



**FACULTAD  
DE INGENIERIA**  
Universidad de Buenos Aires

Algoritmos y Programación I  
Curso Mendez

## TP2 - Destruyendo Horrocruxes



Fecha Presentación	27/09/2018
Fecha Entrega	25/10/2018

## Introducción

Un *Horrocrux* es un objeto muy poderoso en el que un mago o bruja ha ocultado un fragmento de su alma con el propósito de alcanzar la inmortalidad.

La creación de un único *Horrocrux* permite adquirir la capacidad de resucitarse a sí mismo si su cuerpo es destruido, sin embargo la creación de *Horrocruxes* múltiples permite ser inmortal.

La creación de varios *Horrocruxes* es "costosa" para el creador, tanto por la disminución de su humanidad e incluso por la desfiguración física que conlleva.

**Es posible reconstruir el alma, pero es muy doloroso.**

Crear un *Horrocrux* es un proceso complejo, ya que se debe matar a una persona cada vez que se decida crear uno. Cuando uno mata a una persona, el alma se desgarra y así, se introduce en donde se haya decidido guardar cierta fracción de ella.

Con la ayuda del profesor *Horace Slughorn* nos enteramos que *Voldemort* creó una serie de *Horrocruxes* en los cuales hay un pedazo de su alma en cada uno.

Para poder destruirlo y así evitar que inicie su reinado de terror se deben destruir todos, entonces *Voldemort* quedará debilitado.

Nuestra misión como parte del *Ejército De Dumbledore* es encontrarlos y acabar con ellos.

## Objetivo

El presente trabajo práctico tiene como objetivo la creación de una biblioteca que sirva como núcleo para el desarrollo de un juego basado en la saga *Harry Potter*.

Se evaluará la interacción con el usuario, la forma de informar errores en sus ingresos usando una interfaz gráfica rudimentaria.

También se hará foco en el correcto uso de estructuras de control y tipos de dato compuestos: matrices, vectores y registros.

Se requiere que el trabajo cumpla con las buenas prácticas de programación profesadas por la cátedra. Se considerarán críticos la modularización y reutilización de código, la definición y el uso de constantes y la claridad del código.

## Enunciado

En esta oportunidad se quiere ayudar a derrotar *AlQueNoDebeSerNombrado* y para esto se deben destruir todos los *Horrocruxes*.

El juego se desarrolla en **cuatro** niveles, en cada uno de ellos se encuentra:

- Alguno de los *Horrocruxes* a destruir.
- Ítems que se utilizan para superar los niveles (destruir los *Horrocruxes*).
- Obstáculos (que harán más difícil la tarea).

**El objetivo es destruir el Horrocrux a lo largo de cada nivel.**

A continuación se explicará cada elemento que debe estar presente en el juego, cuáles serán sus características y su rol dentro del mismo.

## Jugador

Es el personaje que se moverá a lo largo de los niveles con el objetivo de destruir cada *Horrocrux*, deberá esquivar los obstáculos y conseguir ítems para lograr su objetivo.

- Posición: El **jugador** inicia cada nivel en el centro de la última fila.
  - Nivel 1: (4; 2).
  - Nivel 2: (6; 5).
  - Nivel 3: (4; 2).
  - Nivel 4: (6; 5).
- Movimiento: El **jugador** debe ser manejado por el usuario con las teclas **wasd**.
  - w (arriba).
  - a (izquierda).
  - s (abajo).
  - d (derecha).

No están permitidos los movimientos en diagonal.

- Turnos: Se contará con un límite de turnos en la totalidad del juego. Cada movimiento del **jugador** dará lugar a un turno. Si el **jugador** destruye todos

los *Horrocruxes* y llega a la salida del Nivel 4 antes que se terminen los turnos, habrá ganado el juego. En caso contrario se pierde el juego.

## Niveles

Los niveles son matrices de un tamaño específico donde se tiene al **jugador**, varios ítems, una *Entrada*, una *Salida*, un *Horrocrux* y un obstáculo con movimiento (excepto en el primer nivel). El objetivo de cada nivel es destruir el *Horrocrux*.

