



Universidad Nacional Autónoma de México

Facultad de Ingeniería

FUNDAMENTOS DE PROGRAMACIÓN

2022 – 1

Grupo: 04

**PROYECTO FINAL: JUEGO DEL
AHORCADO**

ESTUDIANTE

García Herrera Diego Vinicio

PROFESOR

Marco Antonio Martínez Quintana



Ciudad Universitaria, Cd. Mx., 10 / 12 / 2021

Contenido del proyecto:

<u>Introducción</u>	<u>03</u>
<u>Algoritmo</u>	<u>04</u>
<u>Diagrama de flujo</u>	<u>05</u>
<u>Pseudocódigo</u>	<u>06</u>
<u>Código fuente comentado</u>	<u>07</u>
<u>Resultados</u>	<u>08</u>
<u>Tabla de Recursos informáticos y de costos</u>	<u>09</u>
<u>Diagrama de Gantt y Manual de uso.....</u>	<u>10</u>
<u>Conclusiones y Referencias</u>	<u>11</u>

Introducción al proyecto realizado sobre el juego del ahorcado

Como se mencionó en clase, se nos presentó la oportunidad de presentar un juego como proyecto final de la materia de fundamentos de programación, el profesor mencionó entre los ejemplos, el juego del ahorcado, el cual siempre me ha parecido un buen juego.

Por lo que investigué en internet la forma en la que se podría desplegar dentro de la terminal, a mi parecer fue simple de comprender y sencillo de plantear.

Seguido de esto, comencé a buscar bases para la codificación del mismo, para tener en donde basar mi proyecto, y encontré diferentes códigos que se podrían complementar correctamente entre sí.

Dentro de este proyecto se planea emplear los conocimientos de codificación sobre lenguaje C impartidos en el curso de primer semestre de fundamentos de programación 2022–1, intentando incluir la mayor cantidad de sentencias y comandos vistos dentro del mismo curso, para estructurar, codificar y ejecutar de manera correcta un juego.

De igual manera se emplearán las herramientas que se nos proporcionaron dentro del curso, mismas como:

- La elaboración de un algoritmo, que nos permitirá conocer la estructura que deberá llevar nuestro proyecto.
- Un diagrama de flujo, donde vaciaremos los datos contenidos en el algoritmo, esto para una mayor organización y menor dificultad para la escritura del código de nuestro juego.
- Pseudocódigo, el cual nos dará una idea más clara del orden que deberemos seguir para codificar.

Los resultados que se esperan de este proyecto, es el correcto desarrollo del juego del ahorcado y completa funcionalidad desde inicio a fin, así como la presentación de las herramientas empleadas, tablas de costo y conclusiones sobre la experiencia acerca de la elaboración de este mismo.

Algoritmo del Juego

“JUEGO DEL AHORCADO EN C”

PROBLEMA: Incógnita de frase o palabra.

ENTRADA: Jugador 1, jugador 2, instrucciones, frase o palabra para adivinar, pista, número de intentos permitidos, interfaz de victoria o derrota.

SALIDA: Victoria / Derrota: Frase descifrada y sumatoria de puntos.

ALGORITMO:

1. Crear la interfaz de bienvenida: colocar instrucciones y solicitar frase o palabra que se desea adivinar.
2. Solicitar una pista (película, lugar o cosa) para apoyar un poco al otro jugador.
3. Establecer un máximo de fallos, en este caso lo haremos de 8
4. Crear la interfaz para el jugador dos: instrucciones y pedirle letra por letra la palabra que planteó el jugador anterior.
5. Mostrar la pista dada por el jugador anterior
6. Comenzar el juego escribiendo letra a letra
7. Si se llega al límite de intentos permitidos, mostrar la interfaz de derrota: la cual contendrá la sumatoria de todos los puntos obtenidos y la frase que no se logró descifrar.
8. Si se descifra la frase o palabra; mostrar la interfaz de victoria: la cual contendrá el mensaje de felicitación, la frase descifrada y la sumatoria de puntos.
9. Preguntar si se desea iniciar una nueva partida (SI o NO).
10. Fin del juego

