UNIVERSIDAD CATÓLICA DE COSTA RICA

INGENIERÍA EN SISTEMAS

PROGRAMACIÓN I

PRIMER PROYECTO, JUEGO AHORCADO

ANDRÉS JIMÉNEZ LEANDRO

JOSUE MUÑOZ SOLIS

SEDE SAN CARLOS, JUNIO, 2016

Contenido

[RESUMEN EJECUTIVO 4](#_Toc454750654)

[DESCRIPCION 5](#_Toc454750655)

[ABSTRACT 6](#_Toc454750656)

[OBJETIVOS GENERALES Y ESPESIFICOS 7](#_Toc454750657)

[INTRODUCCION 8](#_Toc454750658)

[DESARROLLO 9](#_Toc454750659)

[ANALISIS DE LA SOLUCION 10](#_Toc454750660)

[RESULTADOS OBTENIDOS 11](#_Toc454750661)

[CONCLUCION 11](#_Toc454750662)

[RECOMENDACIONES 12](#_Toc454750663)

[CRONOGRAMA 13](#_Toc454750664)

[BIBLIOGRAFIA 14](#_Toc454750665)

[BITAGORA 15](#_Toc454750666)

[ANEXOS 16](#_Toc454750667)

# RESUMEN EJECUTIVO

Juego Ahorcado

Juego programado con base al lenguaje de programación java, cuyo leguaje tiene como definición., Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como WORA, o "write once, run anywhere"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra.

El juego ahorcado será creado en un entorno de desarrollo integrado (IDE) de Netbeans, este programa es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

El ahorcado es un juego de baja complejidad porque este será desplegado en la consola de Netbeans, con una opción de mostrar algunas ventanas con mensajes utilizando el JOptionPane, para esto tendríamos que importar su librería. Siguiendo las instrucciones del enunciado se toma en cuenta algunas palabras claves que nos permiten formar el juego ahorcado de una manera más clara. No necesariamente por que el juego es de baja complejidad quiere decir que sea fácil programar dicho juego, pero si se realizara un ahorcado con paneles se nos complicaría un poco más., este pasaría a ser de nivel medio avanzado.

La composición del juego ahorcado desplegado en la consola del IDE de Netbeans es básico y se conforma de un grupo de palabras en un string, las cuales serán escogidas de manera aleatoria por un método llamado random, de manera siguiente será mostrado mensajes en la consola e ingresaremos letra por letra hasta que acertemos la palabra sin dejar que nuestro contador de vidas llegue a 0., para al final mostrarnos un mensaje de ganaste o perdiste.

En un resumen un poco corto es lo más sobresaliente de la complejidad de este juego ahorcado y su desarrollo en el leguaje java.

DESCRIPCION

Con base a lo solicitado por el profesor de programación, realizaremos un juego de tipo ahorcado utilizando conocimientos básicos aprendidos en clase y la otra gran parte aprendidos en casa, por medio de la investigación de la aplicación de algunos métodos que son utilizados para la programación orientada a objetos, de la cual recién estamos conociendo mediante prácticas guiadas y que para este proyecto son de gran ayuda.

En general la realización de este juego se resume en el siguiente enunciado

“Dicho juego, deberá desplegarse en la consola del NetBeans al ser corrido. En vez de la típica imagen del personaje ahorcándose, que se va armando línea a línea con cada error; este programa contará con un contador de intentos, con un máximo de 11 intentos por juego, que se irá descontando con cada error. Si el contador llega a cero, el programa deberá desplegar un mensaje indicando que se ha fallado y que el juego ha terminado y el jugador ha perdido la partida. Caso contrario, si la palabra es adivinada, se deberá desplegar un mensaje indicando que ha ganado el juego. El juego debe escoger una palabra de forma aleatoria (parte investigativa del proyecto), de un listado de palabras creado. El juego debe validar cada carácter que el jugador digite y si está errado, reducir un intento; si está correcto, deberá desplegar los espacios vacíos, con el carácter correcto digitado en el espacio correspondiente.”

Siguiendo al paso de letra cada una de las intrusiones realizaremos un juego desplegado en la consola el cual será limpio, corto, ejecutable.

# ABSTRACT

Hangman game

Programming game based on the java programming language, whose language has as definition, Java is a programming language of purpose general, concurrent, object that was specifically designed to have as few implementation dependencies as possible. Its intention is to allow application developers to write the program once and run it on any device (known in English as WORA, or once, run anywhere "write" "), which means that the code that is executed on a platform does not have to be recompiled to run on another.

