle sentido dentro del juego.

```
void bucleJuego() {
    while (playerIsAlive && (enemy1IsAlive || enemy2IsAlive)) {
        cout << "A que enemigo quieres atacar?\n[1] Ryu\n[2] Chun Li\n";
        cin >> selectEnemigo;
        if (selectEnemigo == 1) {
            seleccionAtaqueEnemy1();
        }
        else if (selectEnemigo == 2) {
            seleccionAtaqueEnemy2();
        }
        ataqueEnemigo1AHero();
        ataqueEnemigo2AHero();
        statusHero();
        statusEnemy1();
        statusEnemy2();
    }
}
```

Por último tendríamos la función `finalJuego` en la que nos diría si hemos ganado o hemos perdido y la función `main` que nos indica claramente las funciones que utiliza de manera lógica a lo que es el videojuego

```
void finalJuego() {
    if (playerIsAlive) {
        cout << "ENHORABUENA, HAS GANADO!";
    }
    else {
        cout << "Vaya... Has perdido";
    }
}

int main()
{
    srand(time(NULL));
    gamestart();
    bucleJuego();
    finalJuego();
}
```