



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE MÉXICO



Facultad de Ingeniería  
Ingeniería en computación

## *Juego "Gato"*

Alumna: Cuevas Antúñez Samantha  
Semestre 2020-1

Profesor: Marco Antonio Martínez Quintana.

Fundamentos de programación

30 noviembre 2020- 30 enero 2021

# Juego "Gato"

En este trabajo se abordarán e implementarán los temas vistos en la materia de Fundamentos de programación aplicados en el juego de tres en raya, mejor conocido como gato. El juego del gato es un juego que todos hemos jugado al menos una vez en la vida, a pesar de no ser un juego complejo trae consigo horas de diversión.

El juego se juega en una cuadrícula de 3 por 3, se juega entre 2 participantes, a un jugador se le asigna el símbolo "X" y al otro jugador, el símbolo "O". En su turno, cada jugador debe poner su símbolo en un cuadrado vacío, si un jugador tiene tres de sus símbolos en un renglón, columna o diagonal, gana el juego y también se puede dar el caso de que se dé un empate, este ocurre si no quedan cuadrados vacíos y ningún jugador ha ganado.

A continuación, podrá abordarse a mayor detalle mi evolución cursando el primer semestre de ingeniería en computación, el cual al ser el primero lo considero bastante relevante debido a que en los primeros semestres se aprenden las bases para en un futuro aplicarlas a mayor escala. Más allá de la evolución en el ámbito de programación como tal, también se dio una reflexión; ya que muchas veces uno no le da el reconocimiento que deberán a los creadores de videojuegos, parece una tarea hasta cierto punto fácil pero definitivamente no lo es, no basta con saber programar, también se necesita tener imaginación, para que conforme se avanza se pueda hacer el código lo más eficaz posible.

Cuando se programa es muy ocurrente que se den errores desde la lógica aplicada hasta la síntesis incorrecta, por ello es de suma importancia el estar estudiando constante desde las librerías que se deben invocar, hasta el tipo de dato que se debe pedir. Algo sumamente curioso del juego del gato es que a la vista del usuario es un juego de lo más sencillo, pero programarlo si se lleva su esfuerzo. De igual forma en el trabajo se muestra todo el proceso y análisis que se elabora antes de empezar a programa, debe haber una organización y planificación que nos ayude a optimizar nuestros tiempos.

## Introducción

El 2020 fue un año muy sorpresivo, con la llegada del COVID-19 cambio prácticamente el ritmo de vida al que estábamos acostumbrados. Este virus comenzaba a sonar desde diciembre del 2019 pero los casos fueron aumentando exponencialmente, a tal grado de que para marzo del 2020 comenzó una cuarentena que se ha extendido hasta el día de hoy. Esta pandemia ha afectado a todo el mundo en todos los ámbitos además de las millones de defunciones alrededor del mundo, grandes países pararon sus actividades esenciales por un tiempo prolongado afectando la economía, salud mental y estabilidad en todo el mundo.

A pesar de esta pandemia en la que actualmente vivimos hemos tratado de llevar las cosas con mayor normalidad, inclusive se implementaron las clases en línea que, aunque no son lo ideal nos hemos intentado adaptar para sobrellevar y no interrumpir nuestros estudios. Todos estos entornos han propiciado que trastornos como la ansiedad y depresión hayan aumentado considerablemente tanto en adultos, como adolescentes. Lo anterior es algo entendible, nadie estaba preparado mentalmente para un cambio así.

No obstante, afrontando todos los obstáculos, las clases modalidad a distancia han implicado un gran esfuerzo tanto de los profesores, autoridades y alumnos, ya que no todos contamos con las mismas herramientas y posibilidades. Pero a pesar de esto, las clases en línea han tenido su impacto negativo ya que al ser en línea los estudiantes tienden a estresarse más por el hecho de que se mantiene todo el día en su casa, aunque parezca que no, el hecho de trasladarse de la casa a la escuela y de regreso disminuía ese estrés.

