

PROYECTO INTEGRADOR – BÚSQUEDAS LOCALES

En el entregable anterior utilizaste el medio ambiente que definiste en tu primera entrega para realizar búsquedas informadas y no informadas.

Para este tercer entregable, utilizarás tus conocimientos de búsquedas locales. Además, utilizaras la librería Simple AI para implementar búsquedas en el medio ambiente.

Ahora, el rover deberá de encontrar el lugar más profundo y, en otra búsqueda, el lugar menos profundo del medio ambiente. Toma en consideración que:

1. Si hay varias regiones con la misma profundidad, con que encuentre una de ellas, esa es una solución aceptable.
2. El rover no puede moverse en el medio cuando la diferencia entre regiones es mayor a 1.
3. Tampoco puede subirse a los obstáculos.
4. El rover se mueve como cualquier automóvil.

Además, el rover tiene un gasto mayor (costo) mientras más alejado se encuentra del nivel inicial o nivel cero (“nivel de mar”).

Instrucciones

Con tu equipo del curso realiza lo siguiente:

1. Define nuevamente el medio ambiente (si no ha cambiado, entrega lo mismo de antes).
2. Define los 4 elementos necesarios para realizar una búsqueda local con recocido simulado (diapositiva 24).
3. Utilizando la librería Simple AI realiza un programa para las siguientes búsquedas:
 - a. Hill-climbing
 - b. Random-restart Hill climbing
 - c. Simulated Annealing

4. Realiza un análisis de las búsquedas realizadas. Comparalas y justifica cuál es mejor argumentando claramente las razones.

Entregables

1. Documento PDF con la información requerida en la sección de Instrucciones, puntos 1 y 2.
2. Código en Python de tu implementación de la búsqueda.
3. Capturas de pantalla de al menos tres corridas por cada una de las 3 búsquedas solicitadas en el punto 3 de las instrucciones (anéxalas al documento PDF).
 - a. En una corrida muestra cómo encuentra la posición más profunda y en la otra corrida muestra cómo encuentra la posición menos profunda.
4. En el documento PDF añade una sección de “Análisis de las búsquedas” en donde analices y compares entre sí las búsquedas realizadas (justifica claramente).
5. Opcional (10 puntos extra): realiza también una función que implemente un algoritmo de calendarización de temperaturas para el método de Recocido Simulado (debe de ser alguno de los algoritmos que vimos en clase – indicar cuál es, pero no puede ser el mismo que viene en el código de *knapsack.py* que se les dio en una actividad).