

STAR WARS

CÓDIGOS SECRETOS

"EL VIDEOJUEGO"



NOMBRE: CARLOS PÉREZ RAJADELL

CURSO: 1º DESARROLLO DE APLICACIONES MULTIPLATAFORMA

ASIGNATURA: PROGRAMACIÓN

PROFESOR: JAUME ARAGÓ VALLS

INDICE

1. DESCRIPCIÓN GENERAL DEL PROYECTO	2
1.1 OBJETIVOS	2
1.2 TIPO DE JUEGO CONVERSACIONAL	2
1.3 CONTEXTO HISTORICO DE STAR WARS	3
1.4 PRINCIPALES ESTRUCTURAS UTILIZADAS	3
1.4.1 FUNCIONES	3
1.4.2 CASTING	3
1.4.3 VARIABLES	3
1.4.4 CONDICIÓN IF	4
1.4.5 CONDICIÓN ELSE	5
1.4.6 BUCLE FOR	5
2. FUNCIONAMIENTO (MANUAL DE USUARIO)	6
2.1 INTRODUCCIÓN	6
2.2 INSTRUCCIONES	6
2.3 PERSONAJES	7
2.4 MECANICA DEL JUEGO	7
3. EXPLICACIÓN DEL CÓDIGO	8
4. PROBLEMAS Y ERRORES	16
4.1 PROBLEMAS SOLUCIONADOS	16
4.2 PROBLEMAS NO SOLUCIONADOS	16
5. MEJORAS Y/O AMPLIACIONES	16
6. BIBLIOGRAFIA	16

1. DESCRIPCIÓN GENERAL DEL PROYECTO

1.1 OBJETIVO

El principal objetivo de la presente práctica es la desarrollar un videojuego educativo para jóvenes y adolescentes llamado “Star Wars Códigos Secretos”. La finalidad de este juego no es otra que la de fomentar la comprensión lectora y la competencia matemática a través de los míticos personajes de la saga Star Wars. Nos adentramos en el universo de Luke Skywalker y sus compañeros a través de una aventura conversacional en la que el usuario ha de superar 5 retos matemáticos, cada cual, con mayor dificultad, si quiere derrotar el mal y destruir la famosa Estrella de la muerte.

1.2 TIPOS DE JUEGOS CONVERSACIONALES

Existen muchos tipos de juegos conversacionales como:

- **Juegos de preguntas y respuestas:** implican hacer preguntas y respuestas, como su propio nombre indica, y hay que responder esas preguntas para seguir con el juego. Estas preguntas pueden ser triviales o de temáticas mas serias, todo depende del objetivo del juego.
- **Juegos de adivinanzas:** Estos juegos implican hacer adivinanzas y adivinar la respuesta correcta.
- **Juegos de roles:** Este tipo de juego conversacional implica adquirir diferentes roles y actuar en consecuencia con el rol adquirido. Son juegos que pueden funcionar muy bien para fomentar la empatía y la comprensión hacia los demás.
- **Juegos de historia compartida:** Estos juegos implican crear historias juntos, con cada persona agregando una parte de la historia. Estos juegos son muy útiles a la hora de fomentar la creatividad.
- **Juegos de palabras:** Este tipo de juego implica jugar con las palabras, como rimas o acertijos.
- **Juegos de debate:** Estos juegos llaman a discutir un tema en particular y tratar de persuadir a los demás para que estén de acuerdo contigo. Son muy útiles para fomentar y mejorar las habilidades de argumentación y persuasión.
- **Juegos de improvisación:** Estos juegos implican improvisar una escena o un diálogo en el momento.

En el caso de nuestro proyecto es un juego conversacional de preguntas y respuestas con una temática definida, como he comentado con anterioridad. Son problemas matemáticos

de menor a mayor dificultad, en el cual hay que ir superando los niveles de uno al cinco para terminar el juego.

1.3 CONTEXTO HISTORICO DE STAR WARS

La saga de Star Wars es una franquicia de películas de ciencia ficción creada por George Lucas en la década de 1970. La trama se desarrolla en una galaxia ficticia y sigue las aventuras de un grupo de personajes que luchan contra el Imperio Galáctico y otros enemigos. Desde su primera película, “Star Wars: Episodio IV - Una nueva esperanza”, hasta la última, “Star Wars: Episodio IX - El ascenso de Skywalker”, la saga ha sido una historia épica de aventuras, amor y lucha por la libertad. La franquicia ha tenido un gran impacto cultural y ha influido en la ciencia ficción y la cultura popular en general. Se siguen haciendo películas y series, a día de hoy, y cuenta con millones de seguidores en todo el mundo.