El estar tanto tiempo en casa también a propiciado el poder aprender cosas nuevas, cada día las plataformas que en un inicio se usaban como ocio han ido actualizándose para proporcionar algo más al usuario, es por esto por lo que personalmente programar un gato, ya que al inicio del curso mis conocimientos de programación eran nulos, pero, conforme iba avanzando el curso y con ayuda de videos en YouTube pude ir realizando el programa. Considero que el buscar una actividad extra puede ayudar considerablemente al estrés y además de ello nos ayuda a aprender cosas

nuevas, como en mi caso que de hace unos meses no sabía ni programar un hola mundo y actualmente programe mi primer videojuego.

## **Desarrollo**

A pesar de ser un juego relativamente sencillo considero que al ser un juego clásico que al generar nostalgia a los usuarios estos lo jugarían, inclusive empresas medianas o pequeñas de videojuegos podrían implementarlos, inclusive hablando a pequeña escala podría ya sea venderlo o rentárselo a páginas dedicadas a la distribución de videojuegos.

De igual forma este programa podría servir como ejemplos para programadores principiantes, ya que para comprender su funcionamiento es necesario dominar lo principal del lenguaje de; estructuras de selección, repetición, funciones, impresión de datos en pantalla, tipo de datos, declaración de variable. Por ende, considero que la rentabilidad del código para el aprendizaje sería una opción más viable, por el hecho de que además de sacarle provecho podría ayudar a estudiantes a resolver dudas o simplemente tomarlo de ejemplo.

Teniendo en cuenta el hecho de que ya existen miles de programas con el mío, considero que es un gran ejercicio para comprobarse a uno mismo si cuenta con las bases necesarias del lenguaje C.

## **Algoritmo completo o parcial de la solución**

Jugar al gato

1. Inicie el programa
2. Indicar la opción que desea hacer
  - 2.1 (Opción 1) Inicio del juego
  - 2.2 Ingrese un número dependiendo de donde desee colocar la X
  - 2.3 Esperar que el programa juegue su turno

- 2.4 Intentar formar una línea corrida de X, ya sea diagonal, vertical u horizontal
- 2.5 Ir intercambiando turnos con el programa hasta que haya un ganador o se produzca un empate.
- 2.6 Volver a elegir una opción para jugar de nuevo o salir.
- 3. (Opción 2) Agradecimientos y salir.
  - 3.1 Leer el texto presentado en la plataforma

### **Pseudocódigo completo o parcial**

---

INICIO

op: ENTERO

Ganador[3][3]: CHARACTER

Aa,ae,ai,ao,au: CHARACTER

FUNC main (vacío) RET: vacío

    ESCRIBIR "Hola que gusto verte uwu"

    j: ENTERO

    c [3][3]: ENTERO // 9 variables de tipo entero

    HACER

        ESCRIBIR "1)Jugar 2)Agradecimientos y salir"

        ESCRIBIR "Elige una opción"

        Leer op

        SELECCIONAR (op) EN

            CASO 1 ->

                FUNC loop

CASO 2 ->

ESCRIBIR "Quiero agradecerle a mi buen amig Jos%c  
Alfredo por ayudar a que esto fuera posible"

DEFECTO ->

ESCRIBIR "opci%cn no v%clida"

FIN SELECCIONAR

MIENTRAS op != 2

ESCRIBIR "Gracias por jugar conmigo, hasta luego uwu"

RET 0

FIN FUNC

FUNC loop (c [3][3]: ENTERO, j:ENTERO)

I:ENTERO

I : =0

FUNC Intro\_num (c)

HACER

FUN system (limpiar)

FUN tablero (c)

SI (i % 2==0) ENTONCES

FUN Intro\_yo (c)

FIN SI

DE LO CONTRARIO

FUN Intro\_ia (c)

FIN DE LO CONTRARIO

J=Ganador (c)

I ++1

MIENTRAS (i<=8 y j==2)

FUN system (limpiar)