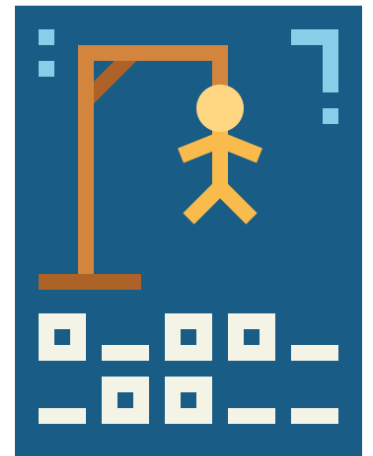
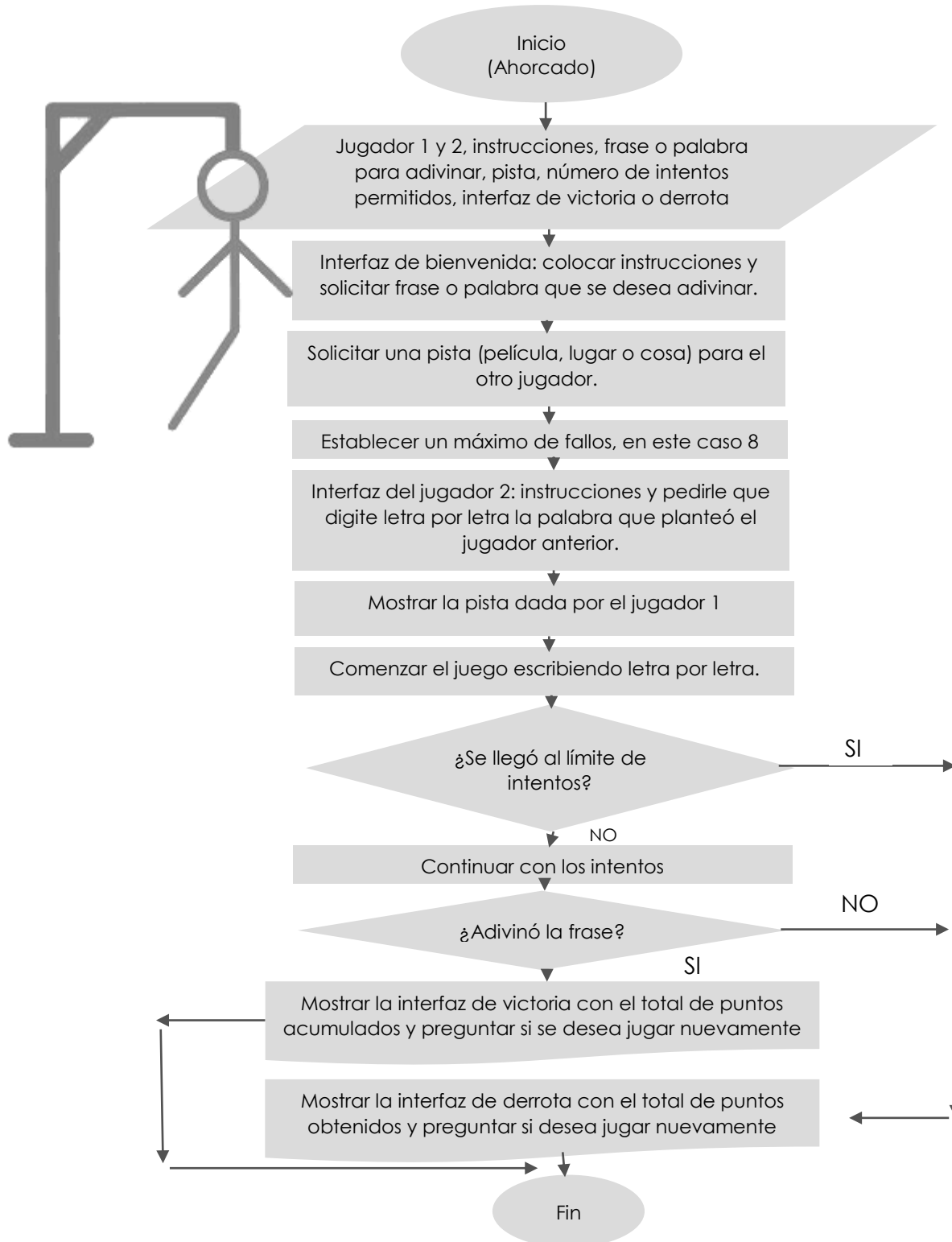