Por este motivo se decide realizar un juego con esta temática, para hacer mas atractivo un juego matemático y de ese modo aprender y resolver problemas junto con los personajes de nuestra saga favorita y con una historia en la cual nuestros resultados (aciertos) influyen, ya que si se acierta se pasa de nivel, si no el juego se termina.

1.4 PRINCIPALES ESTRUCTURAS UTILIZADAS

Para la realización de este proyecto se han utilizado las estructuras explicadas y tratadas en clase a lo largo de lo que llevamos de evaluación, aparte de basarse en fórmulas matemáticas y ejercicios, mucho de ellos resueltos en clase o muy parecidos.

1.4.1 FUNCIONES

Se han importado varias funciones o librerías para este proyecto como:

- **Función Scanner** (*java.util.Scanner*). Es una clase de java que se usa para leer la entrada del usuario. Esta clase proporciona varios métodos para leer diferentes tipos de datos como enteros (int), flotantes (float) o cadenas (String).
- **Función Math** (*java.lang.Math*). Esta clase o función se usa para realizar operaciones matemáticas básicas. Con esta clase se pueden realizar operaciones como exponenciación, logaritmos, raíces cuadradas o funciones trigonométricas.
- **Función Math.random**. Con esta función se generan números aleatorios. Con esta función se nos devuelve un número de tipo double. El numero generado es un valor

pseudoaleatorio, quiere decir que no es realmente aleatorio, sino que el número es generado mediante una fórmula matemática.

- **Función Math.floor.** Es una función de la clase Math que se utiliza para redondear un número decimal hacia abajo al número entero más cercano.

1.4.2 CASTING

Se han realizado varios castings a lo largo de este juego, ya que esto se utiliza para convertir un tipo de dato en otro. En este caso, lo he utilizado para convertir el resultado de Math.random() en un número entero, ya que la función Math.random devuelve un número decimal entre 0 y 1. Cuando se multiplica por (10 -1 +1) se consigue un número entero del 0 al 9 y al sumar luego 1 se obtiene un número decimal entre 1 y 10. En resumen, se convierte un número decimal en un número entero.

1.4.3 VARIABLES

1.4.3.1 VARIABLE TIPO ENTERO

En este proyecto se han utilizado variables de diferentes tipos. Las principales y más utilizadas han sido las variables tipo entero. Para declarar este tipo de variable se utiliza el acrónimo INT seguido del nombre de la variable y opcionalmente el valor inicial de ésta. Este tipo de variable tienen un tamaño de 32 bits y puede almacenar valores enteros desde -2,147,483,648 a 2,147,483,647.

Ejemplo:

```
int s1=0, s2=0,sumaPc=0,P1=0, P2=0,ProductorioPc=1,factorialPc=1,
redondeo=0,P=0,M=0,S=0,factorial1=1,factorial2=1;
```

1.4.3.2 VARIABLE TIPO BOOLEAN

La variable tipo Boolean es un tipo de variable que almacena un valor booleano, que puede ser verdadero o falso (true or false). Para la declaración de esta variable se utiliza la palabra boolean seguida del nombre de la variable y su valor inicial (true or false). Este tipo de variables se suelen utilizar con las estructuras de control "if" o "while".

Ejemplo:

```
boolean primo=true;
```

Para este juego la variable boolean ha sido inicializada en true.

1.4.3.3 VARIABLE TIPO DOUBLE

La variable tipo double es una variable que sirve para representar números reales de doble precisión, ya que puede almacenar números con decimales y un rango mas amplio que el tipo float, ya que este tipo de variable se almacena en 64 bits mientras que el tipo float solo en 32 bits. La declaración de esta variable es como las anteriores, se declara que es una variable tipo double seguido de su nombre y opcionalmente se puede inicializar en un valor.

