



Universidad Nacional
Autónoma de México



Facultad de Ingeniería

Alumno: Castro Rodríguez Ulises

Asignatura: Fundamentos de Programación

Profesor: Marco

Antonio Martínez Quintana

Semestre 2021-1

Proyecto Final: Juego 3 en raya
programado en C

Fecha de elaboración: 20/12/2020

Resumen

En este proyecto haremos un juego sencillo de tres en raya mediante el lenguaje de programación C sin ningún apartado gráfico, es decir, será desplegado mediante la línea de comandos; en este juego el usuario que usa las fichas "X" juega contra la máquina que usa "O" como fichas, es decir una IA muy sencilla que lo que hace es escoger una casilla aleatoria desocupada para poder tirar.

Para poder desarrollar este juego, lo que hice es crear una matriz [3][3] que es básicamente el tablero, cada casilla es separada por líneas verticales y líneas horizontales, además de enumerar las casillas para poder identificarlas fácilmente.

Para poder crear este juego tuve que abarcar los temas de:

Estructuras de selección: Usadas para para poder identificar a un ganador, para poder actuar en caso de que el usuario escoja una casilla que este ocupada al igual que la IA, entre otros.

Estructuras de Repetición: Usadas para seguir repitiendo los ciclos de tiradas entre jugador y la IA, para poder enumerar las casillas del tablero y poder mostrarlo en todo momento y refrescarlo después de cada jugada.

Funciones: Usadas en todo el código, cada acción echa por una función (el mostrar el tablero junto con sus casillas enumeradas, la jugada de cualquier jugador y detectar un ganador).

Matrices: Como este juego tiene un tablero cuadrado en el que cada casilla es un valor, todo el tablero, así como todas las funciones usadas son matrices, además de que cada casilla es un valor de la matriz principal.

Introducción

El juego presentado en este proyecto básicamente es uno de los mas sencillos en los que una persona puede pensar para jugar estando en compañía, las reglas son igual de sencillas como su forma de jugarlo.

Para poder jugar se requiere formar un tablero realizando 2 líneas paralelas verticales y horizontales, así de sencillo se crea el tablero, se jugará en los espacios creados por esas líneas, que en total son nueve.

Este juego requiere 2 jugadores obligatoriamente, cada moverá una ficha diferente, generalmente se escoge primero con que ficha jugar ("X" o "O") y quien tira primero (generalmente es al azar).

El objetivo de este juego es formar una línea recta consecutiva con tus 3 fichas en el tablero, se pueden generar 3 tipos de líneas: verticales, horizontales o diagonales.

La forma de jugar es muy sencilla, un jugador pone una ficha en el tablero en alguna de las casillas, después, el segundo jugador coloca una ficha en el tablero y así sucesivamente, en caso de que haya una ficha en alguna casilla, esta casilla no podrá volver a ser escogida en toda la partida.

En este juego hay dos posibles resultados, el primer caso consta de que alguno de los jugadores forme la línea recta antes de que se acaben las casillas, y el segundo caso es que no queden casillas desocupadas y ninguno de los dos jugadores haya formado la línea recta, en este caso se considera empate.

Desarrollo

• **Descripción General**

Este proyecto está enfocado para cualquier público, se trata de un juego con fines de entretenimiento e incluso académicos pues es una muestra sencilla de lo un alumno de primer semestre es capaz de aprender con los conocimientos adecuados, además de poder llevar a la práctica todo lo aprendido en el curso.