FUN tablero (c)

SI (j==0)

    ESCRIBIR "Felicidades, has ganado uwu"

FIN SI

DE LO CONTRARIO (j==1)

    ESCRIBIR "Suerte para la próxima, has perdido"

FIN DE LO CONTRARIO

FIN FUNC

FUNC Intro\_yo (c [3][3]: ENTERO)

    I,j,k: ENTERO

    Aux: CARÁCTER

    HACER

        HACER

            ESCRIBIR "Coloca una ficha"

            FUN fflush (stdin)

        MIENTRAS (aux < 1 || aux > 9)

        K=0

SELECCIONAR (op) EN

CASO 1 ->

i= 0

j=0

SI

K=1

ESCRIBIR "La casilla está ocupada. Intenta con otro  
numero")

FIN SI

CASO 2 ->

i= 0

j=1

SI

K=1

ESCRIBIR "La casilla está ocupada. Intenta con otro  
numero")

FIN SI

CASO 3 ->

i= 0

j = 2

SI

K=1



numero”

ESCRIBIR “La casilla está ocupada. Intenta con otro

FIN SI

CASO 4->

i= 1

j = 0

SI

K=1

ESCRIBIR “La casilla está ocupada. Intenta con otro

numero”)

FIN SI

CASO 5 ->

i= 1

j = 1

SI

K=1

ESCRIBIR “La casilla está ocupada. Intenta con otro

numero”)

CASO 6->

i= 1

j = 2

SI

K=1

numero")

ESCRIBIR "La casilla está ocupada. Intenta con otro

CASO 7->

i= 2

j = 0

SI

K=1

ESCRIBIR "La casilla está ocupada. Intenta con otro

numero")

CASO 8->

i= 2

j = 1

SI

K=1

ESCRIBIR "La casilla está ocupada. Intenta con otro

numero")

CASO 9->

i= 2

j = 2

SI

K=1

ESCRIBIR "La casilla está ocupada. Intenta con otro

numero")

MIENTRAS (K==%)

C[i][j] =X

FIN MIENTRAS

FIN SI

MIENTRAS

FIN

## Código fuente COMENTADO del proyecto completo

```
1  #include<stdio.h>
2  #include <time.h>
3  #include <stdlib.h>
4
5  void loop (char c [3][3], int j);
6  void Intro_num(char c[3][3]);
7  void tablero (char c[3][3]);
8  void Intro_yo (char c[3][3]);
9  void Intro_IA(char c[3][3]);
10 int Ganador (char[3][3]);
11 int op;
12 char aa=160, ae=130, ai=161, ao=162, au=163, sp=168;
13
14
15 int main()
16 {
17     //Mensaje de bienvenida
18     printf("\n\tHola que gusto verte uwu\n\n");
19
20     int j;
21     char c [3][3];
22
23     do
24     {
25         //mostrar el menú
26         printf("\n1) Jugar\n2) Agradecimientos y salida\n");
27
28         //Solicitar la opción
29         printf("Elige una opción: ",ao);
30         scanf("%d",&op);
31         switch (op)
32         {
33             case 1: //Jugar
34                 loop (c,j); //llama a las funciones fundamentales y hace un reflejo de pantalla
35                 break;
36             case 2: //Agradecimientos y salida
37                 printf("Quiero agradecerle a mi buen amigo Jos%c Alfredo por ayudar a que esto fuera posible",ae);
38
39                 break;
40             default:
41                 printf("Opción no válida\n",ao,aa);
42                 break;
43         }
44     }while(op!=2);
45     printf("\n\tGracias por jugar conmigo, hasta luego uwu \n");
46     return 0;
47
48 }
49
50
51
52 void loop (char c[3][3], int j)
53 {
54     int i;
55     i = 0;
56     Intro_num(c);
57     do
58     {
59         system("cls");
60         tablero(c);
61
62         if (i % 2 == 0)
63         {
64             Intro_yo(c);
65         } else{
66             Intro_IA(c);
67         }
68         j = Ganador(c);
69         i++;
70     } while(i<=8 && j==2);
71
72     system("cls");
73
74 }
```

```

74     system("cls");
75     tablero(c);
76     if(j == 0){
77         printf("\tFelicitades, has ganado uwu\n\n");
78     }
79     else if(j == 1) {
80         printf("\tSuerte para la pr\u00e1xima, has perdido :(\n\n",ao);
81     }
82     else{
83         printf("\tBuen juego, hemos empatado\n\n");
84     }
85 }
86
87
88 void Intro_num (char c[3][3])
89 {
90     int i,j;
91     //Metemos valores a la matriz
92     char aux ;
93     aux = '1';
94
95     for(i=0;i<3;i++)
96     {
97         for(j=0;j<3;j++)
98         {
99             c[i][j]= aux++; //dando valores a la matriz
100         }
101     }
102 }
103
104 void Intro_yo (char c[3][3])
105 {
106     int i, j, k; //variables de tipo entero
107     char aux; //Declarar variable de tipo caracter
108