Diagrama de Flujo

Posterior a plantear el algoritmo en el cual nos vamos a basar para ordenar nuestra codificación, vaciaremos todos los procesos dentro de un diagrama de flujo para una mejor organización y comprensión:



Pseudocódigo

```
// JUEGO DEL AHORCADO EN C

// Definir Bienvenida al juego
"Proyecto Final: Juego del ahorcado en C"
"El jugador 1 está retando al jugador 2"

// ESTABLECEMOS LAS INSTRUCCIONES PARA EL JUGADOR 1
"Escriba una palabra o frase menor a 50 caracteres"
"Escriba una pista que ayudará al jugador 2
(peliculo, color o lugar)"

// INTERFAZ E INSTRUCCIONES PARA JUGADOR 2
"Empice el juego"
"Intente adivinar letra por letra antes de 8 fallos"
"La pista dada por jugador 1 fue: _____"

// EN CASO DE LLEGAR A 8 FALLOS Y NO ADIVINAR
"PERDISTE: intentalo de nuevo"
"La frase o palabra era: _____"
"Tu total de puntos fue: ____"
"¿Desea jugar de nuevo?"

//EN CASO DE ADIVINAR
"GANASTE, FELICIDADEES"
"Tu total de puntos fue: ____"
"¿Desea jugar de nuevo?"

//SALIR
```



Código Fuente (Comentado)

```

1 #include<stdio.h>
2 #include<string.h> // Además de la librería de entrada y salida estándar incluimos esta nuev
3
4 #define MAX 50 // MAXIMO PARA ADIVINAR
5 int main()
6 {
7     //DECLARACIÓN DE VARIABLES
8     char palabras[MAX+1]; //Palabra o frase para adivinar.
9     unsigned short i; //variable contadora para el ciclo for
10    unsigned short puntos=0, fallos=0,logros=0; //puntos acumulados, fallos y letras adivinadas
11    unsigned char intento; //letra que ingresa el jugador para adivinar
12    unsigned short bandera; //indicará si el jugador adivinó alguna letra o no
13    unsigned char pista[MAX+1];
14    unsigned char reinicio[3]='n','i','\0'; //respuesta para otra partida o salir del juego
15    unsigned short num_espacios=0; //variable acumuladora, numero de espacios de la frase ingresada
16
17    //DEFINIMOS LA BIENVENIDA A NUESTRO JUEGO
18    while (strcmp(reinicio,"si")==0){
19
20
21        printf(">>BIENVENIDO A MI PROYECTO FINAL: AHORCADO<<\n\n");
22        printf("jugador 1. reta al jugador 2");
23
24        //DAMOS LAS INSTRUCCIONES PARA EL JUGADOR 1
25        printf("\nEscriba una palabra o frase y pulse ENTER para comenzar. No debe sobrepasar las %d letras.\n",MAX);
26        gets(palabras); //se guarda la frase
27        unsigned char elegida[strlen(palabras)]; //creamos otra cadena del mismo tamaño que la cadena ingresada
28        strcpy(elegida,palabras); //copiamos la cadena en "elegida". Ahora cuenta con posiciones exactas
29
30
31        unsigned char oculta[strlen(elegida)]; //esta palabra será la elegida pero con guiones bajos para ocultarla y mostrar solo las que adivine el jugador
32        for(i=0;i<strlen(elegida);i++){ //ciclo for para llenar de guiones bajos hasta la ultima posición antes del caracter nulo
33            oculta[i]='_'; //oculta " _ _ _ _ _"
34
35
36        for(i=0;i<strlen(elegida);i++){ //ciclo para colocar espacios en caso de que la cadena "elegida" contenga más de una palabra
37            if(elegida[i]!=' '){
38                oculta[i]='_'; //colocamos la palabra a adivinar.
39                num_espacios++;
40            }
41
42
43        printf("\nDale una pista a tu contrincante (pelicula, lugar, color, etc.)\n");
44        gets(pista);
45