The Hangman game will be created in Netbeans (IDE) integrated development environment, this program is an integrated development environment free, made mainly for the Java programming language. There is also a significant number of modules to extend it. NetBeans IDE is a free product without restrictions of use.

But what is the programming oriented to objects (OOP)?

It is a special way of programming, closer to how we could things in reality than other types of programming.

With the POO, we must learn to think about things in a different way, to write our programs in terms of objects, properties, methods, and other things that can be found quickly to clarify concepts and give a small base that allows you to drop us a bit with this type of programming.

Hangman is a game of low complexity because this will be displayed in Netbeans console, with an option to show some windows with messages using the JOptionPane, for this we would have to import your library. Following the instructions of the wording taking into account some key words that allow us to form the game hanged in a clearer way. Not necessarily because the game is low-complexity you want to say that it is easy to program the game, but if it is a Hangman with panels complicate us a little more., this would be between advanced level.

The composition of the Hangman game displayed on the console of the Netbeans IDE is basic and is made up of a group of words in a string, which will be selected randomly by a method called random, as follows will be displayed messages in the console and we enter letter by letter until we hit the correct word without leaving our lives counter reaches 0. to finally show us a message of you won or lost.

In a somewhat short summary is the most outstanding of the complexity of this Hangman game and its development in the language java.

# OBJETIVOS GENERALES Y ESPESIFICOS

Objetivo general

El objetivo general de este proyecto es la aplicación de los conocimientos obtenidos en la programación orientada a objetos, operaciones aritméticas, tanto salida como entra de datos y cosas importantes como la utilización de métodos., todo esto en la creación de un juego simple de intentos, un Ahorcado.

Objetivos específicos

Comprobar con un código en el IDE de Netbeans, que se puede aplicar los métodos investigados, los arreglos, la forma correcta de la utilización de las clases.

Hacer que el juego se desplegué en el ide sin ningún problema, también apegarse a las convenciones java y comentariado.

# INTRODUCCION

Este proyecto se realiza en un entorno de programación ideal para poner en función ciertos aspectos los cuales hemos investigado durante unos pocos días, de estos aspectos siendo los más importantes son los arreglos los cuales nos permitirán realizar una gran parte de este proyecto.

Un juego de baja complejidad como un ahorcado, nos plantea un bonito reto para mostrar lo investiga y aprendido, no obstante, este proyecto el cual es de suma importancia para el aprendizaje y el curso, nos invita a pensar y desarrollar mucho más la lógica y la aplicación de una que otra solución a conforme avanzamos en él.

Es muy importante tomar en cuenta lo desarrollado en este proyecto programado, para los próximos proyectos que si serán de una dificultad más alta., los utilizado en mayoría por este juego ahorcado serán simples métodos en base a lo sencillo del juego, el cual tendrá como máximo un numero de intentos y un grupo de palabras a adivinar. Con todo esto más otro conjunto de conocimientos será un poco sencillo crear este juego en el IDE de Netbeans.

# DESARROLLO

¿Qué es lo que hay que resolver?

En este proyecto demos tomar en cuenta muchos detalles que pueden resultar muy tediosos a la hora de comenzar a programar. Una de las cosas es el uso correcto de los métodos, hacer que funcionen de manera que nos ejecuten los que deseamos, ingresar datos por teclado con el scanner es una de los puntos importantes ya que si el no lograremos jugar nuestro Ahorcado.

El incluir una lista de palabras para ser adivinadas, pero para que esto sea posible, no solo las incluimos en un arreglo string que las almacenara, sino que también necesitamos que sean escogidas alzar y es ahí donde entra en juego el método Random con el Split que separa letras y el Length que mide el largo de ellas., en resumen lo principal es que el juego corra correctamente te pida una letra y la compare con las letras de la palabra escogida alzar y te presente si has acertado o no, al final cuente los intentos y determinar si se perdió o si se ganó.

¿Cómo se va resolver el problema?

Se debe crear una clase main en la que comentariemos todo el código que se va a escribir, comenzando con los importes que haremos como el scanner y el random.