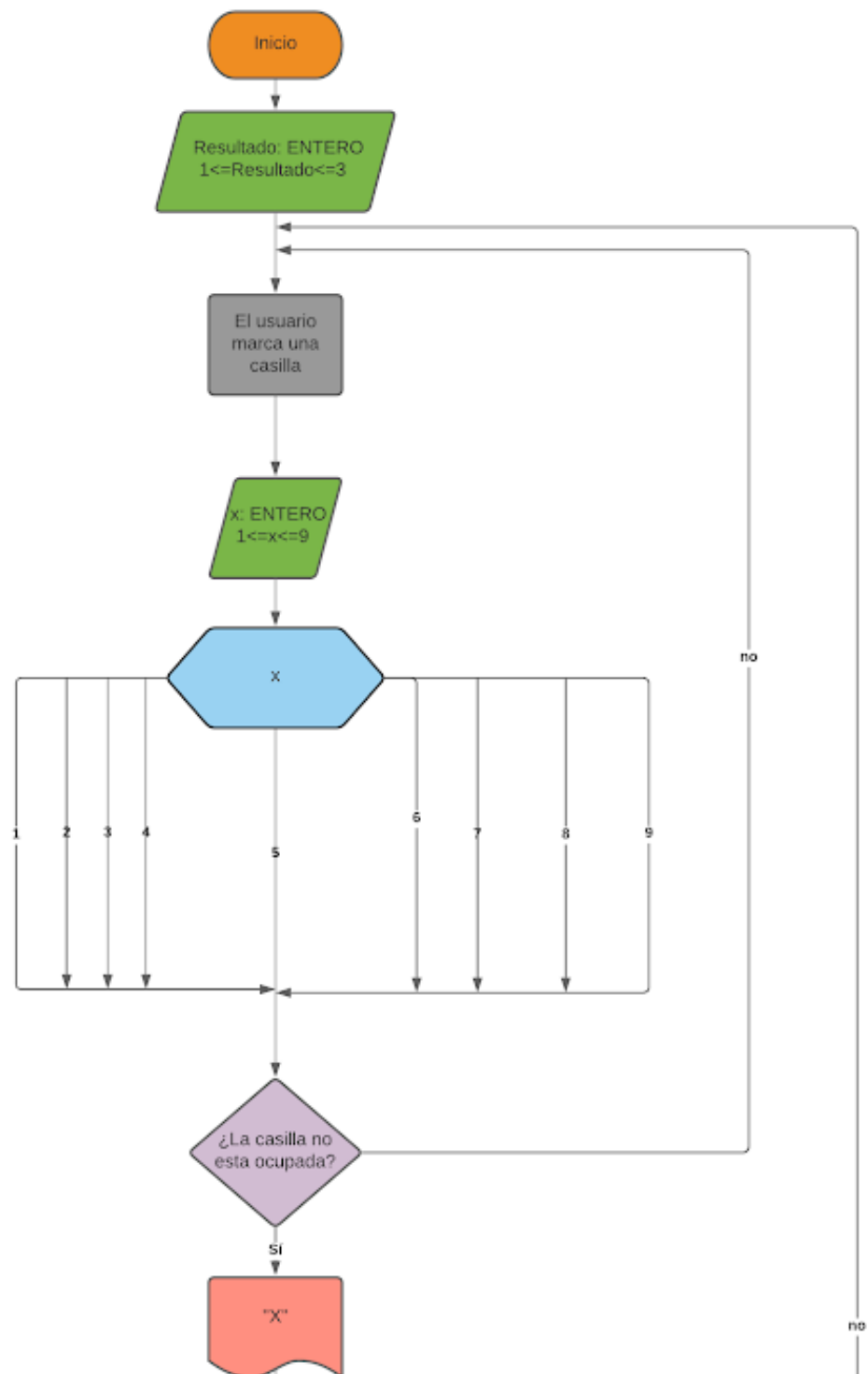
Este programa es capaz de correr en cualquier computadora, pues está en un tipo de archivo ejecutable (.exe) universal, además de tener unos requisitos mínimos, con un peso prácticamente nulo de 133 KB, este juego no contiene errores y se ejecuta de manera inmediata al abrir el ejecutable