```

```

1 #include<stdio.h>
2 #include<string.h> // Además de la librería de entrada y salida estándar incluimos esta nuev
3
4 #define MAX 50 // MAXIMO PARA ADIVINAR
5 int main()
6 {
7     //DECLARACIÓN DE VARIABLES
8     char palabras[MAX+1]; //Palabra o frase para adivinar.
9     unsigned short i; //variable contadora para el ciclo for
10    unsigned short puntos=0, fallos=0,logros=0; //puntos acumulados, fallos y letras adivinadas
11    unsigned char intento; //letra que ingresa el jugador para adivinar
12    unsigned short bandera; //indicará si el jugador adivinó alguna letra o no
13    unsigned char pista[MAX+1];
14    unsigned char reinicio[3]='n','i','\0'; //respuesta para otra partida o salir del juego
15    unsigned short num_espacios=0; //variable acumuladora, numero de espacios de la frase ingresada
16
17    //DEFINIMOS LA BIENVENIDA A NUESTRO JUEGO
18    while (strcmp(reinicio,"si")==0){
19
20
21        printf(">>BIENVENIDO A MI PROYECTO FINAL: AHORCADO<<\n\n");
22        printf("jugador 1. reta al jugador 2");
23
24        //DAMOS LAS INSTRUCCIONES PARA EL JUGADOR 1
25        printf("\nEscriba una palabra o frase y pulse ENTER para comenzar. No debe sobrepasar las %d letras.\n",MAX);
26        gets(palabras); //se guarda la frase
27        unsigned char elegida[strlen(palabras)]; //creamos otra cadena del mismo tamaño que la cadena ingresada
28        strcpy(elegida,palabras); //copiamos la cadena en "elegida". Ahora cuenta con posiciones exactas
29
30
31        unsigned char oculta[strlen(elegida)]; //esta palabra será la elegida pero con guiones bajos para ocultarla y mostrar solo las que adivine el jugador
32        for(i=0;i<strlen(elegida);i++){ //ciclo for para llenar de guiones bajos hasta la ultima posición antes del caracter nulo
33            oculta[i]='_'; //oculta " _ _ _ _ _"
34
35
36        for(i=0;i<strlen(elegida);i++){ //ciclo para colocar espacios en caso de que la cadena "elegida" contenga más de una palabra
37            if(elegida[i]!=' '){
38                oculta[i]='_'; //colocamos la palabra a adivinar.
39                num_espacios++;
40            }
41
42
43        printf("\nDale una pista a tu contrincante (pelicula, lugar, color, etc.)\n");
44        gets(pista);
45

```

```

88    }
89
90    //RESULTADOS DE LA PARTIDA
91    if(fallos==0){ //Si se llegó al limite de fallos (5) mostrar: DERROTA
92        system("cls");
93        printf("\n\nNOOOO, PERDISTE. Sigue intentando en otra partida :)\n");
94        printf("\nRESULTADOS DE LA PARTIDA:\n");
95        printf("Puntaje final:      %d",puntos);
96        printf("\nLetras adivinadas: %s",oculta);
97
98    if(logros==strlen(elegida)){ //si tadvinó la palabra o frase, mostramos: "VICTORIA"
99        system("cls");
100        printf("\n\nRUUUUUUL, TERMINASTE EL JUEGO Y GANASTE \n");
101        printf("\nRESULTADOS:\n");
102        printf("Fallos:          %d",fallos);
103        printf("\nPuntaje final:    %d",puntos);
104        printf("\nFrase adivinada: %s",oculta);
105    }
106
107    printf("\n\n¿Quieres jugar otroooo? (SI O NO)\n");
108    scanf("%s",&reinicio);
109    fflush(stdin);
110    system("cls");
111    }
112
113    //INTERFAZ DE SALIDA DEL JUEGO
114    printf("\nENTER para salir");
115    fflush(stdin);
116    getchar();
117
118    return 0;
119
120