Ejemplo:

```
double n=1;
```

1.4.3.4 VARIABLE STRING

Un string es una secuencia de caracteres que se utiliza para representar texto. Se pueden realizar operaciones con String, como la concatenación o comparaciones, pero estas operaciones crean nuevos objetos String en lugar de modificar el original. Esta variable se puede inicializar como texto o como una cadena vacía de caracteres.

Ejemplo:

```
String Txt2="Los problemas empiezan cuando deben realizar un salto  
hiperespacial hasta al \n" +  
"sistema " + s1 + " en el sector " + s2 + ",pero el sistema de  
navegación está estropeado y el \n" +  
"computador tiene problemas para calcular parte de las coordenadas de  
salto. Chewbacca,\n" +  
"piloto experto, se da cuenta que falta el cuarto número de la serie.
```

1.4.4 CONDICIÓN IF

He utilizado para la creación de este juego estructuras condicionales como "If" y "else". La condición If es una estructura de control que se utiliza para la toma de decisiones, es decir, si se cumple una condición se ejecuta un bloque del código, en el caso de que no se cumpla se salta ese bloque de código. En el caso de juego si la condición o condiciones if se cumplen se pasa al siguiente nivel del juego.

Ejemplo:

```
if (sumaPc==resultado) {  
  
    System.out.println("Nivel 2");
```

En este caso si se cumple la condición de que el resultado que realiza la CPU es igual al resultado que introduce el usuario, se pasa al siguiente nivel.

1.4.5 CONDICION ELSE

La condición else, o también llamada cláusula “else” es una estructura de control que se utiliza juntos con la estructura o condición “if”. Si la condición de la estructura “if” no se cumple se ejecutará el bloque de código que se encuentra dentro de la estructura else. Esta estructura se utiliza para tomar decisiones basadas en diferentes condiciones o situaciones.

Ejemplo:

```
}else{
    System.out.println("Ese no era el código correcto... La misión ha
    sido un fracaso... :( :(\n" +
        "Todavía no eres un Maestro Jedi de las Matemáticas. ¡Vuelve a
    intentarlo!");
```

En este caso la condición “else” indica que la respuesta introducida ha sido incorrecta y es el final del juego, ya que una vez fallada la respuesta se termina el juego y hay que volver a empezar.

1.4.6 BUCLE FOR

El bucle “for” es útil cuando se sabe de antemano el número interacciones que se van a realizar es decir, es una estructura de control que permite ejecutar un bloque de código repetidamente un número específico de veces. Primero se establece una variable de control y se le asigna un valor inicial y solo se realiza al inicio del bucle (una sola vez). La condición se evalúa antes de cada vuelta del bucle. Si la condición es verdadera, el código que hay dentro del bucle se ejecuta, en el caso contrario, si la condición es falsa el bucle se para. Además de esto, después de cada vuelta, se realiza una actualización de la variable de control, que puede ser tanto un incremento como un decremento en la variable.

Ejemplo:

```
for(int i=s1; i<=s2; i++){
    sumaPc += i;
}
```

En este ejemplo maestro el bucle for del primer nivel o juego de nuestro programa. En este caso se inicializa la variable i con el valor de S1, luego antes de cada vuelta se comprueba si la i es menor o igual que S2. Dentro del bucle se suma el valor de i a la variable sumaPc y después de cada vuelta el valor de i incrementa en 1 y este bucle finaliza cuando i sea mayor que S2.

2. FUNCIONAMIENTO DEL JUEGO (MANUAL DE USUARIO)

2.1 INTRODUCCIÓN

Bienvenido al emocionante juego “Star Wars: Códigos Secretos”. En este juego conversacional, te embarcarás en una épica aventura en la galaxia Star Wars, donde tu destreza con las matemáticas será la clave para derrotar la Estrella de la Muerte y de ese modo conseguir salvar la galaxia.

La estrella de la Muerte se encuentra en el limite de destruir planetas enteros, y tu y solo tú, un intrépido Jedi matemático podrás detenerla. Para avanzar por la galaxia y acercarte al objetivo, debes resolver una serie de desafiantes ejercicios matemáticos que pondrán a prueba tus habilidades numéricas. Cada respuesta correcta te acercará un paso más a la Estrella de la Muerte y te permitirá desactivar sus mortales sistemas.