```

```

109     do
110     {
111         do //Número menor que 1 y mayor a 9
112         {
113             printf("Coloca una ficha: ");
114             fflush(stdin);
115             scanf("%c", &aux);
116         } while (aux < '1' || aux > '9' );
117
118         k = 0;
119
120         switch (aux) //Núm repetido
121         {
122             case '1':
123                 i = 0;
124                 j = 0;
125                 if (c[i][j] == 'X' || c[i][j] == 'O')
126                 {
127                     k = 1;
128                     printf("La casilla est\u00e1 ocupada\nIntenta con otro n\u00famero\n\n",aa,au);
129                 }
130                 break;
131             case '2':
132                 i = 0;
133                 j = 1;
134                 if (c[i][j] == 'X' || c[i][j] == 'O')
135                 {
136                     k = 1;
137                     printf("La casilla est\u00e1 ocupada\nIntenta con otro n\u00famero\n\n",aa,au);
138                 }
139                 break;
140             case '3':
141                 i = 0;
142                 j = 2;
143                 if (c[i][j] == 'X' || c[i][j] == 'O')
144                 {
145                     k = 1;
146                     printf("La casilla est\u00e1 ocupada\nIntenta con otro n\u00famero\n\n",aa,au);
147                 }
148                 break;
149             case '8':
150                 i = 2;
151                 j = 1;
152                 if (c[i][j] == 'X' || c[i][j] == 'O')
153                 {
154                     k = 1;
155                     printf("La casilla est\u00e1 ocupada\nIntenta con otro n\u00famero\n\n",aa,au);
156                 }
157                 break;
158             case '9':
159                 i = 2;
160                 j = 2;
161                 if (c[i][j] == 'X' || c[i][j] == 'O')
162                 {
163                     k = 1;
164                     printf("La casilla est\u00e1 ocupada\nIntenta con otro n\u00famero\n\n",aa,au);
165                 }
166                 break;
167         }
168     } while (k == 1);
169
170     c[i][j] = 'X';
171 }
172
173 void Intro_IA (char c[3][3])
174 {
175     int i, j, k;
176
177     srand(time(NULL));
178
179     do
180

```

```

147     }
148     break;
149     case '4':
150     i= 1;
151     j=0;
152     if (c[i][j]!='X' || c[i][j] == 'O')
153     {
154         k = 1;
155         printf("La casilla est%c ocupada\nIntenta con otro n%cmero\n\n",aa,au);
156     }
157     break;
158     case '5':
159     i= 1;
160     j =1;
161     if (c[i][j]!='X' || c[i][j] == 'O')
162     {
163         k = 1;
164         printf("La casilla est%c ocupada\nIntenta con otro n%cmero\n\n",aa,au);
165     }
166     break;
167     case '6':
168     i= 1;
169     j =2;
170     if (c[i][j] == 'X' || c[i][j] == 'O')
171     {
172         k = 1;
173         printf("La casilla est%c ocupada\nIntenta con otro n%cmero\n\n",aa,au);
174     }
175     break;
176     case '7':
177     i= 2;
178     j =0;
179     if (c[i][j]!='X' || c[i][j] == 'O')
180     {
181         k = 1;