De manera seguida crearemos el arreglo string que almacenara las palabras del juego, luego mostraremos un mensaje por la consola de Netbeans para que ingrese una letra, esta será por medio del teclado gracias al método scanner, por consiguiente la palabra será seleccionada alzar, en una instacia juego ahorcado, con el método random y se aplicara espacio con el Split y también lo largo de la palabra con el length para que se dé una pista de cuantas letras contiene la palabra que vamos a adivinar., lo siguiente es crear el contador en un int para que nos registre de cuántas vidas es el juego y como se van consumiendo las oportunidades, a continuación entramas en el for para ir determinando cada letra ingresada por el usuario con esto anteriormente utilizamos el equals que compara letras en un (= =) y de manera seguida en las condiciones de if, else para ir agregando letra acertados o ir quitando vidas en el contador dependiendo de la palabra que se esté o no se esté formando, cada uno de estas condiciones son cerradas con un break; para ir determinándolas, por otro lado se van desplegando mensajes con System.out.println para ir dando información al usuario sobre cada resultado. Luego colocamos un public boolean para la comparación de cada letra y palabras y un return al final para que nos devuelva el dato si se ganó.

Análisis del problema

Se tiene que conformar el código de manera racional y congruente con los que se pide, es importante que se realice de manera ordenada cada condición del código para que las letras ingresadas sean bien comparadas., temas importantes como el scanner, el random, el contador y los for son el corazón de dicho juego y que los hacen funcionar correctamente.

ANALISIS DE LA SOLUCION

Gran parte de lo pedido se logró implementar a pesar que era de mucha investigación, logre crear el juego con mucho apoyo de tutoriales y ejemplos en internet, ya que era un proyecto con una gran parte investigativa como lo asigno el profesor de programación I.

Se logró crear una categoría de palabras las cuales es de unos equipos de futbol de primera división de Costa Rica.

Se creó un método (random) que lograra escoger alzar palabras y las presentara en forma de signo para estar en incógnita (\*), también que fuera separadas con el (Split) y que se mostrara su largo con el (length).

Dentro de ese mismo public juego ahorcado se creo las condiciones for… if, else. Comprendiendo las exepciones de cada letra ingresada.

Se creó un public boolean para determinar si se acertó con un return para devolver el dato.

No logre implementar el menú con las categorías aunque al final investigue un poco sobre los (case) casos los cuales nos permite crear un menú., en ese mismo menú también se escoge categorías y jugar.

No logre desplegar mensajes con el JOptionPane, no porque no lo supiera hacer sino porque considere mejor mantener el juego únicamente en la consola y no desplegarlo en ventas, al final borre el importe de Joption por estar sin utilizar.

Como detalles importantes, no se logró depurar algunos errores como el ingreso de otros caracteres que no fueran letras, a lo cual el juego reacciona de manera que resta una vida., a parte de un problema que se presenta si se juega el ahorcado con el bloc mayus activado, ya que las palabras ingresadas están en minúsculas, esa mezcla entre mayúsculas y minúscula no pude depurarlo.

# RESULTADOS OBTENIDOS

Se pudo crear un código de manera simple, pero si con mucho trabajo de comprensión, se logró que en este juego se desplegara completamente en la consola de Netbeans y que se comparara cada letra que ingresáramos, analizándoles y contando cada error y cada acierto en el juego del ahorcado, la impresión de cada dato y al final el resultado óptimo de poder jugar sin que el programa se detenga.

A pesar de que no se tenía mucho conocimiento en algunos métodos necesarios para la creación del juego ahorcado, se suplemento con bastante investigación y con las misma practica de prueba y error en la utilización de cada una de los métodos.

# CONCLUCION

La realización de este proyecto me deja con un mayor conocimiento en algunos aspectos de la programación orientada a objetos de java, encontré diferentes maneras de crear el código con distintos métodos clases y entre más cosas que podemos implementar para generar un juego aún más complejo y creativo., no obstante, me quedaron algunas dudas de otras maneras de crear un ahorcado desde java con paneles y otras maneras muy bonitas pero un poco confusas.

El amplio conocimiento del lenguaje y convenciones java te pueden ayudar a crear más rápido un código, pero todo esto se obtiene con practica e investigación., también se necesita una guía necesaria para saber aplicar cada cosa que queramos.

El juego ahorcado que realizamos es de baja complejidad y nos ayudó a avanzar mucho más en el curso con conocimiento más en métodos, clases, entre otros.

# RECOMENDACIONES

Como recomendaciones puedo instar a comenzar a investigar bastante sobre el lenguaje de programación java y su programación orientada a objetos.