¿Tienes lo que se necesita para derrotarla y de ese modo convertirte en el héroe que la galaxia necesita? Prepárate para sumergirte en un emocionante viaje a través del espacio y el tiempo, donde las matemáticas son tus aliadas y la Fuerza está de tu parte. ¡Que la Fuerza y los números te acompañen en esta épica misión matemática en el universo de Star Wars!

2.2 INSTRUCCIONES DE JUEGO

Para jugar a este juego solamente necesitas seguir estos pasos:

- Preparación:
 - Enciende el ordenador y abre el juego con IntelliJ
 - Consigue papel y un boli para hacer y anotar los cálculos.
- Introducción
 - Después de una breve introducción te embarcarás en una emocionante misión matemática
- Los retos:
 - En cada nivel, de los 5 que existen, te enfrentarás a un desafío matemático único.
 - Lee cuidadosamente la pregunta y realiza los cálculos necesarios para encontrar la solución.
- Introduce tu respuesta
 - Una vez que hayas encontrado la respuesta en papel introdúcela en tu ordenador utilizando el teclado.
- Verificación
 - El juego verificará si tu respuesta es correcta. Si lo es avanzarás al siguiente nivel. Si es incorrecta el juego terminará teniendo que volver a empezar de nuevo.
- Avance
 - Con cada nivel superado el desafío se vuelve mas emocionante y las matemáticas más intrigantes
- Victoria
 - Tu objetivo final el completar todos los niveles de juego y alcanzar así la victoria.

2.3 PERSONAJES DEL JUEGO

En la galaxia de “Códigos Secretos”, ha nacido un nuevo héroe, el Maestro Jedi de las matemáticas. Con una misión crucial, se encuentra acompañado de los míticos personajes de Star Wars: Luke Skywalker, el valiente líder, Chewbacca, el leal copiloto, la Princesa Leia la astuta diplomática, Han Solo, el audaz contrabandista, R2-D2, el fiel droide y C-3PO, el protocolo dorado.

Bajo la guía del Maestro Jedi de las matemáticas, estos personajes míticos contarán contigo para superar los desafíos matemáticos que se presentan en esta aventura. ¡Que la Fuerza matemática te acompañe!

2.4 MECÁNICA DEL JUEGO

Una vez que arranques tu IDE favorito, serás recibido por la introducción de este épico juego. La fuerza está de tu lado, y tu misión es crucial: convertirte en el Maestro Jedi de las matemáticas.

La introducción

La pantalla se ilumina con la misteriosa insignia de Star Wars, seguida de una breve introducción que te sumerge en esta emocionante travesía. Cuando estés listo solamente hay que presionar “Intro” para continuar.

Los retos

El sumatorio: En el primer nivel, se debe calcular un sumatorio de dos números para conseguir realizar con éxito el salto hiperespacial.

El productorio: En el segundo nivel se debe calcular el productorio de dos números para averiguar el código de acceso.

El factorial: El tercer reto es el cálculo de un factorial para conseguir la clave de acceso a los planos de la nave.

Números Primos: El cuarto reto, ya el penúltimo, es el averiguar si el número propuesto es primo o no y con ello conseguimos descryptar el código de acceso a la puerta del reactor.

Suma Final: El quinto y último reto es la suma de dos factoriales para de este modo con nuestra habilidad y astucia convertir los minutos y segundos en minutos y segundos del sistema Zorgdiano.

Superando los niveles

Con cada respuesta correcta se avanza al siguiente nivel. Recuerda, cada vez que juegues los valores serán diferentes, lo que garantiza que siempre debes calcular y usar las habilidades matemáticas. Si en algún momento te equivocas, el juego se terminará, y deberás comenzar de nuevo la batalla.

La batalla Final

Tu objetivo final es completar los cinco niveles del juego. Si lo logras, demostrarás ser el Maestro Jedi de las Matemáticas.

