UNIVERSIDAD CATÓLICA DE COSTA RICA

INGENIERÍA EN SISTEMAS

PROGRAMACIÓN I

SEGUNDO PROYECTO, JUEGO BUSCAMINAS

ANDRÉS JIMÉNEZ LEANDRO

JOSUE MUÑOZ SOLIS

DANIEL MURILLO RODRIGUEZ

SEDE SAN CARLOS, AGOSTO, 2016

Contenido

[RESUMEN EJECUTIVO 3](#_Toc457847496)

[DESCRIPCION 4](#_Toc457847497)

[ABSTRACT 5](#_Toc457847498)

[OBJETIVOS GENERALES Y ESPESIFICOS 6](#_Toc457847499)

[Objetivo general 6](#_Toc457847500)

[Objetivos específicos 6](#_Toc457847501)

[INTRODUCCION 7](#_Toc457847502)

[DESARROLLO 8](#_Toc457847503)

[RESULTADOS OBTENIDOS 11](#_Toc457847504)

[CONCLUCION 11](#_Toc457847505)

[RECOMENDACIONES 12](#_Toc457847506)

[CRONOGRAMA 12](#_Toc457847507)

[BIBLIOGRAFIA 13](#_Toc457847508)

[BITAGORA 14](#_Toc457847509)

[ANEXOS 15](#_Toc457847510)

# RESUMEN EJECUTIVO

Juego Busca minas

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.

En este juego buscaminas se nos plantea un interesante reto el cual con base a las ultimas clases y prácticas, podemos entender y desarrollar el código de una mejor manera. Parte de lo pedido también fue investigado y analizado por los dos estudiantes que realizaron este trabajo., algunos puntos importantes de este juego programado se basan en el conocimiento de métodos los cuales se aplican de la manera correcta para darle forma y sentido a la lógica de este buscaminas, el alto nivel de desarrollo invita e incluso obliga a una investigación sobre este juego en internet para entender la manera en la que se despliega cada detalle de dicho juego. No obstante, se revisado cada punto de lo escrito para que si fuera acorde con lo solicitado por el profesor.

# DESCRIPCION

En el enunciado del proyecto se plantea un buscaminas con una parte de visualización grafica para ser mostrado únicamente en la consola del IDE de Netbeans, como gran detalle para estos juegos la aplicación de ellos debe ser limpia tanto el código como al correr el mismo.

Gran parte de lo requerido es lo siguiente:

“El proyecto consiste en realizar un programa en el cual el usuario pueda jugar Buscaminas desde la consola, teniendo en cuanta las siguientes consideraciones:

El usuario debe ingresar el tamaño del lado del tablero (L).

El programa generará aleatoriamente la ubicación (coordenadas) de [2\*L] minas. (Tener en cuenta que en cada casilla no podrá quedar más de una mina y que se deben ubicar todas las minas).

Se debe presentar una interface en la cual no se muestre la ubicación de las minas, solo se mostrará lo que el usuario vaya marcando o destapando.

El usuario ingresa una coordenada y una instrucción (marcar (X), destapar o desmarcar).

Si una mina está marcada no se puede pisar.

Si una mina es destapada se pierde el juego.

El juego termina cuando se pierda o cuando se ubiquen exactamente todas las minas.

El número de marcas es igual al número de minas.

Si el usuario destapa una casilla que no tenga minas, debe aparecer en dicha casilla el número de minas que limitan con la coordenada ingresada.

Si el usuario pierde, se debe mostrar la ubicación de todas las minas

Si el usuario destapa una casilla que no tiene minas ni minas que limiten con ella, se deben descubrir todas las casillas que no tengan minas ni vecinas a su alrededor y mostrar las vecinas de estas (en el ejemplo, si se da clic en una casilla 0, se deben destapar hasta los números vecinos a la zona de ceros).”

# ABSTRACT

Game searching for mines., Game programmed with base to the language of programming java, whose language has as definition., Java is a language of programming of purpose general, concurrent, oriented to objects that was designed specifically for having so few dependencies of implemented as out 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 compiled to run on another., In this game seeks to mine is us raises an interesting challenge which with base to the latest classes and practices, can understand and develop the code of a best way. Part of the request was also investigated and analysed by the two students who carried out this work. some important points of this programmed game are based on the knowledge of methods which are applied in the right way to give form and meaning to the logic of this search for mines, the high level of development invites and even forcing a research about this online game to understand the way in which every detail is displayed for that game. However, is revised every point of what has been written for Yes out according as requested by the teacher.

