

Proyecto Final de Programación Funcional – Ciclo 01/2014

“Bolita busca su queso”



Objetivo

Que los y las estudiantes:

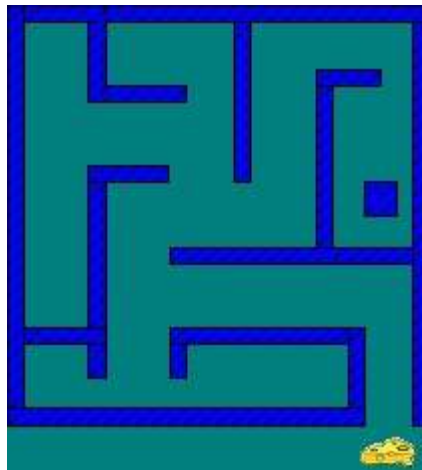
- Pongan en práctica los conocimientos adquiridos a lo largo del semestre en la materia.
- Desarrollen la habilidad de trabajo en equipo, de gran importancia para su vida laboral.



Marco de referencia

En la universidad de Harvard, hace varios años, el doctor Robert Rosenthal llevó a cabo una extraña serie de experimentos con estudiantes y ratas.

Se realizó con tres grupos de estudiantes y tres grupos de ratas. Al primer grupo de estudiantes se le dijo: *"Han tenido suerte. Trabajarán con ratas genios. Fueron criadas en base a su inteligencia brillante. Llegarán al final del laberinto antes de un parpadeo, y comen mucho queso, por lo que tendrán que tener una provisión importante"*.



Al segundo grupo se le dijo: *"Trabajarán con ratas promedio. En un tiempo normal para casi todos los animales, llegarán al final del laberinto, y comen una proporción considerable de queso. No se puede esperar mucho de ellas, apenas un comportamiento normal"*.

Al tercer grupo: *"Les ha tocado ratas torpes. Si llegan al final del laberinto será de pura suerte. Casi con seguridad no pasarán las pruebas. Son obtusas, de rendimiento muy bajo. Quizá ni necesitan comprarles queso. Sólo pongan un letrero al final del laberinto, que diga: queso"*.

Durante seis semanas los estudiantes llevaron adelante el experimento bajo rigurosas normas científicas. Las supuestas ratas genios respondieron como genios. Las supuestas

Los laberintos pueden ser contruidos en cualquier editor de texto ASCII, como bloc de notas o gedit.

- b) Asimismo debe poder solicitar al usuario que introduzca un pseudónimo y que elija un nombre de archivo que contiene un laberinto. Este laberinto será cargado en una estructura tipo matriz. Esto se reflejará en pantalla por medio de dibujar en una ventana gráfica un laberinto, con sus muros, corredores y salida visibles por medio de elementos gráficos. Su ratón tendrá la misión de salir de cualquier laberinto, si quiere llegar al queso.
- c) Una vez cargado el laberinto, el programa requerirá la ubicación de inicio se su ratón en el interior. Puede hacer esto pidiendo coordenadas desde teclado o pulsando con el puntero en algún punto de algún corredor. Se mostrará un elemento gráfico que simbolice al ratón en pantalla, ubicado en dicha coordenada.
- d) A continuación se deberá iniciar la búsqueda de la salida por algún medio, por ejemplo, pulsando un botón.
- e) Deberá también implementarse el algoritmo de búsqueda de la salida. Tome en cuenta que el usuario no va a indicar al ratón hacia donde moverse para salir del laberinto. La única intervención del usuario es el instante previo, cuando indica las coordenadas iniciales de ubicación del ratón y le indica al programa que inicie la búsqueda de la salida. Es recomendable que el algoritmo de búsqueda de la salida sea recursivo.
- f) Una vez se haya encontrado la salida se marcará la ruta que ha seguido el ratón desde el punto de partida hasta ella. Esta ruta no tomará en cuenta los intentos fallidos que hayan obligado a dar pasos hacia atrás al ratón.
- g) Desplegará por algún medio, ya sea en un espacio reservado de la misma ventana o en una ventana emergente, el tiempo que se tardó el ratón en encontrar la salida.
- h) Luego de finalizar la búsqueda de la salida y de mostrar el tiempo invertido, esta información se almacenará una lista lineal. Los datos almacenados, en cada ejecución del juego podrían ser: nombre de jugador, laberinto utilizado, coordenadas de partida y tiempo de salida. Al salir totalmente del juego, el contenido de la lista será almacenado en un archivo en disco.
- i) Preguntar al usuario si desea jugar de nuevo. Si la respuesta es sí, se iniciará nuevamente pidiendo un nombre de archivo para cargar un laberinto, puede asumir el mismo laberinto del juego previo. Si la respuesta es no, finalizará el juego.
- j) El entorno de su proyecto deberá proveer una opción para poder ver el "Top Ten" de los mejores tiempos de salida. Mostrará nombre del jugador, nombre del laberinto y tiempo de salida. Podrá mostrar la información en orden alfabético según los nombres de los jugadores o en orden descendente de tiempo de ejecución. Esta información será leída del archivo y almacenada en una lista lineal, que luego será manipulada para mostrar la información descrita.
- k) Su proyecto deberá contar con una ventana de ayuda que contendrá indicaciones de cómo elaborar un archivo de laberinto, la explicación de caracteres utilizados para la construcción de los laberintos, explicación de la secuencia del juego y otros detalles del juego que los usuarios deban saber. Esta puede ser construida en SDL o pueden elaborar un archivo pdf que será invocado por la opción de ayuda.

- I) La construcción de todas las funcionalidades mencionadas tendrán un valor del 80%. La construcción del entorno gráfico tendrá un valor del 20%.

El programa deberá ser capaz de resolver el problema para cualquier laberinto que se le proporcione, no importando su configuración: cantidad y forma de conexión de sus pasillos y ubicación de la salida.

Formato de la entrega

El proyecto deberá ser entregado en un archivo comprimido (estrictamente un zip). Deberán colocarse entre comentarios, en el archivo de la función *main()*, los nombres de l@as integrantes del grupo. En el interior del archivo comprimido deberá encontrarse el proyecto completo, que incluirá: programas en C++, archivos de imágenes utilizados para dibujar los elementos gráficos y algunos archivos de laberintos que el grupo haya probado con su programa.

Grupos de trabajo

El proyecto deberá realizarse en grupos con un máximo de cuatro estudiantes.

Fecha de entrega y defensa del proyecto

La entrega del proyecto será el domingo 3 de diciembre, hasta la medianoche. El grupo deberá enviar el archivo comprimido del proyecto a los correo del profesor e instructores para que cualquiera de ellos pueda bajar el proyecto al momento de la defensa.

La defensa del proyecto será el martes 5 de diciembre por la mañana, a partir de las 8:00 am. Se emitirá días antes un listado con horarios de defensa para que se anoten. Todos los integrantes deberán presentarse a la defensa.