```

```

219     {
220         i = rand()%3;
221         j= rand()%3;
222         k = 0;
223
224         if(c[i][j] == 'X' || c[i][j] == 'O')
225         {
226             k = 1;
227         }
228     } while (k == 1);
229
230     c[i][j] = 'O';
231 }
232
233 void tablero (char c[3][3])
234 {
235     int i,j;
236     for(i=0;i<3;i++)
237     {
238         for(j=0;j<3;j++)
239         {
240             if(j<2)
241             {
242                 printf (" %c |",c[i][j]);
243             }
244             else
245             {
246                 printf (" %c ",c[i][j]); //Para que el 3 no tenga |
247             }
248         }
249         if(i<2) //para que no salga una --- de más
250         {
251             printf ("\n-----\n");
252         }
253     }
254     printf("\n\n");

```

```

254     printf("\n\n");
255 }
256
257 int Ganador (char c[3][3]){
258     if (c[0][0] == 'X' || c[0][0] == 'O'){
259         if (c[0][0] == c[0][1] && c[0][0] == c[0][2]){
260             if (c[0][0] == 'X'){
261                 return 0; //He ganado
262             }
263             else {
264                 return 1; //He perdido
265             }
266         }
267         if (c[0][0] == c[1][0] && c[0][0] == c[2][0]){
268             if (c[0][0] == 'X'){
269                 return 0; //He ganado
270             }
271             else {
272                 return 1; //He perdido
273             }
274         }
275     }
276
277     if (c[1][1] == 'X' || c[1][1] == 'O'){
278         if (c[1][1] == c[0][0] && c[1][1] == c[2][2]){
279             if (c[1][1] == 'X'){
280                 return 0; //He ganado
281             }
282             else {
283                 return 1; //He perdido
284             }
285         }
286         if (c[1][1] == c[1][0] && c[1][1] == c[1][2]){
287             if (c[1][1] == 'X'){
288                 return 0; //He ganado
289             }
290             else {

```

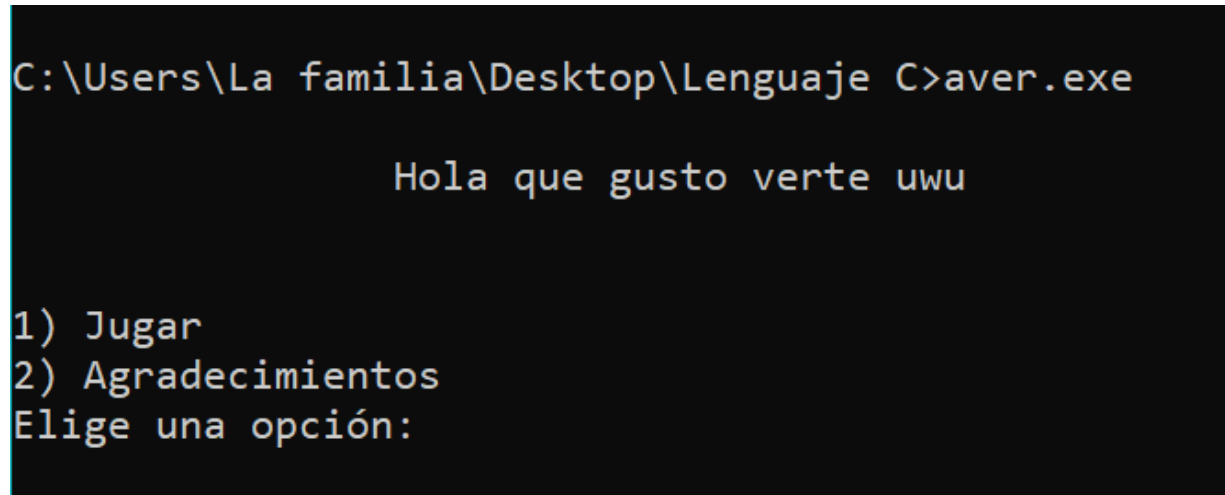
```

291         return 1; //He perdido
292     }
293 }
294 if(c[1][1] == c[2][0] && c[1][1] == c[0][2]){
295     if(c[1][1] == 'X'){
296         return 0; //He ganado
297     }
298     else {
299         return 1; //He perdido
300     }
301 }
302 if(c[1][1] == c[0][1] && c[1][1] == c[2][1]){
303     if(c[1][1] == 'X'){
304         return 0; //He ganado
305     }
306     else {
307         return 1; //He perdido
308     }
309 }
310 }
311 if(c[2][2] == 'X' || c[2][2] == 'O'){
312     if(c[2][2] == c[2][0] && c[2][2] == c[2][1]){
313         if(c[2][2] == 'X'){
314             return 0; //He ganado
315         }
316         else {
317             return 1; //He perdido
318         }
319     }
320     if(c[2][2] == c[0][2] && c[2][2] == c[1][2]){
321         if(c[2][2] == 'X'){
322             return 0; //He ganado
323         }
324         else {
325             return 1; //He perdido
326         }
327     }
328 }
329 return 2;
330 }