The creation of looking for mines as a game run in console presents a high complexity for the student, much of studying it in class and investigated at home will be applied in this game looking for mines. Importantly to carry the correct order when starting to write a code, since not only is sit down and program, if not you should understand and study the issue before giving solution.

This is to have a final result more optimal and with fewer errors. What is essential in the development of a searching for mines in java it is based on the correct classes management and implementation, could handle packages and would be much better for a much larger order, but that is a decision that is in each student at the time of choosing the way of how you will solve the problem and the creation of a clean and executable code.

# OBJETIVOS GENERALES Y ESPESIFICOS

## Objetivo general

Aplicar los conocimientos relacionados a la programación básica y de estructura de datos, para programas, en un juego buscaminas de baja complejidad, que contenga la aplicación correcta de la convención java y sea desplegado en la consola del IDE de NETBEANS

## Objetivos específicos

El buscaminas sea completamente ejecutable en el IDE de Netbeans, contemplando lo requerido como demostrar la aplicación de los métodos necesarios, que corran sin errores y de manera limpia respetando las exigencias de las convenciones java.

# INTRODUCCION

La creación de busca minas como un juego corrido en consola presenta una alta complejidad para el estudiante, gran parte de lo estudiando en clase e investigado en casa será aplicado en este juego busca minas. Lo importante de llevar un orden correcto a la hora de comenzar a escribir un código, ya que no solo se trata de sentarse a programar, si no que se debe comprender y estudiar el problema antes de darle solución.

Esto para tener un resultado final más óptimo y con menos errores. lo esencial en el desarrollo de un buscaminas en java tiene como base el manejo correcto de las clases y la implementación en ellas, se podría manejar con paquetes y sería mucho mejor para un orden mucho más grande, pero eso es una decisión que queda en cada estudiante al momento de escoger la manera de cómo va a darle solución al problema y la creación de un código limpio y ejecutable.

# DESARROLLO

¿Qué es lo que hay que resolver?

Se debe crear un código donde de manera ordenada, ya sea en 1, 2 o más paquetes incluyamos las distintas clases en la que debemos crear una para el menú, otra para el juego, una para la parte de visualización y obviamente la clase main que debe ir lo más limpia posible si recargarla mucho para cumplir con lo solicitado.

También hay que darle solución al descubrir un numero en el tablero y que se muestre las minas., para esto se aplicaron métodos, de la misma manera que para solicitar datos y una parte de ventana desplegable como la de un menú en el que debemos crear de manera de selección para que el usuario escoja si jugar o no. Por otro la debemos incluir mensaje, los cuales nos mostraran los distintos resultados que tengamos en el juego programado buscaminas.

La importancia de inclusión de la parte visual en la consola de NETBEANS, nos dicta que es en coordenadas de fila y columna o viceversa.

Se tendrá que mostrar si se pisa una mina o si se descubre un número.

¿Cómo se va resolver el problema?

Dicho lo anterior, procedemos a desarrollar el código después de haber analisado paso por paso el problema a resolver, de una manera recomendada por cualquier desarrollador de un software o programa.

Comenzamos creando un main donde solo tendremos el llamado de una de las clases como menú,

De manera seguida se creará un menú con los métodos “case”

También se creará otra clase donde contenga el juego con los métodos de while do, importes de JOptionPane para los mensajes del juego.

De manera seguida, por último, pero no menos importante esta la creación de la clase tablero, que llegaría a ser parte de lo visual de juego programado buscaminas., allí tendremos que tomar en cuenta métodos como el scanner, random, el for, if., para cada una de las excepciones de las columnas y filas al ingresar distintos números y la selección aleatoria de las bombas o minas del juego.

Análisis del problema

Se deberá crear un código completamente ordenado, limpio, tomando en las convenciones java para el debido comentario de cada línea que deba ser explicada, sin espacios ni despilfarros de memoria en la reproducción del código, esto como recomendación.

En si el juego platea el ingreso de datos como coordenadas para que el código debidamente aplicable compare cada coordenada ingresada por el jugador con una mina o bomba y numero, que se han ubicado aleatoriamente por un método como random., se mostrara el resultado de cada una de las coordenadas ingresadas con un mensaje si es el caso de pisar una mina y perder o si se evitó una mina se muestra los números circundantes.