Tener en cuenta que:

- No se puede volver atrás en los niveles. Esto quiere decir que si estoy en el nivel 3 no puedo volver para el nivel 2 ni 1.
- Solo puede pasar al siguiente nivel una vez que se haya destruído el *Horrocrux* del nivel actual. Se pasa al siguiente nivel posicionándose en el casillero de salida.
- Una vez que se está en el último nivel (4) aunque se haya destruído al *Horrocrux* correspondiente (el guardapelos), se requiere que el **jugador** vaya hasta la salida para dar el juego por ganado. Esto quiere decir que en caso de destruir todos los *Horrocruxes* pero no tener turnos para llegar a la salida, el juego se da por perdido.
- Los ítems que se consiguen en un nivel son guardados y se pueden usar en cualquier nivel siguiente. *Todos desaparecen de la mochila una vez usados, excepto la espada.*

## Horrocruxes

Los *Horrocruxes* a destruir son los siguientes:

- Nivel 1: Anillo.
- Nivel 2: Diario.
- Nivel 3: Copa.
- Nivel 4: Guardapelos.

Aparecen en posiciones aleatorias dentro del nivel al que corresponden. No pueden estar sobre la posición de *Salida* ni sobre la posición de *Entrada*, tampoco sobre algún ítem u objeto.

## Modalidad de Juego

### Nivel 1

Consideraciones:

- En el nivel 1 se debe destruir el **anillo**.
- El tamaño del escenario en este caso es 5x5.
- El jugador comienza en la posición (4; 2).

El anillo se encuentra en una posición aleatoria dentro del escenario. A su vez hay 3 curas también en posiciones aleatorias. Para agarrar las curas basta con que el personaje se encuentre en la misma posición que éstas, las curas se guardan en la mochila para usar luego.

Para destruir el anillo el personaje debe pararse sobre él. Éste, envenena a quien lo destruye, por lo que debés conseguir una cura lo antes posible. En caso de que el personaje tenga la cura la usa automáticamente, de no ser así, debe encontrar una antes de que pasen 10 turnos o se termina el juego.

Una vez destruido el *Horrocrux* y usada la cura se sale del nivel dirigiéndose a la *Salida*.

		S		
			P	
A				
				P
P				

## Nivel 2

Consideraciones:

- En el nivel 2 se debe destruir el **diario**.
- El tamaño del escenario en este caso es 7x7.
- El **jugador** comienza en la posición (6; 3).

El **diario** se encuentra en una posición aleatoria dentro de los casilleros determinados por el recorrido del Basilisco, es decir que puede estar en las posiciones (3; 2), (3; 3) o (3; 4).

¡Cuidado! Hay un Basilisco suelto. El Basilisco se mueve sobre el rectángulo mostrado en la imagen, en sentido horario, arrancando desde la posición (3;5). En caso de pisar el Basilisco, si el **jugador** tiene la Espada lo mata (debe eliminarse del escenario) en caso contrario el **jugador** pierde el juego.

En la matriz hay dos ítems importantes:

- **Espada de Godric Gryffindor:** La Espada sirve para matar al Basilisco que cuida el Horrocrux .
- **Colmillos de Basilisco:** Sirven para destruir el Horrocrux .

La Espada se puede usar todas las veces que se quiera, esto significa que no se le saca la Espada al **jugador** después de usada.

Para destruir el diario el personaje debe pararse sobre él y tener uno de los colmillos de Basilisco.

Una vez destruido el Horrocrux se sale del nivel dirigiéndose a la Salida.

	F		S			
				F		F
			D			E
F	B					
						F
	F					

## Nivel 3

Consideraciones:

- En el nivel 3 se debe destruir la **copa**.
- El tamaño del escenario en este caso es 5x5.
- El **jugador** comienza en la posición (4; 2).

La **copa** se encuentra en una posición aleatoria dentro del escenario, no pudiendo estar en la fila por la que se mueve el Duende.

Sobre la fila 2 hay un Duende patrullando, que comienza en la posición (2; 2), se mueve horizontalmente de derecha a izquierda, al llegar al borde izquierdo, cambia su dirección hacia la derecha, hasta llegar al borde derecho y así sucesivamente. Si el **jugador** choca o es chocado por el Duende, perdés 7 turnos.

Es importante aclarar que el Duende no muere al chocar con el **jugador**, en el turno que están posicionados sobre el mismo casillero, debe verse el **jugador**, y el Duende estará oculto hasta que alguno se mueva.