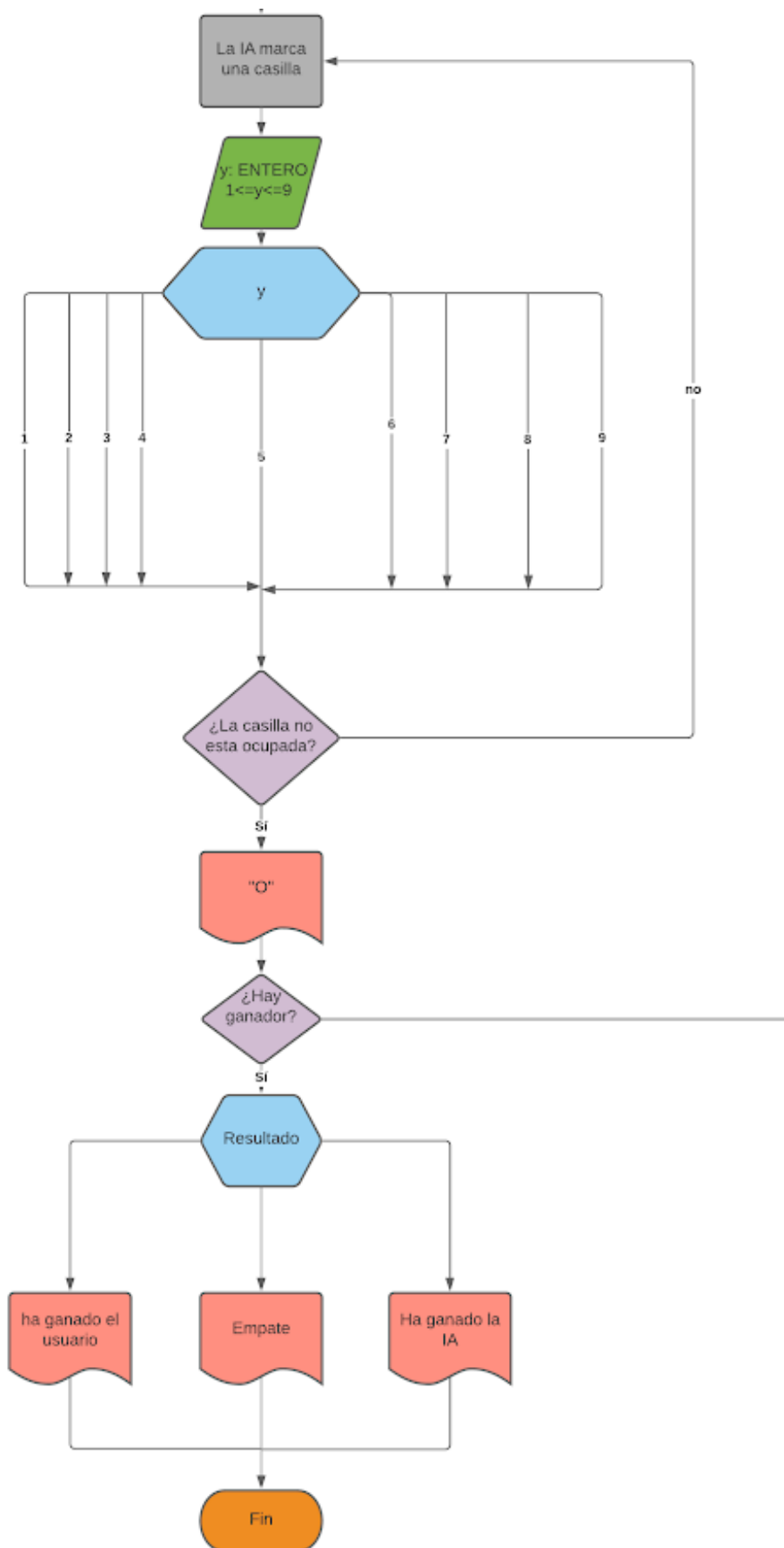
• **Algoritmo**

1. Establecer una matriz (3X3) en la cual se mostrará el tablero
2. Definir una función que se repetirá hasta que haya un ganador o hasta que se hayan ocupado todas las casillas.
3. Establecer una función para poder marcar las casillas en las que se jugara la partida.
 - a. Establecer una matriz (3X3) en la marcaremos las casillas de forma numérica.
 - b. Escoger la forma que deseamos enumerar.
4. Definir al usuario como las casillas "X" y la computadora como "O".
5. Crear una función para poder determinar las jugadas creadas por el usuario.
 - a. Pedirle al usuario que indique la casilla que desea marcar
 - i. En caso de que la casilla este ocupada, pedirle al usuario que marque otra casilla
6. Crear una función para determinar las jugadas hechas por la computadora.
 - a. Hacer que la computadora escoja una casilla al azar.

- i. En caso de que escoja una casilla ya ocupada, repetir el proceso.
- 7. Crear la función para detectar los casos en los que alguna ficha se repita 3 veces consecutivas en una línea recta, la cual será la que detecte al ganador de la partida
 - a. Hacer que la función detecte para los casos en los que las fichas formen líneas verticales, horizontales o diagonales
- 8. Mostrar al ganador de la partida.