ANALISIS DE LA SOLUCION

Como algunos no conocíamos el juego buscaminas, nos abocamos a la investigación del mismo ya que la composición del mismo era un poco compleja para el nivel de programación de nosotros los estudiantes.

Gracias a la idea brindada por el profesor Andrés recurrimos a analizar el juego WinMine” incluido en los sistemas operativos Windows, con una idea del funcionamiento del juego, iniciamos el análisis del mismo y lo requerido por el proyecto. Gran parte del problema para esto es la parte gráfica, ya que no se puede aplicar el JFRAME la parte grafica de él.

El inicio del código fue gran parte un respaldo de lo investigado en el internet, para entender la estructura del código y de cómo debería aplicarse y verse. Después de lo investigado se comenzó creando un proyecto normal java, sin Jframe, en el cual creamos una clase main en el que irían los llamados únicamente de menú. La clase menú se realizó después e incluimos en ella parte de lo que se despliega al iniciar el juego buscaminas, con los case para la libre escogencia del jugador si ya sea que quisiera jugar o salir del juego, parte de esto se logra con int, this, el método switch el cual nos dará la opción y por ultimo un default que nos excluye algunos caracteres indeseados en el menú.

La segunda parte de la realización del código tiene que ver con las clases juego y tablero. En la clase juego creamos los métodos de while do, esto para conocer el resultado de los datos de la clase tablero, en él se compara este resultado para ser imprimido en un mensaje de final, si se perdió o se ganó. Se debe tomar en cuenta la clase tablero para obtener los datos de él.

En la clase tablero se crea gran parte de lo visual y contiene las columnas y filas del juego desplegado en la consola del IDE de Netbeans, para esto se creó un int bomba, int pizarra, iniciarBomba(), randomBomba(), llenarCampos(), iniciarTablero()… cada una con sentido obvio por su nombre, donde rellenaremos espacios con minas o bombas en iniciar bombas, luego la escogencia aleatoria con el random, y inicializamos el tablero y el juego con iniciar tablero., se mostrara bombas con mostrar.

A pesar de que el código corre de manera óptima, hubo algunos detalles que, si se tomaron en cuenta, pero no se depuraron del todo. Estos detalles son mínimos, pero si dan al traste con el juego programado en algunos lapsos a la hora de correrlo, por ejemplo, a la hora de ingresar un carácter diferente a un número cuando se pide las coordenadas., si se tomó en cuenta ya que la solución a esto es incluir esto dentro de un ciclo con un try catch, para evitar que se bote el programa.

Como aspecto importante, la inclusión de la visualización de las minas circundantes a un espacio si mina, no se solucionó ya que no que la excepción no se encontró como crearla y aplicarla.

# RESULTADOS OBTENIDOS

Se obtuvo un código funcional, completamente reproducido en la consola de netbeans y con el menú y otro mensaje el joptionpane, también se muestra los mensajes de victoria y derrota sin ningún problema, la visualización es casi idéntica a la solicitado en el proyecto. Se logra escoger coordenadas por el usuario y la decisión en el menú es la indicada.

# CONCLUCION

La creación de este II proyecto es ampliamente un reto ya que por la dificultad del mismo nos retó a resolver el problema de la manera más adecuada, primero conocer un poco sobre aplicación de winmine para desarrollar la propia, pero en la consola de netbeans.

La importancia de la aplicación de algunos métodos y la inclusión de otros como los del menú que antes no los había aplicado en un juego programado de java.

Si bien el juego solo era en consola al principio se desarrolló una igual, pero en la parte grafica de jframe y así se exploró un poco también sobre esa área de la programación.

Volviendo a el programa en si podemos decir que la importancia en él es la condiciones y el llamado en cada una de las clases ya que esto genera que el programa corra con naturaleza, sin olvidar el orden y la importancia también del contrariado de cada línea.

