



USAID
FROM THE AMERICAN PEOPLE

AMERICAN SCHOOLS
& HOSPITALS ABROAD

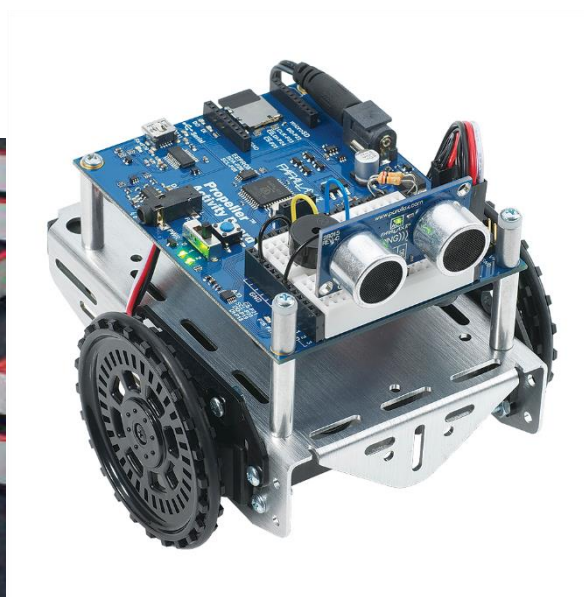
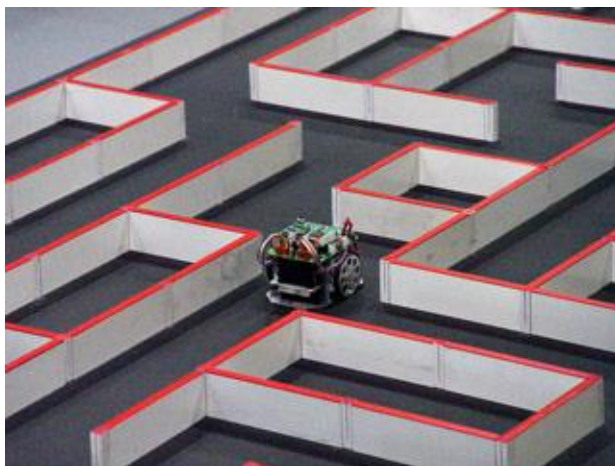


Un Robot sale de un laberinto

Competencias a desarrollar:

- Diseña un algoritmo general que permita que el robot salga de cualquier laberinto.
- Programa el robot para que use sus sensores para salir de un laberinto real.

TEMA: Su grupo de proyecto, debe programar un robot Parallax Activitybot¹ para que salga de un laberinto. Similar a: <https://www.youtube.com/watch?v=YqTaLl2camI>



¹ Robots donados por el programa ASHA American Schools and Hospitals Abroad

Fase 1. Diseño de un algoritmo para salir de cualquier laberinto

Fecha de entrega: semana del 12 al 16 de febrero de 2018

Instrucciones:

- Investigue los diferentes algoritmos que existen para salir de un laberinto
- Explique las razones por las cuales seleccionó el algoritmo que implementará. Si desechó los algoritmos documentados y decidió hacer uno propio explique por qué. Recuerde que puede basarse en los análisis de complejidad para poder justificar sus elecciones. Tenga en cuenta que debe utilizar el mismo algoritmo que diseño y probó en la segunda fase del proyecto.
- Explique la estructura de datos a utilizar, y diseñe el diagrama de clases que utilizará.
- Explique el algoritmo que sirve para salir del laberinto agregando el pseudocódigo o el diagrama de flujo.
- Pruebe que el algoritmo es útil para ayudar a un elemento a salir de cualquier algoritmo, para eso implemente el algoritmo en un entorno virtual². Puede cargar laberintos prediseñados, pero el algoritmo no se debe modificar, por lo que debe salir de cualquier laberinto.
- Mida el tiempo en que se demora el elemento en encontrar la salida en laberintos de distinta complejidad, siempre usando el algoritmo seleccionado.
- Suba todos los archivos a un repositorio de github, recuerde que debe haber **varios commits** de **todos** los miembros del equipo, durante todo el tiempo de disponibilidad de la tarea.

Productos a entregar

- Archivo .pdf con:
 - o La investigación de los algoritmos existentes para resolver la situación que se presenta.
 - o La explicación de la selección del algoritmo a utilizar en el proyecto.
 - o El pseudocódigo o diagrama de flujo del algoritmo.
- Archivos de código del programa implementado en java
- Vínculo del repositorio de github

Evaluación:

Se evaluarán los siguientes aspectos:

ASPECTO	PUNTEO
Investigación sobre algoritmos existentes para resolver la situación que se presenta.	10
Explicación de la selección del algoritmo. Se basó en elementos como complejidad de los algoritmos para decidir qué algoritmo implementar. La explicación es clara y convincente.	20
Explicación de la estructura de datos utilizada, incluyendo los ADTs que diseñó y/o utilizó. Está representado en un diagrama de clases utilizando UML.	10
Pseudocódigo o diagrama de flujo del algoritmo a utilizar para recorrer el laberinto	5
Programas elaborados y dirección del repositorio del control de versiones. Se evidencian varios commits de TODOS los miembros del equipo de desarrollo, durante toda disponibilidad de la actividad.	10
Ejecución del programa. Se prueba con la ejecución del programa que el algoritmo funciona sin sufrir cambios para cualquier laberinto. La implementación del algoritmo coincide con el pseudocódigo o diagrama de flujo propuesto.	40
Medición de los tiempos de ejecución del algoritmo con laberintos de distinta complejidad y tamaño.	5
TOTAL:	100

² Puede usar el ambiente Rurple (Python)

Fase 2. Implementando el algoritmo en el robot

Fecha de entrega: semana del 12 al 16 de marzo de 2018

Sobre el robot Parallax ActivityBot.

1. Se tienen disponibles 20 robots, los cuales puede pedir prestados en la oficina J-103 por un plazo limitado para dar oportunidad a otros alumnos a usarlo³.
2. La única inversión que deberá realizar su grupo será para adquirir baterías, porque estas no le serán proporcionadas.
3. Adjunto a esta guía puede ver el instructivo inicial que han elaborado otros alumnos sobre este robot y un pequeño programa en C para manejar algunos de sus sensores.
4. Tutorial sugerido para conocer y programar el robot: <http://learn.parallax.com/activitybot>

La tarea:

1. Su grupo deberá utilizar C para programar al robot. En otros cursos ya han empleado C, pero si es la primera vez, este lenguaje tiene una sintaxis parecida a Java.
2. El programa que su grupo desarrolle le permitirá al robot salir de un laberinto. El algoritmo utilizado debe coincidir con el diseñado e implementado en la fase 1 del proyecto. El laberinto será el mismo para todos los grupos.
3. Usted debe ajustar los desplazamientos del robot, lo cual implica programar sus sensores y servos o motores para que estén en sintonía con la programación del sistema diseñado.

Productos a entregar:

Su grupo deberá entregar:

1. Un instructivo para preparar al robot para que pueda salir del laberinto. Este instructivo debe mostrar como instalar el software necesario, tanto en la computadora de desarrollo, como en el robot. La información sobre los sensores y servos o motores utilizados y sus parámetros adecuados para que logre salir del laberinto.
2. La descripción del desempeño de la estructura de datos y el algoritmo utilizado cuando fue implementado en el robot comparada con la simulación hecha en la fase 1 del proyecto.
3. El programa o programas fuente elaborados (con documentación interna) y el vínculo del repositorio de github, con la evidencia de colaboración de TODOS los integrantes del equipo de desarrollo.
4. El video que muestre como funciona su robot y programa, en un laberinto físico de prueba que cree su grupo.

Presentación en clase:

El día de la presentación, su robot deberá salir de un laberinto que han preparado los auxiliares. Todos los grupos usarán el mismo laberinto. Se seleccionarán los primeros tres robots que logren salir en menor tiempo del laberinto, para representar a la sección en un concurso.

³ Su grupo podrá utilizar otro tipo de robot, pero debe ser aprobado por su catedrático y programarse en C.

Si su robot no logra salir del laberinto en un plazo de 5 minutos, se le bajarán 20 puntos de la parte de ejecución del programa, y se le dará una nueva oportunidad al final de todas las presentaciones de los grupos. Si aún allí no logra salir del laberinto, se le quitará la nota completa de la ejecución del programa.

Concurso entre secciones del curso.

En cada presentación se seleccionarán los tres robots que hagan menos tiempo para salir del laberinto. Estos participarán en un concurso que se hará en un lugar público de la UVG, tal como la plaza Paiz-Riera, para que todos los alumnos los vean. El ganador será aquel que salga del laberinto en el menor tiempo, y tendrá un premio sorpresa y puntos extras en esta fase del proyecto. El laberinto a utilizar en el concurso será elaborado por todos los auxiliares de las diferentes secciones para garantizar que no se favorece el diseño de ningún grupo.

Evaluación:

Se evaluarán los siguientes aspectos:

ASPECTO	PUNTEO
Instructivo de preparación del robot y ambiente de programación	20
Programas elaborados y dirección del repositorio del control de versiones. El repositorio debe evidenciar varios commits de todos los integrantes del equipo.	30
Video de funcionamiento (el robot sale del laberinto creado por el grupo)	10
Funcionamiento en clase (si sale del laberinto antes de 5 minutos se da la nota completa). Se le bajan 20 puntos por una segunda oportunidad. Si no sale del laberinto no obtiene puntos en este rubro.	40
Adicionales: su robot es seleccionado para el concurso.	10
Adicionales: su robot GANA el concurso.	10
TOTAL:	120