

Proyecto 3: Resolución de Laberintos

Miércoles 16 de Mayo

I. DESCRIPCIÓN GENERAL

Este proyecto es una ampliación del proyecto previo. Se deben programar varios algoritmos para encontrar la solución a un laberinto. Las rutas encontradas por cada uno serán desplegadas sobre el laberinto y se contabilizará el número de casillas que cada uno visita¹. La interfaz debe ser gráfica. Toda la programación debe realizarse en C sobre Linux. No se pueden cambiar las especificaciones de este documento.

II. PREÁMBULO

Antes que nada, se deben corregir todos los errores y problemas detectados en su proyecto previo. Luego se le dará funcionalidad al botón **Resolver**. Este botón está normalmente deshabilitado, pero se pone activo una vez que haya un laberinto desplegado (ya sea que haya sido cargado o generado).

III. RESOLUCIÓN

Si se pide resolver el laberinto, se desplegará una sub-ventana donde el usuario seleccionará uno o más de los algoritmos disponibles para resolverlo. Por defecto, todas las casillas estarán preseleccionadas. Una vez que se indiquen los algoritmos a usar, habrá un par de botones para cancelar o para ejecutar.

Cada grupo debe investigar e implementar los siguientes 6 algoritmos:

- Ratón aleatorio.
- Regla de la mano derecha.
- Regla de la mano izquierda.
- Algoritmo de Pledge.
- Algoritmo de Trémaux.
- Algoritmo de Fattah.

Cada algoritmo marcará una ruta en el laberinto (incluyendo las ocasiones en que deba dar “vuelta en U”). Asígnele un color diferente a cada algoritmo e inclusive un carril dentro del laberinto para que los 6 puedan coexistir con la mayor claridad posible. Toda esta funcionalidad debe ser compatible con las capacidades de navegación solicitadas para el proyecto previo.

Trabajo extra opcional 1: haga que cada ruta se vaya desplegando paso a paso con una velocidad apropiada para ver el avance como una animación.

Se debe desplegar en un recuadro la cantidad de casillas visitadas por cada algoritmo solicitado, usando colores consistentes con las rutas.

Trabajo extra opcional 2: si su proyecto previo generaba laberintos hexagonales... resuelvalos.

IV. REQUISITOS INDISPENSABLES

La ausencia de uno solo de los siguientes requisitos vuelve al proyecto “no revisable” y recibe un 0 de calificación inmediata:

- Todo el código debe estar escrito en C (no C++).
- El proyecto debe compilar y ejecutar en Linux.
- Todas las interfaces deben ser gráficas.
- Se debe usar GTK, Glade y Cairo.
- No debe dar “Segmentation Fault” bajo ninguna circunstancia.

Trabajo extra opcional 3: hacer la demostración en una máquina que levante Linux de manera real (puede ser dual), es decir no usar máquinas virtuales.

V. FECHA DE ENTREGA

Revisiones a las 11:30am el **Miércoles 16 de Mayo**. Mande además un .tgz con todo lo necesario (fuentes, makefile, readme, etc.) a torresrojas.cursos@gmail.com. Ponga como subject: A.A. - Proyecto 3 - Fulano - Mengano, donde Fulano y Mengano son los 2 miembros del grupo.

¹Cualquier uso de software tomado de Internet para resolver laberintos se considerará fraude académico y será severamente castigado