Para destruir la **copa** el personaje debe pararse sobre ella y tener la Espada o 2 Colmillos (en caso de tener la Espada y 2 Colmillos, se usa la Espada).

¡A tener en cuenta! En caso de no tener la Espada o los 2 Colmillos para destruir el Horrocrux, no podrás volver al **Nivel 2**.

Una vez destruido el Horrocrux se sale del nivel dirigiéndose a la Salida.

		S		
C				
G				

## Nivel 4

Consideraciones:

- En el nivel 4 se debe destruir el **guardapelo**.
- El tamaño del escenario en este caso es 7x7.
- El jugador comienza en la posición (6; 3).

El guardapelo se encuentra en una posición aleatoria dentro del escenario.

Alrededor del **guardapelo** se encuentra *Dolores Umbridge* (Ugh, que pesada mujer). Si pisas a *Dolores* perdes 10 turnos.

*Dolores* gira en sentido horario alrededor del Horrocrux.

Para destruir el **guardapelo** el personaje debe pararse sobre él y tener la *Espada* o 1 *Colmillo* (en caso de tener la *Espada* y 1 *Colmillo*, se usa la *Espada*).

¡A tener en cuenta! En caso de no tener la *Espada* o 1 *Colmillo* para destruir el *Horrocrux*, no podrás volver al **Nivel 2**.

			S			



## Convenciones

Se debe usar **snake\_case** para el desarrollo del trabajo.

Cada elemento estará identificado por un caracter de acuerdo a la siguiente lista:

- Jugador: **J**.
- Anillo: **A**.
- Diario: **D**.
- Copa: **C**.
- Guardapelos: **L**.
- Cura: **P**.
- Colmillos: **F**.
- Basilisco: **B**.
- Dolores: **U**.
- Duende: **G**.
- Espada de Godric Gryffindor: **E**.
- Salida: **S**.
- Arriba: tecla **w**.
- Izquierda: tecla **a**.
- Abajo: tecla **s**.
- Derecha: tecla **d**.

Para el estado del juego:

- Jugando: **0**.
- Ganado: **1**.
- Perdido: **-1**.

## Consideraciones

### Generales:

- La *Entrada* no debe ser representada en el escenario.
- Ningún elemento puede posicionarse ni sobre la *Entrada* ni sobre la *Salida*.
- En ninguno de los niveles puede haber más de un elemento en cada casillero.
- Cuando el **jugador** está posicionado sobre la *Salida*, debe verse el **jugador**.
- Si el **jugador** se posiciona sobre la *Salida* y aún no destruyó el Horrocrux, no podrá salir del juego.
- La cantidad de turnos con que cuenta el **jugador** para destruir a todos los Horrocruxes es de 75.
- Siempre, después del movimiento de algún elemento (ya sea **jugador**, *Duende*, *Dolores* o el *Basilisco*) debe actualizarse el estado del juego, así como también reflejarse en el escenario.

### En el Nivel 1:

- Tanto el **anillo** como las *Curas*, se posicionan aleatoriamente en el escenario.
- Las *Curas* deben ser 3.
- Las *Curas* se guardan en la mochila, al usar una, se debe restar una de la mochila.
- El orden de posicionamiento debe ser, primero el **anillo**, después las *Curas*.

### En el Nivel 2:

- El **diario** se posiciona aleatoriamente en una de las 3 posiciones que quedan dentro del recorrido del *Basilisco*.
- Los *Colmillos* de basilisco son 6 y se posicionan aleatoriamente dentro del escenario.
- Los *Colmillos* de basilisco no pueden ocupar ninguna de las posiciones determinadas por el recorrido del *Basilisco*, ni dentro de éste.
- El *Basilisco* se mueve en sentido horario.
- El *Basilisco* comienza en la posición (3;5).
- Los *Colmillos* de basilisco se guardan en la mochila, al usar uno, se debe restar uno de la mochila.
- El orden de posicionamiento debe ser, primero el **diario**, después el *Basilisco*, después los *Colmillos* y después la *Espada*.
- El orden de movimiento debe ser, primero el **jugador**, después el *Basilisco*.

### En el Nivel 3:

- La **copa** se posiciona aleatoriamente en el escenario.
- La **copa** no puede posicionarse en la fila por donde se mueve el *Duende*.