3. EXPLICACIÓN DEL CÓDIGO

```
import java.util.Scanner;
import java.lang.Math;
```

Primero se importan las librerías a utilizar.

```
public class JuegoStarWars {
    public static void main(String [] args){

        System.out.println("      88888888888      888      888888");
        System.out.println("      88      88      88 88      88 88");
        System.out.println("      8888      88 88      88 8888888");
        System.out.println("      88 88 8888888888 88      88");
        System.out.println(" 8888888888 88 88      88 88      8888888");

        System.out.println();

        System.out.println(" 88      88
88      888      88888      888888");
        System.out.println(" 88      88      88      88 88      88 88 88");
        System.out.println(" 88      8888      88 88      88 888888      88888");
        System.out.println("      888      888      8888888888 88      88      88");
        System.out.println("      88 88      88      88 88      88888888");
        System.out.println();
        System.out.println("=====STAR WARS: CÓDIGOS
SECRETOS=====");
        System.out.println();
        System.out.println("                                /~\\");
        System.out.println("                                |oo  )");
        System.out.println("                                \\=/  _");
        System.out.println("                                /  _  ");
        System.out.println("\\\\");
        System.out.println("      /      ()      \\      // | / \\
|\\");
        System.out.println("      _|_____|_      \\\\      \\_/_
||");
        System.out.println("      | |      ==  | |      \\ \\ |\\ / |
||");
        System.out.println("      |_|      0      |_|      #  _ _/_
#");
        System.out.println("      | |      *      | |      |  |
|");
        System.out.println("      | |_____| |      |  |
|");
        System.out.println("      |~      \\\\_/_      ~|      [ ] |
[]");
        System.out.println("      /=\\      /=\\      /=\\      |  |
|");
        System.out.println("      _____[__]_
[___]__[___]_____/___][___\\____");
        System.out.println();

        String Txt1="Hace mucho tiempo, en una galaxia muy, muy
lejana... La Princesa Leia, Luke\n" +
```

```

        "Skywalker, Han Solo, Chewbacca, C3PO y R2D2 viajan en
una nave imperial robada\n" +
        "en una misión secreta para infiltrarse en otra
estrella de la muerte que el imperio\n" +
        "está construyendo para destruirla. (Presiona Intro
para continuar)";

        System.out.println(Txt1);
        Scanner Teclado=new Scanner(System.in);
        String respuesta=Teclado.nextLine();

```

Después se abre la clase public y main. Luego sale la presentación del juego con las letras y el dibujo de star wars. Sale una breve introducción de la historia y se espera que el jugador pulse intro para comenzar a jugar. Con esta ultima acción se utiliza un objeto Scanner para tomar la entrada del jugador.

```

if(respuesta.isBlank()){
    // Para Iniciar el juego he utilizado este método de clase String.
    De esta manera se comprueba si la cadena respuesta está vacía o
    contiene espacios en blanco. Si está en blanco el resultado es True y
    se ejecuta el bloque If. Si es False se omite el bloque If y pasa al
    bloque else.

    System.out.println("Nivel 1");

    int s1=0, s2=0,sumaPc=0,P1=0, P2=0,ProductorioPc=1,factorialPc=1,
    redondeo=0,P=0,M=0,S=0,factorial1=1,factorial2=1;
    double n=1;
    int resultado=0; // En este lugar inicializo todas las variables
    del juego.

    s1=(int) (1+Math.random()*(10 -1 +1));
    s2=(int) (20+Math.random()*(30 -20 +1));
    P1=(int) (1+Math.random()*(7 -1 +1));
    P2=(int) (8+Math.random()*(12 -8 +1));
    n=50+Math.random()*(100-50+1);
    redondeo=(int) Math.floor(n/10);
    P=(int) (10+Math.random()*(100 -10 +1));
    M=(int) (5+Math.random()*(10 -5 +1));
    S=(int) (5+Math.random()*(10 -5 +1));
    //Aquí he realizado todos los cálculos de los números aleatorios
    del juego.

    String Txt2="Los problemas empiezan cuando deben realizar un salto
hiperespacial hasta al \n" +
    "sistema " + s1 + " en el sector " + s2 + ",pero el sistema de
navegación está estropeado y el \n" +
    "computador tiene problemas para calcular parte de las coordenadas
de salto. Chewbacca,\n" +
    "piloto experto, se da cuenta que falta el cuarto número de la
serie.\n" +
    "Recuerda de sus tiempos en la academia de pilotos que para
calcularlo hay que\n" +
    "calcular el sumatorio entre el nº del sistema y el nº del sector
(ambos inclusive).\n" +
    "¿Qué debe introducir?";

    System.out.println(Txt2);
    //En el print del String primero he creado un String y lo he