```

## Resultados

Capturas de pantalla del funcionamiento de su proyecto



```

C:\Users\La familia\Desktop\Lenguaje C>aver.exe

        Hola que gusto verte uwu

1) Jugar
2) Agradecimientos
Elige una opción:

```

Una vez iniciado el programa nos abre un menú con 2 opciones.

```
C:\Users\La familia\Desktop\Lenguaje C>aver.exe

                Hola que gusto verte uwu

1) Jugar
2) Agradecimientos
Elige una opción: 1
```

Cuando elegimos la opción 1 se limpia la pantalla:

```
1 | 2 | 3
--
4 | 0 | 6
--
X | 8 | 9

Coloca una ficha: 9
```

Al nosotros  
presionar una tecla  
se pone una X en  
dicha posición

```
1 | 2 | 0
--
4 | 0 | 6
--
X | 8 | X

Coloca una ficha: 8_
```

En cuanto nosotros  
colocamos una ficha la  
“computadora” coloca  
una ficha, ya que como  
sabemos en el gato se  
va jugando turno por  
turno

```
Simbolo del sistema - aver.exe
1 | 2 | 0
-----
4 | 0 | 6
-----
X | X | X

Felicitades, has ganado uwu

1) Jugar
2) Agradecimientos
Elige una opción:
```

Cuando ganamos se muestra una felicitación y podemos volver a elegir las 2 opciones del menú

```
X | X | 0
-----
0 | 0 | X
-----
0 | 8 | X

Suerte para la próxima, has perdido :(

1) Jugar
2) Agradecimientos
Elige una opción:
```

Además, también nos sale un aviso cuando perdemos.

De igual forma, al momento de querer introducir una ficha en un lugar donde está ocupado nos sale la advertencia:

```
Simbolo del sistema - aver.exe
1 | 2 | 0
-----
0 | 5 | X
-----
7 | 8 | X

Coloca una ficha: 4
La casilla está ocupada
Intenta con otro número

Coloca una ficha: 4
La casilla está ocupada
Intenta con otro número

Coloca una ficha: 6
La casilla está ocupada
Intenta con otro número

Coloca una ficha: _
```



```
O | X | O
-----
X | O | X
-----
X | O | X

      Buen juego, hemos empatado

1) Jugar
2) Agradecimientos
Elige una opción: 2
```