- Diagrama de flujo





• Código fuente

```
ProyectoFinal.c
1
2 #include <stdio.h>
3 #include <time.h>
4
5 void loop (char c[3][3]);
6 void Intro_Primer (char c[3][3]);
7 void tablero (char c[3][3]);
8 void Intro_Usuario (char c[3][3]);
9 void Intro_IA (char c[3][3]);
10 int ganador(char[3][3]);
11
12 int main() //Funcion principal
13 {
14     char c[3][3];
15
16     loop (c);
17
18     system ("pause");
19     return 0;
20 }
21
22 void loop (char c[3][3])//Hace llamado a las Funciones fundamentales, refresca la pantalla
23 // y se repite hasta que haya un resultado
24 {
25     int i, j; //Declaramos un contador de tiradas y la funcion que muestra al ganador
26
27     i=0;
28
29     Intro_Primer(c); //Muestra el tablero por primera vez
30
31     do
32     {
33         system("cls");
34         tablero(c);
35         if(i%2==0) //Decide quien tira
36         {
37             Intro_Usuario(c); //Tira el usuario
38         }
39         else
40         {
41             Intro_IA(c); //Tira el Usuario
42         }
43         j=ganador(c);
44         i++;
45     }while (i<=9 && j==2);
46
47     system("cls");
48     tablero(c); //Muestra el tablero final
49
50     if (j==0) //En caso de que haya ganado el usuario
51     {
52         printf("Has ganado, Felicidades xD\n\n");
53     }
54     else if (j==1) //En caso de que haya ganado la IA
55     {
56         printf("Chale dude, has perdido :(");
57     }
58     else //en caso de un empate
59     {
60         printf("Empataste con la maquina xD, intentalo de nuevo\n\n");
61     }
```



```

62     }
63 }
64
65
66 void Intro_Primeras (char c[3][3])//Poniendo valores al tablero
67 {
68     int i,j;
69     char aux;
70     aux='1';
71
72     for (i=0; i<3; i++)           // 1 | 2 | 3
73     {                             // -----
74         for(j=0; j<3; j++)        // 7 | 8 | 9
75         {                         // -----
76             c[i][j]=aux++;        // 4 | 5 | 6
77         }
78     }
79 }
80
81 void Intro_Usuario(char c[3][3]) //Funcion del usuario
82 {
83     int i, j, k;
84     char aux;
85     do
86     {
87         //Introduccion de la casilla elegida por el usuario
88         {
89             printf("Introduzca la casilla seleccionada: ");
90             fflush (stdin);
91             scanf("%c", &aux);
92         } while( aux<'1' || aux>'9');
93
94         k=0;
95
96         switch (aux) //Aqui se detecta si la casilla esta ocupada, Un caso por casilla
97         {
98             case '1':
99             {
100                 i=0;
101                 j=0;
102                 if (c[i][j]=='X' || c[i][j]=='O')

```

```

103     {
104         k=1;
105         printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
106     }
107     break;
108 }
109 case '2':
110 {
111     i=0;
112     j=1;
113     if (c[i][j]=='X' || c[i][j]=='O')
114     {
115         k=1;
116         printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
117     }
118     break;
119 }
120 case '3':
121 {
122     i=0;
123     j=2;
124     if (c[i][j]=='X' || c[i][j]=='O')
125     {
126         k=1;
127         printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
128     }
129     break;
130 }
131 case '4':
132 {
133     i=1;
134     j=0;
135     if (c[i][j]=='X' || c[i][j]=='O')
136     {
137         k=1;
138         printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
139     }
140     break;
141 }
142 case '5':
143 {
144     i=1;
145     j=1;

```

```

146         if (c[i][j]=='X' || c[i][j]=='O')
147         {
148             k=1;
149             printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
150         }
151         break;
152     }
153     case '6':
154     {
155         i=1;
156         j=2;
157         if (c[i][j]=='X' || c[i][j]=='O')
158         {
159             k=1;
160             printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
161         }
162         break;
163     }
164     case '7':
165     {
166         i=2;
167         j=0;
168         if (c[i][j]=='X' || c[i][j]=='O')
169         {
170             k=1;
171             printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
172         }
173         break;
174     }
175     case '8':
176     {
177         i=2;
178         j=1;
179         if (c[i][j]=='X' || c[i][j]=='O')
180         {
181             k=1;
182             printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
183         }
184         break;
185     }
186     case '9':
187     {
188         i=2;
189         j=2;
190         if (c[i][j]=='X' || c[i][j]=='O')
191         {
192             k=1;
193             printf("La casilla ya esta ocupada, trata con otra casilla\n\n");
194         }
195         break;
196     }
197 }
198 }while (k==1);
199
200 c[i][j]='X'; //El tablero muestra la jugada
201 }
202
203 void Intro_IA (char c[3][3]) //Funcion de la IA oponente
204 {
205     int i, j, k;
206
207     srand (time(NULL));
208     do // Aqui la IA escoge una casilla al azar
209     {
210
211         i=rand() % 3;
212         j=rand() % 3;
213         k=0;
214
215         if (c[i][j]== 'X' || c[i][j]=='O') // La IA detecta si la casilla esta ocupada
216         {
217             k=1;
218         }
219     }while (k==1);
220     c[i][j]='O'; //El tablero muestra la jugada de la IA