```

```
concatenado con las variables del primer juego, de esta manera los
números aleatorios forman parte del enunciado del juego.
//Aunque podría haber introducido directamente el texto dentro del
print y concatenar a la vez, he preferido realizarlo en dos pasos
diferentes. Y luego vincular el String txt con el print.

// Se realiza el primer juego con un bucle for. Se inicializa la
variable i con el valor de s1, luego antes de cada vuelta se comprueba
si la i es menor o igual que s2.
//Dentro del bucle se suma el valor de i a la variable sumaPc y
después de cada vuelta el valor de i incrementa en 1 y este bucle
finaliza cuando i sea mayor que s2.
for(int i=s1; i<=s2; i++){

    sumaPc += i;
}
System.out.println(sumaPc);
System.out.println("Introduce el resultado");
resultado=Teclado.nextInt();

//En este condicional si el resultado introducido por el usuario y
el resultado calculado por la máquina es el mismo, entonces se pasa al
siguiente nivel. En caso contrario saldrá del juego.
if (sumaPc==resultado){
```

Luego se declaran las variables del juego, se inicializan y se realiza la formula `math.random` para todos los números aleatorios que han de aparecer en el juego. Acto seguido comienza el nivel 1, donde se debe de calcular el sumatorio utilizando un bucle `for`. El resultado del sumatorio se muestra y el jugador debe ingresar su respuesta. Si la respuesta es correcta, es decir si la respuesta del jugador es igual a la respuesta del juego se pasa al nivel 2, si no se sale del juego y hay que volver a empezar.

```
System.out.println("Nivel 2");
System.out.println();//He dejado print en blanco para dejar espacio
entre el titulo de Nivel y el enunciado del juego.

String Txt3= "Gracias a Chewbacca consiguen llegar al sistema correcto
y ven a lo lejos la estrella\n" +
"de la muerte. Como van en una nave imperial robada se aproximan
lentamente con \n" +
"la intención de pasar desapercibidos. De repente suena el
comunicador. "Aquí \n" +
"agente de espaciopuerto " + P1 + " contactando con nave imperial " +
P2 + ". No están destinados \n" +
"en este sector. ¿Qué hacen aquí?". Han Solo coge el comunicador e
improvisa. "Eh... \n" +
"tenemos un fallo en el... eh... condensador de flujo... Solicitamos
permiso para \n" +
"atrascar y reparar la nave". El agente, que no se anda con tonterías,
responde \n" +
"- Proporcione código de acceso o abriremos fuego -. Han Solo ojea
rápidamente el \n" +
"manual del piloto que estaba en la guantera y da con la página
correcta. El código \n" +
"es el productorio entre el nº del agente y el nº de la nave (ambos
inclusive).\n" +
"¿Cuál es el código?";

System.out.println(Txt3);
```

```

//En el segundo nivel también he utilizado el bucle for. El
funcionamiento de este bucle en este caso es el siguiente: Se inicia
la variable i con el valor de P1, que es la condición inicial del
bucle.
//Antes de cada vuelta se comprueba si la variable i es menor o igual
que la variable P2. Dentro del bucle se multiplica el valor de la
variable i por la variable productorioPc
//Y eso pasa por cada vuelta que hace el bucle. Por cada vuelta que
da, se incrementa el valor de i en 1. Se vuelve a la condición y se
repite hasta que i sea mayor que P2.
//Al final del bucle la variable productorioPc nos dará el resultado
del productorio entre la variable P1 y la variable P2.
for(int i=P1; i<=P2; i++){

    ProductorioPc*=i;
}
System.out.println(ProductorioPc);
System.out.println("Introduce tu respuesta");
resultado=Teclado.nextInt();

//Con este condicional pretendo igualar los resultados. Si
ProductorioPc es igual al resultado introducido por el usuario
entonces se pasa al siguiente nivel. De lo contrario nos sacará del
juego.
if (ProductorioPc==resultado){

    System.out.println("Nivel 3");
    System.out.println();

    String Txt4="Han Solo proporciona el código correcto. Atracan en
la estrella de la muerte, se\n" +
        "equipan con trajes de soldados imperiales que encuentran
en la nave para pasar\n" +
        "desapercibidos y bajan. Ahora deben averiguar en qué
nivel de los " + redondeo + " existentes se\n" +
        "encuentra el reactor principal. Se dirigen al primer
panel computerizado que\n" +
        "encuentran y la Princesa Leia intenta acceder a los
planos de la nave pero necesita\n" +
        "introducir una clave de acceso. Entonces recuerda la
información que le proporcionó\n" +
        "Lando Calrissian "La clave de acceso a los planos de la
nave es el factorial de N/10\n" +
        "(redondeando N hacia abajo), donde N es el nº de
niveles".\n" +
        "¿Cual es el nivel correcto?";

    System.out.println(Txt4);

    //Para el nivel 3 también he utilizado el bucle for. Para este
ejercicio he necesitado utilizar la función Math.floor (Es la
encargada de redondear, un número dado, hacia abajo).
    //El principio del bucle se inicializa con la variable i con el
valor de 1. Su condición es que hay que comprobar que la i es menor o
igual al numero obtenido como redondeo.
    // ""El número redondeo se consigue primero con la función
Math.random para conseguir el numero aleatorio y ese valor luego se
introduce en la función Math.floor, que es la variable redondeo.
    //Una vez dentro del bucle, se multiplica el valor de i por la
variable factorialPc y se repite cada vuelta del bucle. Por cada