- El orden de posicionamiento debe ser, primero la **copa**, después el *Duende*.
- El *Duende* se mueve horizontalmente de extremo a extremo y comienza en la posición (2; 2).
- Al colisionar el **jugador** y el *Duende*, se muestra al **jugador**, y al turno siguiente o al moverse alguno de los dos, vuelven a mostrarse ambos.
- El orden de posicionamiento debe ser, primero la **copa**, después el *Duende*.
- El orden de movimiento debe ser, primero el **jugador**, después el *Duende*.

En el **Nivel 4**:

- El **guardapelo** se posiciona aleatoriamente en el escenario.
- El **guardapelo** no puede posicionarse sobre los bordes del escenario ya que debe haber lugar para que *Dolores* circule alrededor del *Horrocrux*. Asimismo, el movimiento de *Dolores* no debe chocar contra la *Entrada* ni *Salida*.
- *Dolores* se posicionará aleatoriamente en alguna de las posiciones adyacentes al **guardapelo** y girará en sentido horario.
- El orden de posicionamiento debe ser, primero el **guardapelo**, después *Dolores*.
- Al colisionar el **jugador** y *Dolores*, se muestra al **jugador**, y al turno siguiente o al moverse alguno de los dos, vuelven a mostrarse ambos.
- El orden de movimiento debe ser, primero el **jugador**, después *Dolores*.

## Resultado Esperado

El programa creado debe:

- Determinar de forma aleatoria, una posición para cada elemento, dentro del campo, teniendo en cuenta que no puede haber dos elementos en la misma posición y que el **jugador** siempre comienza en la *Entrada*.
- Mover los elementos que tienen movimiento, a cada paso del **jugador**, según el criterio indicado para cada uno en las secciones anteriores.
- Informar la finalización del juego, ya sea porque se destruyeron todos los *Horrocruxes*, se terminaron los movimientos permitidos o no se logró alcanzar la *Cura* luego de ser envenenado.
- Actualizar a cada movimiento el estado del juego, tanto la visualización de los elementos en el nivel, como los elementos de la mochila.

- Informar al usuario, en cada momento, el nivel en el que se encuentra, el *Horrocrux* a destruir, la cantidad de pasos que le quedan y las teclas a presionar para moverse en el nivel.

## Requerimientos

El entregable que deberá presentar deberá constar de 3 archivos fuente.

- horrocrux.h
- horrocrux.c
- juego.c

## Biblioteca Horrocrux

Esta biblioteca deberá implementar el funcionamiento del juego a partir de las funciones propuestas. Éstas son las que deberán ir en el archivo de cabeceras **horrocrux.h** que será público.

De más está decir que pueden (y deben) crear sus propias funciones en **horrocrux.c** pero serán de uso interno por lo que no deben formar parte del **horrocrux.h**

Use la siguiente plantilla para su código:

### horrocrux.h

```
#ifndef __HORROCRUX_H__
#define __HORROCRUX_H__

#include <stdbool.h>

#define MAX_OBJETOS 50
#define MAX_ITEMS 50
#define CANTIDAD_NIVELES 4
#define MAX_FILAS 10
#define MAX_COLUMNAS 10

typedef struct coordenadas {
    unsigned int fila;
    unsigned int columna;
} coordenadas_t;

typedef struct elemento {
    coordenadas_t posicion;
    char codigo;
} elemento_t;

typedef struct nivel {
    unsigned int numero;
    elemento_t horrocrux;
    bool horrocrux_destruido;
}
```

```
bool tiene_guardian;
bool basilisco_destruido;
char direccion_duende; // 'd': si se mueve a la derecha - 'i': si se mueve a la izquierda.
elemento_t guardian;
coordenadas_t entrada;
coordenadas_t salida;
unsigned int filas;
unsigned int columnas;
elemento_t objetos[MAX_OBJETOS];
unsigned int cantidad_objetos;
} nivel_t;

typedef struct jugador {
    coordenadas_t posicion;
    elemento_t items[MAX_ITEMS];
    unsigned int cantidad_items;
    bool envenenado;
    unsigned int turnos_envenenado;
    unsigned int turnos_restantes;
} jugador_t;

typedef struct juego {
    jugador_t jugador;
    nivel_t niveles[CANTIDAD_NIVELES];
    unsigned int nivel_actual;
    int estado;
} juego_t;

/*
 * Inicializa el estado de todo el juego, cargando todos los valores necesarios como
 * la cantidad de movimientos, el estado del juego, las coordenadas de todos los
 * objetos, las dimensiones de los niveles, etc.
 * Precondiciones: -.
 * Postcondiciones: El juego queda en un estado válido y correcto para ser jugado.
 * Todos los elementos están dentro de los límites de su nivel.
 * El jugador se inicializa no envenenado y sin items.
 */
void inicializar_juego(juego_t *juego);

/*
 * Actualiza la matriz que representa visualmente el estado del juego en el nivel actual.
 * Precondiciones: El juego está en un estado válido.
 * Postcondiciones: Se posicionarán en la matriz aquellos elementos pertenecientes al
 * nivel actual y el jugador, los casilleros sin elementos quedan con
 * un espacio vacío.
 */
void actualizar_escenario(juego_t juego, char escenario[MAX_FILAS][MAX_COLUMNAS]);

/*
 * Actualiza la posición del jugador, así como todos los valores que deben ser modificados
 * producto de moverse a la nueva posición (estado del juego, cantidad de movimientos
 * restantes, nivel actual, etc).
 * Precondiciones: La dirección a la que se mueve es válida y está dentro de los límites
 * del campo.
 * Postcondiciones: Se devuelve el juego actualizado y válido.
 */
void mover_jugador(juego_t *juego, char direccion);

/*
```

```
* Dependiendo del nivel actual, se moverá el guardián del Horrocrux, si lo hay.
* Precondiciones: El estado del juego es válido.
* Postcondiciones: Se devuelve el juego actualizado y válido.
*/
void mover_guardian(juego_t *juego);

/*
* Muestra por pantalla la matriz correspondiente al nivel actual.
* Precondiciones: -.
* Postcondiciones: -.
*/
void mostrar_escenario(char escenario[MAX_FILAS][MAX_COLUMNAS], unsigned int filas, unsigned int
columnas);

/*
* Devuelve un entero que representa el estado del juego.
* Precondiciones: El estado del juego es válido.
* Postcondiciones: 0 si está jugando, 1 si el jugador ganó, -1 si perdió.
*/
int estado_juego(juego_t juego);

/*
* Devuelve una posición aleatoria.
* Precondiciones: -.
* Postcondiciones: Coordenada válida dentro de los límites definidos.
*/
coordenadas_t posicion_aleatoria(unsigned int filas, unsigned int columnas);

/*
* Valida si el movimiento solicitado puede realizarse.
* Precondiciones: -.
* Postcondiciones: Devuelve verdadero si el jugador puede moverse en la dirección indicada en
'movimiento' y
* falso en caso de que no pueda moverse, ya sea porque la letra es inválida o que pretende
* moverse fuera del campo.
*/
bool puede_moverse_jugador(char movimiento, juego_t juego);

#endif /* __HORROCRUX_H__ */
```

**Importante:** No modifique este archivo ya que es el contrato que  
deberán cumplir y... NO SE PUEDE ROMPER...

## Compilación y Entrega

El trabajo práctico debe ser realizado en un archivo llamado **juego.c**, y la biblioteca de funciones para jugarlo **horrocrux.c** y **horrocrux.h**, y debe poder ser compilado sin errores con el comando:

```
gcc -o horrocrux -std=c99 -Wall -Wconversion -Werror horrocrux.c juego.c
```

Por último debe ser entregado en la plataforma de corrección de trabajos prácticos Kwyjibo en la cual deberá aparecer con la etiqueta **successful**, de no tener una entrega en este estado, no se corregirá e irá directamente a reentrega, en caso de tampoco tener una entrega en estado **successful** para la reentrega, quedará fuera de la materia.

Para la entrega en Kwyjibo, recuerde que deberá subir un archivo zip conteniendo únicamente los 3 archivos antes mencionados, sin carpetas internas ni otros archivos. De lo contrario, la entrega no será validada por la plataforma.

## Bibliografía

- <http://es.harrypotter.wikia.com/wiki/Horrocrux>
- Libros de Harry Potter