# RECOMENDACIONES

Tomar en cuenta el orden del código

La importancia de extends object

El paqueteando para una mejor distribución

El trabajar con distintas clases y no cargar mucho en main., prácticamente no cargarlo…

|  |  |  |
| --- | --- | --- |
| CRONOGRAMA | | |
|
|
| EN QUE TRABAJARE | FECHA | DIA |
|
| LEAMOS Y ANALIZAMOS EL PROYECTO | 04/07/2016 | LUNES |
| INICIAMOS A INVESTIGAR SOBRE EL JUEGO WINMINE | 09/07/2016 | SABADO |
| SE COMENZARÁ A CREAR EL UML, PARA EL AVANCE | 13/07/2016 | MIERCOLES |
| NOS REUNIREMOS PARA HABLAR DE LA COMPOSICION DEL CODIGO | 16/07/2016 | DOMINGO |
| SE CREARÁ EL PROYECTO EN GITHUB | 19/07/2016 | MARTES |
| SE ENTREGARÁ EL UML AL PROFESOR | 20/07/2016 | MIERCOLES |
| TRABAJAR EN EL INICIO DEL CODIGO | 25/07/2016 | LUNES |
| SE ENTREGARÁ EL SEGUNDO AVANCE | 27/07/2016 | MIERCOLES |
| INICIAR LA PARTE ESCRITA | 27/07/2016 | MIERCOLES |
| TRABAJAR EN EL CODIGO Y LA PARTE ESCRITA | 1/08/2016 | LUNES |
| CONSTRUIR CODIGO | 3/08/2016 | MIERCOLES |
| CAMBIAR Y AVANZAR EN EL CODIGO | 5/08/2016 | VIERNES |
| SE REALIZARÁ LAS CLASES LA PRIMERA | 7/08/2016 | DOMINGO |
| SE TERMINA EL RESTO DE LAS CLASES | 8/08/2016 | LUNES |
| SE COMPLETARÁ EL CODIGO Y UNA PARTE DE LO ESCRITO | 9/08/2016 | MARTES |
| SE ENTREGARÁ EN PROYECTO COMPLETO | 10/08/2016 | MIERCOLES |

# CRONOGRAMA

# BIBLIOGRAFIA

http://winmine-cafesanu.softonic.com/

http://www.aprenderaprogramar.com/index.php?option=com\_content&view=article&id=168:repaso-arrays-o-arreglos-unidimensionales-en-java-tipos-de-inicializacion-ejemplos-de-codigo-cu00903c&catid=58:curso-lenguaje-programacion-java-nivel-avanzado-i&Itemid=180

https://www.youtube.com/watch?v=rXRAQDO5STw

http://www.codifica.me/codigo/java/codigo-busca-minas-en-java/

http://tubuscaminas.blogspot.com/

# BITAGORA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FECHA | DIA | HORA | ACTIVIDAD REALIZADA | TIEMPO QUE SE UTILIZO |
|  |  |  |  |  |
| 04/07/2016 | LUNES | 9:00pm | Se leyó el proyecto | 30 minutos |
| 09/07/2016 | SABADO | 7:00pm | comenzamos a investigar el juego winmine | 4 horas |
| 13/07/2016 | MIERCOLES | 5:00pm | Comenzamos a crear el UML | 2 horas |
| 16/07/2016 | DOMINGO | 9:00am | Discutimos sobre la realización optima del código | 3 hora |
| 19/07/2016 | MARTES | 1:16am | hice el primer commit (github) | 1 hora con 30 minutos |
| 20/07/2016 | MIERCOLES | 10:46pm | entregue el avance UML, al profesor (github) | 1 hora con 30 minutos |
| 25/07/2016 | LUNES | 9:05pm | trabajamos en el código | 3 hora |
| 27/07/2016 | MIERCOLES | 3:24pm | Iniciamos la parte escrita | 2 horas |
| 27/07/2016 | MIERCOLES | 5:00pm | Entregamos el segundo avance | 3 minutos |
| 1/08/2016 | LUNES | 10:05pm | Trabajamos en el código y la parte escrita | 3 hora |
| 3/08/2016 | MIERCOLES | 7:28pm | Hicimos grandes cambios en el código | 3 horas |
| 5/08/2016 | VIERNES | 11:51pm | Se continuó trabajando en el código | 3 horas |
| 7/08/2016 | DOMINGO | 11:29pm | Se trabajó en la clase tablero | 4 horas |
| 8/08/2016 | LUNES | 8:38pm | Se agregó más en las distintas clases | 5 horas |
| 9/08/2016 | MARTES | 11:17pm | Se terminó el código y un 80% de lo escrito | 6 horas |
| 10/08/2016 | MIERCOLES | 11:50pm | Se creó el archivo rar y se entregó el proyecto | 2 horas |

# ANEXOS