```

```

vuelta del bucle incrementa el valor de i en 1 y se repite el
//proceso hasta que i sea mayor que redondeo.
for(int i=1;i<=redondeo;i++){

    factorialPc*=i;
}
System.out.println(factorialPc);
System.out.println("Introduce tu respuesta");
resultado=Teclado.nextInt();

//Con este condicional vuelvo a igualar resultados, es decir si el
resultado calculado por la máquina es el mismo resultado que el
introducido por el usuario se pasa al siguiente nivel. En caso
contrario se sale del juego con el bloque else.
if (factorialPc==resultado){

    System.out.println("Nivel 4");
    System.out.println();

    String Txt5 ="Gracias a la inteligencia de Leia llegan al
nivel correcto y encuentran la puerta\n" +
                "acorazada que da al reactor principal. R2D2 se
conecta al panel de acceso para\n" +
                "intentar hackear el sistema y abrir la puerta. Para
desencriptar la clave necesita\n" +
                "verificar si el número " + P + " es primo o no. Si es
primo introduce un 1, si no lo es\n" +
                "introduce un 0.";

    System.out.println(Txt5);
    System.out.println();
    System.out.println("El numero es: " + P);

    boolean primo=true;
    //He creado una variable booleana que es la que me permitirá
dentro del bucle saber si la variable primo es verdadera o falsa.

    //Comienza el bucle for inicializando la variable i con el
valor 2. Luego la condición es comprobar que la variable i es menor
que la variable P.
    //Luego se verifica mediante la condición if de que P es
divisible. Si es divisible asigna false a la variable primo. Por cada
vuelta que el bucle
    // se incrementa el valor de i en 1. Vuelve a la condición y
se repite el bucle hasta que i sea igual a P. Por ultimo si la
variable primo es verdadera
    // es porque P es un número primo, de lo contrario no es
primo.
    for(int i=2;i<=P;i++){

        if(P%i==0){
            primo= false;
        }
    }
    //Fin del For

    System.out.println("Es primo: " +(primo ? "1" : "0"));
    //En este print se imprime el número P, si el numero es primo aparecerá un
1, si no lo es aparecerá un 0.
    System.out.println("Introduce tu respuesta.");
    resultado=Teclado.nextInt();