Aviso que aparece cuando existe un empate

```
O | X | O
-----
X | O | X
-----
X | O | X

      Buen juego, hemos empatado

1) Jugar
2) Agradecimientos
Elige una opción: 2
Quiero agradecerle a mi buen amigo José Alfredo por ayudar a que esto fuera posible
      Gracias por jugar conmigo, hasta luego uwu

C:\Users\La familia\Desktop\Lenguaje C>_
```

Al elegir la opción 2 sale un pequeño agradecimiento, se sale del programa y además aparece una despedida

Tabla comparativa de recursos informáticos [software y hardware] necesarios para llevar a cabo el proyecto

NOVIEMBRE 2020	ENERO 2021
<ul style="list-style-type: none"><li>• Tener una computadora con gran capacidad</li><li>• Servicio de alojamiento</li><li>• Cursos de criptografía, diseño, programación</li><li>• Aprender a programar</li></ul>	<ul style="list-style-type: none"><li>➤ Buen manejo del lenguaje en C</li><li>➤ Entender como funcionan las estructuras de selección y repetición</li><li>➤ Saber imprimir en pantalla, declarar variables</li><li>➤ Comprender el uso de funciones</li></ul>

Evidentemente en un principio del curso el haber cambiado de proyecto y tener nulos conocimientos acerca de programar no estaba consciente de que lo implicaba un videojuego. Ya que a pesar de que en este caso el usuario solo coloca fichas el programa está haciendo variadas funciones, unas para identificar dónde ya hay fichas, en donde elige poner la ficha el usuario, cuando gana, pierde o se empata el juego.

### Tabla comparativa de costos asociados al proyecto

NOVIEMBRE 2020	ENERO 2021
Pagaría lo que el trabajador me solicitará, ya que el desarrollar algún proyecto no es fácil, se necesita toda una preparación, estudios, organización y por lo mismo considero que el pagarle a alguien para realizar dicho proyecto se le paga también por su tiempo, dedicación y conocimientos. Hablando de mi proyecto aproximadamente entre unos \$500 - \$1,000.	Sigo considerando que cuando se le paga a una persona por realizarle algún trabajo o proyecto se le paga más que nada por la dedicación, empeño, preparación y conocimientos que posee el programador. A pesar de ello, considero que mi programa es un tanto sencillo de realizar, ya que hay diversos videos en los que te muestra cómo hacerlo desde 0. Sin embargo, conociendo el trabajo que me costó hacerlo cobraría unos \$800

A pesar de que baje el precio de mi trabajo es debido a la gran competencia que tienen este tipo de juegos, a pesar de ello personalmente como estudiante que necesita hacer ejercicios para entender funcionamientos me parece solidario de mi parte el brindarles ese apoyo a los nuevos programadores para que poco a poco vayamos creciendo.

Comparación de diagramas de Gantt para la elaboración del proyecto (Octubre 2020 vs Enero 2021)

ACTIVIDAD	INICIO	FIN	DURA- CIÓN (se- manas)	TIEMPO (MESES)				
				Nov.	Dic.	Enero	Feb	Marzo
Investigación	29-11-2020	27-12-2020	4					
Planificación (limitar las características, etc.)	28-12-2020	17-01-2021	3					
Diseño	18-01-2021	31-01-2021	3					
Pruebas	01-02-2021	22-02-2021	3					
Exportación, delimitación del uso	23-02-2021	08-03-2021	3					

ACTIVIDAD	INICIO	FIN	DURA- CIÓN (se- manas)	TIEMPO (MESES)		
				Nov.	Dic.	Enero
<b>Investigación</b>	29-11-2020	14-12-2020	2			
<b>Planificación</b> (limitar las características, etc.)	28-12-2020	17-01-2021	1			
<b>Diseño</b>	18-01-2021	31-01-2021	2			
<b>Pruebas</b>	01-02-2021	22-02-2021	2			
<b>Exportación, delimitación del uso</b>	23-02-2021	29-01-2021	1			

Canal de YouTube donde se encuentra el video:

<https://www.youtube.com/channel/UCChgj|VF4ZFN-AR0a0Bh4NQ>

<https://youtu.be/GwgFtlKtHog>

Repositorio de GitHub del Proyecto Final:

<https://github.com/fp03alu12/Fundamentos-de-programacion>

## Conclusiones

El realizar este proyecto me ayudó bastante en muchos sentidos, principalmente me ayudó a reforzar los temas abordados en mi curso de Fundamentos de programación, además de que me amplió las ideas acerca de lo que es programar. Al inicio del curso yo realmente no sabía ni lo que era un hola mundo, pero ahora, me voy satisfecha de mi progreso con ganas de aprender cada vez más.