```

```

221 }
222
223 void tablero (char c[3][3]) //Esta funcion se encarga de mostrar el tablero en toda la partida
224 {
225     int i,j;
226
227     for(i=0;i<3;i++)
228     {
229         for(j=0;j<3;j++)
230         {
231             if(j<2)
232             {
233                 printf(" %c |", c[i][j]); //Estas son las divisiones verticales
234             }
235             else
236             {
237                 printf(" %c ", c[i][j]);
238             }
239         }
240         if (i<2)
241         {
242             printf("\n-----\n"); //Son las divisiones en horizontales
243         }
244     }
245     printf("\n\n");
246 }
247
248 int ganador(char c[3][3])
249 {
250     if(c[0][0]=='X' || c[0][0]=='O')
251     {
252         if (c[0][0]==c[0][1] && c[0][0]==c[0][2])
253         {
254             if (c[0][0]=='X')
255             {
256                 return 0; //El usuario ha ganado
257             }
258             else
259             {
260                 return 1; //Ha perdido el usuario
261             }

```

```

262     }
263     if (c[0][0]==c[1][0] && c[0][0]==c[2][0])
264     {
265         if (c[0][0]=='X')
266         {
267             return 0; //El usuario ha ganado
268         }
269         else
270         {
271             return 1; //Ha perdido el usuario
272         }
273     }
274 }
275
276 if(c[1][1]=='X' || c[1][1]=='O')
277 {
278     if (c[1][1]==c[0][0] && c[1][1]==c[2][2])
279     {
280         if (c[1][1]=='X')
281         {
282             return 0; //El usuario ha ganado
283         }
284         else
285         {
286             return 1; //Ha perdido el usuario
287         }
288     }
289     if (c[1][1]==c[1][0] && c[1][1]==c[1][2])
290     {
291         if (c[1][1]=='X')
292         {
293             return 0; //El usuario ha ganado
294         }
295         else
296         {
297             return 1; //Ha perdido el usuario

```

```

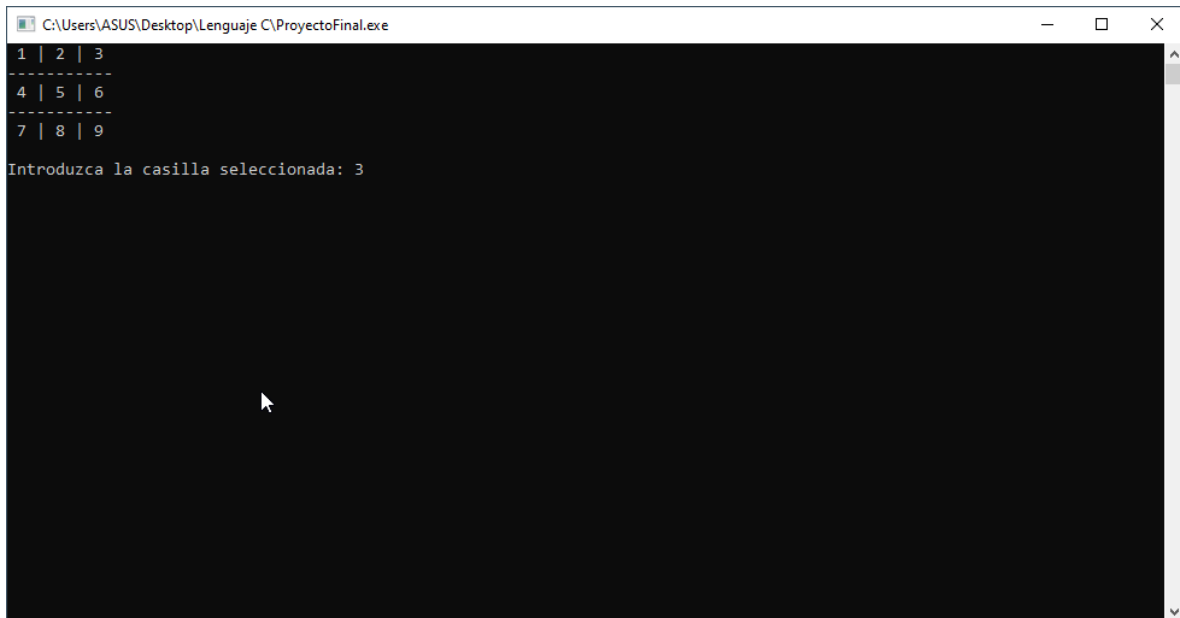
298     }
299     }
300     if (c[1][1]==c[2][0] && c[1][1]==c[0][2])
301     {
302         if (c[1][1]=='X')
303         {
304             return 0; //El usuario ha ganado
305         }
306         else
307         {
308             return 1; //Ha perdido el usuario
309         }
310     }
311     if (c[1][1]==c[0][1] && c[1][1]==c[2][1])
312     {
313         if (c[1][1]=='X')
314         {
315             return 0; //El usuario ha ganado
316         }
317         else
318         {
319             return 1; //Ha perdido el usuario
320         }
321     }
322 }
323 if (c[2][2]=='X' || c[2][2]=='O')
324 {
325     if (c[2][2]==c[2][0] && c[2][2]==c[2][1])
326     {
327         if (c[2][2]=='X')
328         {
329             return 0; //El usuario ha ganado
330         }
331         else
332         {
333             return 1; //Ha perdido el usuario
334         }
335     }
336 }
337 if (c[2][2]==c[0][2] && c[2][2]==c[1][2])
338 {
339     if (c[2][2]=='X')
340     {
341         return 0; //El usuario ha ganado
342     }
343     else
344     {
345         return 1; //Ha perdido el usuario
346     }
347 }
348 return 2; //Empate entre el usuario y la maquina
349 }
350