```

Resultados (En la terminal)

```
Símbolo del sistema - proyectoahorcado.exe

C:\Users\Cuchus\Desktop\Lenguaje C\PROYECTO>gcc proyectoahorcado.c -o proyectoahorcado.exe
proyectoahorcado.c: In function 'main':
proyectoahorcado.c:46:2: warning: implicit declaration of function 'system' [-Wimplicit-function-declaration]
  46 |     system("cls"); //limpiamos pantalla
      |     ^~~~~~

C:\Users\Cuchus\Desktop\Lenguaje C\PROYECTO>proyectoahorcado.exe
>>BIENVENIDO A MI PROYECTO FINAL: AHORCADO<<

jugador 1. reta al jugador 2
Escriba una palabra o frase y pulse ENTER para comenzar. No debe sobrepasar las 50 letras.
el rey leon

Dale una pista a tu contrincante (película, lugar, color, etc.)
película disney
```

```
Símbolo del sistema - proyectoahorcado.exe

Numero de fallos: 0
Puntos:          25000
letras adivinadas: el _e_ le_

Ingresa una letra: _
```

```
0000, PERDISTE. Sigue intentando en otra partida :)
RESULTADOS DE LA PARTIDA:
Puntaje final:      19000
letras adivinadas: _ _ _ _ a
```

En caso de perder

```
Símbolo del sistema - proyectoahorcado.exe

KUUUUUL, TERMINASTE EL JUEGO Y GANASTE
RESULTADOS:
Fallos:            0
Puntaje final:     52700
Frase adivinada:  el rey leon

Quieres jugar otroooo? (SI O NO)
_
```

En caso de adivinar

Tabla de Recursos Informáticos

Software	Hardware
Compilador de archivos (MinGW)	Ordenador
Símbolo del sistema	

Tabla de Costos Asociados

Proceso	Costo \$
DIAGRAMA DE GANTT PARA CALCULAR TIEMPOS ESTIMADOS	\$ 1000
DESCRIPCIÓN DEL PROYECTO, Y ELABORACIÓN DE ALGORITMO INICIAL.	\$ 1500
CREACIÓN DE DIAGRAMA DE FLUJO	\$ 1000
TRANSCRIPCIÓN DE UN PSEUDOCÓDIGO	\$ 1000
CODIFICACIÓN DEL PROYECTO	\$ 5500
ENTREGA DE RESULTADOS PROBADOS (TOTAL)	\$ 10,000.00

Diagrama de Gantt sobre la entrega

	22 Nov	26 Nov	03 Dic	10 Dic
Definición de proyecto a realizar				
Diagrama de Gantt				
Algoritmo y diagrama de flujo				
Entrega de Avances				
Entrega final del proyecto				

Manual de uso

1. Abrir la terminal en el ordenador
2. Dirigirse a la carpeta donde se encuentra el código fuente (cd desktop / cd lenguaje c / cd proyecto)
3. Compilar el archivo y crear el ejecutable (gcc proyectoAhorcado.c -o proyectoAhorcado.exe)
4. Ejecutar el programa (ProyectoAhorcado.exe)
5. El jugador 1 deberá ingresar la frase que se desea adivinar, al igual, deberá dar una pista sobre la palabra o frase dada anteriormente.
6. El jugador 2 intentará descifrar la palabra o frase ingresada por el jugador 1.
7. Para salir del juego escriba la palabra "NO"
8. Pulse enter.

Conclusiones y Referencias

La elaboración de este proyecto fue más complicada de lo que creía, ya que tal vez aún no domino al 100% los temas y las formas correctas de manipular las sentencias del lenguaje C, fue gracias a investigaciones en internet que pude guiarme y sacar adelante este proyecto...

De igual manera fue difícil asignarles un costo a los procesos realizados en este proyecto, ya que no tengo conocimientos básicos sobre estos, los asigné en base lo que se ha comentado en clases anteriores.

La implementación de herramientas como el diagrama de Gantt favorecieron la organización y la medición de mis tiempos para entregar en tiempo y forma este proyecto.

Por otra parte, como ya lo había mencionado, el diagrama de flujo y el algoritmo me ayudaron demasiado para identificar el orden en el que debería correr el código y hacerlo de la manera más ordenada posible.

En conclusión, considero que la realización de este proyecto me ayudará un poco a reforzar mis conocimientos del curso, y así gradualmente ir mejorando mis habilidades en programación, para de esta forma, en un futuro, no depender o apoyarme de otras fuentes, como en este caso fue internet.

Referencias Empleadas

- Juego “El ahorcado” en C. (s. f.). Blogspot. Recuperado 9 de diciembre de 2021, de <https://programavideojuegos.blogspot.com/2013/05/juego-el-ahorcado-en-c.html>
- C. Bautista. (2020, 30 junio). *El Ahorcado en Lenguaje C* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=yJWtwhLHj4&list=WL&index=8&t=310s>