Personalmente como ya he mencionado, publicaría este programa debido a que existe la posibilidad de que programadores que apenas se van adentrando en lo que implica programar, debido a las estructuras fundamentales e importantes que se usan, recalcando el hecho de que con la pandemia miles de estudiantes se han visto en la necesidad de aprender basándose en diversos videos de internet, al menos en mi caso si fue así, cuando alguna estructura se me complicaba buscaba videos en lo que resolvieran ejemplos con dichas estructuras, provocando que me quedará más claro el tema. De igual forma, las plataformas tecnológicas cada vez nos dar la oportunidad de enriquecernos con nuevos conocimientos y a mi parecer el conocimiento se debe compartir, para que junto como sociedad vayamos evolucionando. Me gusta pensar que comparto ideas y valores con la IEEE, ya que dicho Instituto pone de prioridad la innovación y preparación de las distintas generaciones.

Hablando del lado de los videojuegos, es relevante recalcar que estos ayudan mucho a reducir el estrés, por lo mismo me pareció interesante el explorar esta cuestión, ay que nunca había pensando en dedicarme en programar videojuegos e inclusive pensaba que era una carrera relativamente sencilla, definitivamente con este “gato” me di cuenta de lo difícil que es el hacer cualquier tipo de videojuegos, también es importante recalcar que a pesar de que me basé mucho en un vídeo yo le fui agregando ciertas cuestiones que considere más estéticas. Realmente aún me falta mucho que aprender, pero al ser mi primer programa me siento orgulloso de él, me gustaría en un futuro hacer más eficiente el código y además hacer todo un menú con diversos juegos en él.

## Referencias

- ✓ Bloque 1.1 Juegos Simples en C: Tres en raya. Empieza a programar, 2014. Consultado: 03 diciembre en [https://www.youtube.com/watch?v=b5gOQ4\\_zgRc&t=375s](https://www.youtube.com/watch?v=b5gOQ4_zgRc&t=375s)
- ✓ Bloque 1.2 Juegos Simples en C: Tres en raya. Empieza a programar, 2014. Consultado: 13 diciembre en <https://www.youtube.com/watch?v=dakMghNMjwE&t=4s>
- ✓ 21. Programación en C++ || Ciclos o Bucles|| La sentencia for. Programación ATS (2016).
- ✓ 30. Programación en C++ || Arreglos || Concepto de Arreglo Unidimensional. Programación ATS (2016). <https://www.youtube.com/watch?v=l-kOjxvgyDQ>
- ✓ 34. Programación en C++ || Arreglos || Almacenando el contenido de 2 arreglos en un solo arreglo. Programación ATS (2016). <https://www.youtube.com/watch?v=6t9BRH8NCGQ>
- ✓ 63. Programación en C++ || Funciones || Definición de Funciones. Programación ATS (2016). <https://www.youtube.com/watch?v=vLZymLKd6Mo&t=210s>
- ✓ 82. Programación en C++ || Punteros || Declaración de Punteros. Programación ATS (2016). [https://www.youtube.com/watch?v=\\_pcfFMFs9-g](https://www.youtube.com/watch?v=_pcfFMFs9-g)
- ✓ Facultad de Ingeniería. (2018, 6 abril). Manual de prácticas de fundamentos de programación MAD0-17\_FP. Laboratorio de Computación Salas A y B. <http://lcp02.fi-b.unam.mx/>