```

Resultados

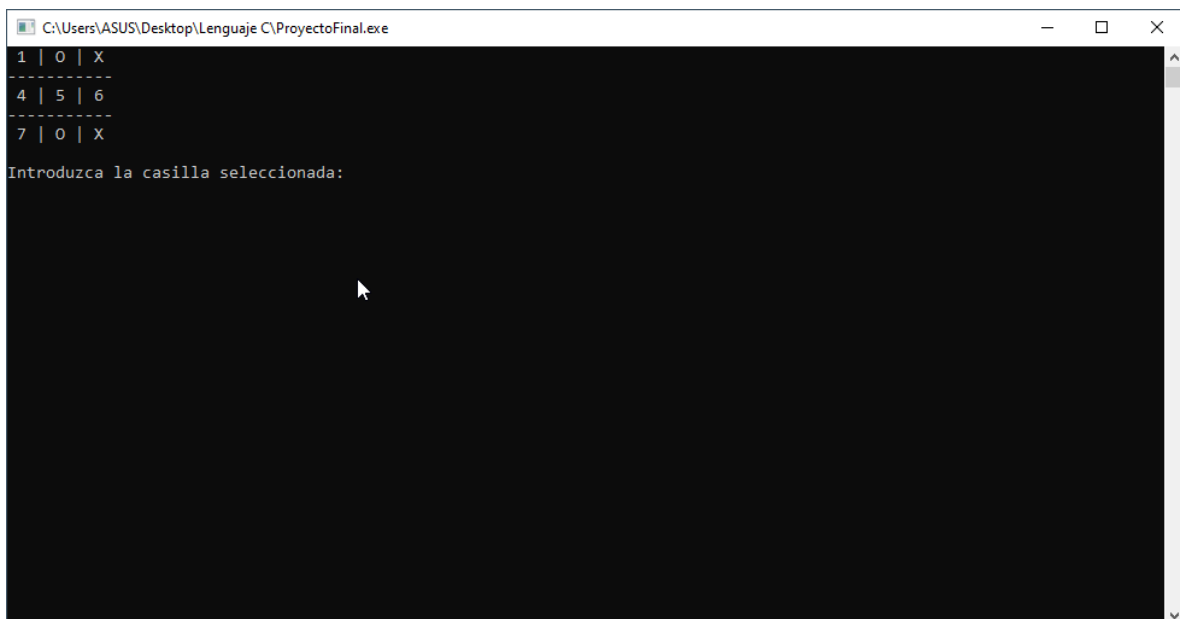
- **Funcionamiento del programa**

En esta captura se muestra el inicio del programa:



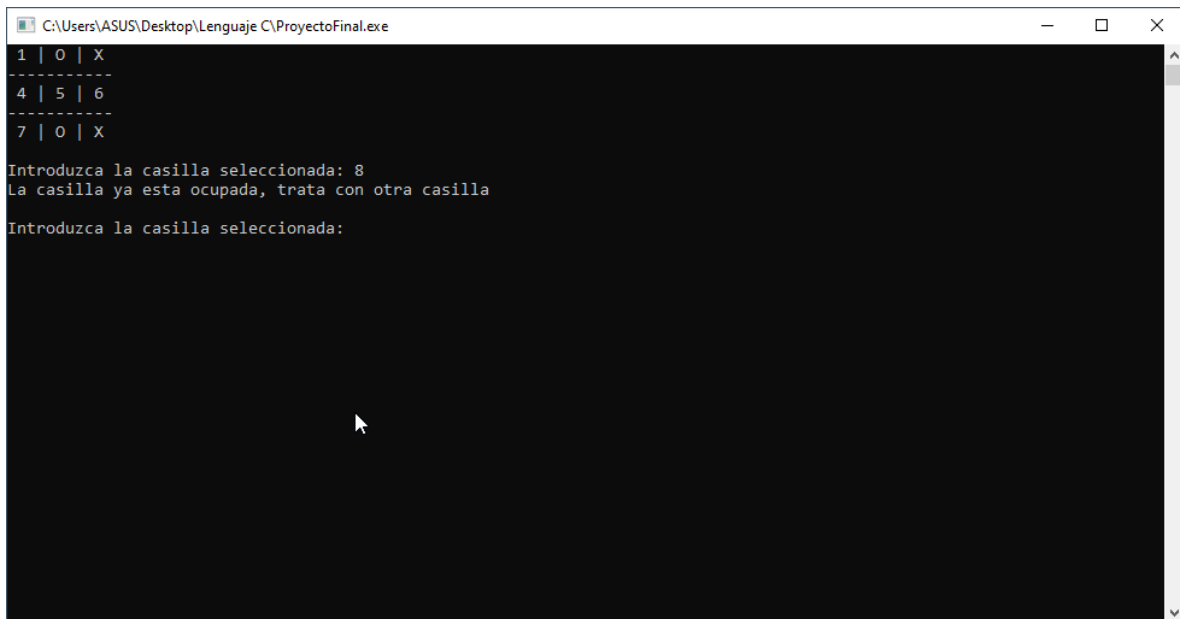
```
C:\Users\ASUS\Desktop\Lenguaje C\ProyectoFinal.exe
1 | 2 | 3
-----
4 | 5 | 6
-----
7 | 8 | 9
Introduzca la casilla seleccionada: 3
```

Se muestra el funcionamiento de como se muestra las tiradas del usuario, así como de la IA programada:



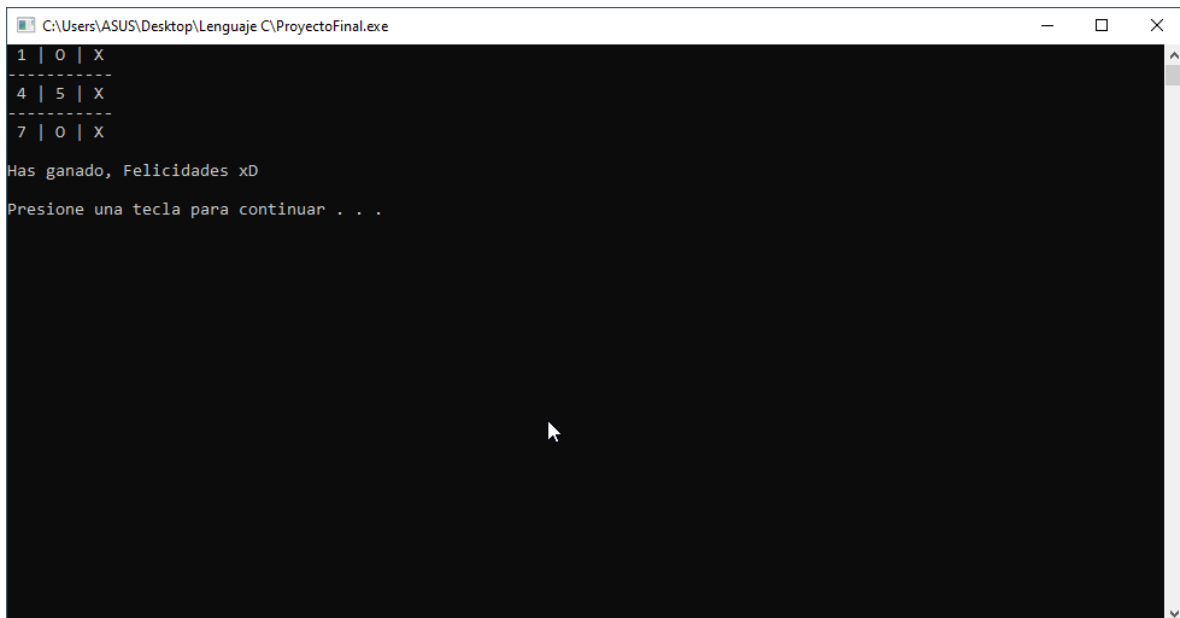
```
C:\Users\ASUS\Desktop\Lenguaje C\ProyectoFinal.exe
1 | O | X
-----
4 | 5 | 6
-----
7 | O | X
Introduzca la casilla seleccionada:
```