También comenzar a investigar sobre las convenciones java y sus totalidades para lograr tener más dominio del lenguaje.

Practicar es esencial y solo la practica hace al maestro, esto es lo que nos da la velocidad de programar y la familiaridad en cada método y sus aplicaciones.

# CRONOGRAMA

|  |  |  |
| --- | --- | --- |
| CRONOGRAMA | | |
|
|
| EN QUE TRABAJARE | FECHA | DIA |
|
| COMENZARE UNA INVESTIGACION SOBRE LOS CODIGOS DEL JUEGO | 07/06/2016 | LUNES |
| ESTUDIARE Y BUSCARE UN POCO DE INFORMACION DE LOS METODOS RANDOM, SCANNER, FOR | 08/06/2016 | MIERCOLES |
| SE INICIARÁ LA ELABORACION DEL CODIGO | 10/06/2016 | VIERNES |
| CREARE EL PROYECTO EN GITHUB Y HARE EL PRIMER COMMIT | 12/06/2016 | DOMINGO |
| SE ENTREGARÁ EL PRIMER AVANZE AL PROFESOR | 15/06/2016 | MIERCOLES |
| TRATARE DE ADELANTAR UN POCO EL CODIGO | 17/06/2016 | VIERNES |
| TRABAJAR BASTANTE EL CODIGO Y LO PARTE ESCRITA | 20/06/2016 | LUNES |
| DEPURARE UN POCO EL CODIGO YA QUE TIENE ERRORES | 22/06/2016 | MIERCOLES |
| DAR LOS TOQUES FINALES AL PROYECTO | 25/06/2016 | SABADO |
| HARE EL COMMIT NUMERO 4, CON UNA PARTE DEL CODIGO Y LO ESCRITO | 27/06/2016 | LUNES |
| SE ENTREGARÁ EL PROYECTO | 29/06/2016 | MIERCOLES |

# BIBLIOGRAFIA

http://www.desarrolloweb.com/articulos/499.php

http://www.aprenderaprogramar.com/index.php?option=com\_content&view=category&id=68&Itemid=188

http://elvex.ugr.es/decsai/java/

https://codigofacilito.com/

http://danielbrena.blogspot.com/2013/04/juego-del-ahorcado-en-java.html

http://www.discoduroderoer.es/como-generar-numeros-aleatorios-con-el-metodo-math-random-de-java/

http://puntocomnoesunlenguaje.blogspot.com/2012/08/java-scanner.html

https://es.answers.yahoo.com/question/index?qid=20100907170025AAogJdT

# BITAGORA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FECHA | DIA | HORA | ACTIVIDAD REALIZADA | TIEMPO QUE SE UTILIZO |
|  |  |  |  |  |
| 04/06/2016 | viernes | 10:17am | hice lectura del proyecto | 30 minutos |
| 06/06/2016 | lunes | 7:00pm | comencé a investigar el código | 2 horas |
| 08/06/2016 | miércoles | 5:00pm | busque palabras claves, de los métodos para el proyecto | 2 horas |
| 10/06/2016 | viernes | 9:00am | inicie el proyecto | 1 hora |
| 12/06/2016 | domingo | 9:38pm | hice el primer commit (github) | 1 hora con 30 minutos |
| 15/06/2016 | miércoles | 8:37pm | entregue el avance al profesor (github) | 1 hora con 30 minutos |
| 17/06/2016 | viernes | 9:00am | se avanzó en el código | 2 hora |
| 20/06/2016 | lunes | 10:00am | complete la mitad del código y la parte escrita | 2 horas |
| 22/06/2016 | miércoles | 5:00pm | solucione algunos errores en el código | 3 minutos |
| 25/06/2016 | sábado | 8:00pm | termine el proyecto | 3 hora |
| 27/06/2016 | lunes | 1:45am | entregue un 30% de la parte escrita y un 40% de lo programado (github) | 3 horas |
| 29/06/2016 | miércoles | 11:00pm | se entregó el proyecto programado | 3 horas |

# ANEXOS

# C:\Users\Josue\AppData\Local\Microsoft\Windows\INetCache\Content.Word\slide_4.jpgC:\Users\Josue\AppData\Local\Microsoft\Windows\INetCache\Content.Word\curso-java-03-mtodos-e-parmetros-6-728.jpg



# C:\Users\Josue\AppData\Local\Microsoft\Windows\INetCache\Content.Word\82d9d5977132032df4324ce34bab1a07.jpg