```

```

        int respuestaAcertada = primo ? 1 : 0; //Aquí estoy asignando un
valor a la variable respuestaAcertada utilizando un operador ternario.
        //Si primo es verdadero la variable respuestaAcertada se
establece en 1 y si es falsa se establece en 0.

        //En esta condición se igualan resultados de nuevo. Si el
resultado introducido por el usuario es el mismo que respuestaAcertada
se pasa al siguiente nivel. Si no se sale del juego.
        if (resultado == respuestaAcertada) {

            System.out.println("Nivel 5");
            System.out.println();

            String Txt6 = "Consiguen entrar al reactor. Ya solo queda
que Luke Skywalker coloque la bomba,\n" +
                " programe el temporizador y salir de allí
corriendo. Necesita programarlo para que\n" +
                " explote en exactamente " + M + " minutos y " + S
+ " segundos, el tiempo suficiente para escapar\n" +
                " antes de que explote pero sin que el sistema de
seguridad anti-explosivos detecte y\n" +
                " desactive la bomba. Pero el temporizador utiliza
un reloj Zordgiano un tanto\n" +
                " peculiar. Para convertir los minutos y segundos
al sistema Zordgiano hay que sumar\n" +
                " el factorial de M y el factorial de S. ¿Qué valor
debe introducir?";

            System.out.println(Txt6);
            System.out.println();

            //Para el ultimo nivel he utilizado tambien el bucle
for. Se inicializa el bucle con la variable i dandole el valor 1. Antes
de cada vuelta
            //comprueba que la i es menor o igual a la variable M.
Luego se multiplica el valor de i por la variable factorial1 y se
repite en cada vuelta del bucle
            //Por cada vuelta que da el bucle se incrementa el valor
de i en 1. Luego se vuelve a la condición y se repite el bucle hasta
que i sea igual a la variable M.
            for (int i = 1; i <= M; i++) {

                factorial1 *= i;
            }

            System.out.println("El factorial de " + M + " es: " +
factorial1);

            //Se inicializa el bucle con la variable i dandole el
valor 1. Antes de cada vuelta
            //comprueba que la i es menor o igual a la variable S.
Luego se multiplica el valor de i por la variable factorial1 y se
repite en cada vuelta del bucle
            //Por cada vuelta que da el bucle se incrementa el valor
de i en 1. Luego se vuelve a la condición y se repite el bucle hasta
que i sea igual a la variable S.
            for (int i = 1; i <= S; i++) {

                factorial2 *= i;
            }
            System.out.println("El factorial de " + S + " es: " +

```


El ultimo "If" es la ultima verificación de respuesta/nivel, es decir si la respuesta es acertada entonces sale el texto final diciendo que el jugador ha ganado y sale el dibujo de Twenty Century Fox.

Aunque parece que esta explicación es demasiado breve, dentro de la propia programación de IntelliJ está detallada cada condición y cada bucle. Pero en las capturas de pantalla de este punto se pueden observar las explicaciones dentro de la programación.

4. PROBLEMAS Y ERRORES

4.1 PROBLEMAS SOLUCIONADOS

A lo largo de la programación de este juego se han planteado varias dudas y problemas. La primera fue el tener que averiguar como realizar la primera función del juego, como pulsar intro y que pasara al nivel 1. Una vez averiguado esto mediante ayuda de algún compañero se procede a realizar la programación de cada juego.

Otro de los problemas fue una duda de como resolver el nivel 4. No había caído en utilizar una variable booleana y después de comentar este problema con compañeros me comentaron que mediante esa variable se podría resolver el problema. Después de intentarlo un par de veces conseguí dar con la solución al problema y seguí con la programación hasta terminar.

4.2 PROBLEMAS NO SOLUCIONADOS

He intentado añadir música al proyecto, que, aunque no es requisito indispensable y ni siquiera se pide para este proyecto, me parece interesante poner en situación y contexto al jugador. Después de mirar tutoriales en Youtube y en otras webs de como hacerlo siempre me salía el mismo error y es como que Java no reconocía las librerías "media" que le importaba. Al final he dejado el proyecto sin música, aunque seguiré intentando averiguar como hacerlo, me parece algo muy interesante.

5. MEJORAS Y/O AMPLIACIONES

Las mejoras que podría realizar en mi programación en principio creo que la más importante es optimizar más el código y que sea más legible, ya que creo que se puede realizar sin tantas líneas, pero para eso también creo que hay que tener mas conocimientos.

Como mejoras he añadido el título de inicio "STAR WARS" y los dibujos de sus droides R2-D2 y C-3PO y al final del juego como ilustración final el logo de TWENTY CENTURY FOX.

6. BIBLIOGRAFIA

He visitado ciertas webs para poder describir con mejor precisión algunos datos que muestro a lo largo de este proyecto.

- [Java Tutorial \(w3schools.com\)](https://www.w3schools.com/)
- [Java Tutorial | Learn Java Programming \(geeksforgeeks.org\)](https://www.geeksforgeeks.org/)