Se muestra el caso en el que el usuario desea marcar una casilla ocupada, por lo que se le pide que marque una diferente:



```
C:\Users\ASUS\Desktop\Lenguaje C\ProyectoFinal.exe
1 | 0 | X
-----
4 | 5 | 6
-----
7 | 0 | X
Introduzca la casilla seleccionada: 8
La casilla ya esta ocupada, trata con otra casilla
Introduzca la casilla seleccionada:
```

Por ultimo se muestra el resultado, en este caso el usuario gano, por lo que muestra el mensaje ganador:



```
C:\Users\ASUS\Desktop\Lenguaje C\ProyectoFinal.exe
1 | 0 | X
-----
4 | 5 | X
-----
7 | 0 | X
Has ganado, Felicidades xD
Presione una tecla para continuar . . .
```

- **Recursos informáticos [software y hardware]**

Diciembre (2020)	Enero (2021)
Computadora de escritorio Compilador GCC Notepad++ (programa)	Computadora de escritorio Dev C++

- **Costos asociados:** Básicamente como en este proyecto se trata de un juego básico en C, no hay costos asociados al proyecto, pues solo fue necesario el uso de mi computadora y use programas gratuitos como Dev C++ para programar.

Canal donde se encuentra el video acerca de este proyecto:

<https://www.youtube.com/channel/UCVCg18rgE4vR4XVwui48SWA>

Enlace del repositorio en GitHub:

<https://github.com/lulascas12/ProyectoFinal>

Conclusiones

Al realizar este proyecto he podido reflexionar un poco más acerca de GNU y la FSF, he cambiado de opinión, la opción de que sea un software depende de la persona que trabaje en él, ya que al realizar este proyecto me di cuenta de la dificultad de realizar un programa es más alta de lo que pensé, sin embargo, al dejar esta decisión el criterio del creador del programa no podemos afectarlo negativamente de alguna manera.

Ya al final me di cuenta de que el mundo de la programación es más complejo de lo que pensé, pero me sentí realizado al momento de que termine de programar y lo ejecute por primera vez sin errores, espero poder seguir aprendiendo a partir de esta experiencia, es por eso que decidí cambiar de proyecto, actualmente no me siento con las capacidades para crearla, quiero seguir aprendiendo no solo acerca de los temas más complejos, sino de poder aprender otros lenguajes de programación, tal es el caso de HTML pues, me mantengo firme con mi decisión, para el final de la carrera deseo poder crear una página web desde cero y que funcione perfectamente como lo he soñado.

Bibliografía

- “Bloque 1.1 Juegos Simples en C: Tres en Raya”, Empieza a programar (28/09/2013) Consultado el (01/01/2021) de <https://www.youtube.com/watch?v=dakMghNMjwE>
- “Bloque 1.2 Juegos Simples en C: Tres en Raya”, Empieza a programar (29/09/2013) Consultado el (08/01/2021) de <https://www.youtube.com/watch?v=dakMghNMjwE>
- “Bloque 1.1 Juegos Simples en C: Tres en Raya”, Empieza a programar (30/09/2013) Consultado el (15/01/2021) de <https://www.youtube.com/watch?v=dakMghNMjwE>
- Zambrano, R. (2011, 28 octubre). *13 juegos Cásicos desarrollados en lenguaje C*. 13 juegos clásicos desarrollados en C. <http://www.lapolitecnica.net/c/practicas-c/48-juegosC>
- Apuntes de Informática educativa de Bachillerato.
- *Organización de la Referencia del lenguaje C*. (2016, 4 noviembre). Microsoft Docs. <https://docs.microsoft.com/es-es/cpp/c-language/organization-of-the-c-language-reference?view=msvc-160>