

---

## EXPLORACIÓN R2E2

---

201807066 – Brayan Andrés Cholotío Tum

### Resumen

El proyecto uno del curso de Introducción a la Programación y Computación 2, tiene como fin el uso de la implementación de memoria dinámica, implementando los Tipos de Datos Abstractos (TDA), con la finalidad de comprender el proceso interno que realiza el ordenador para destinar un espacio en memoria para cualquier acción a ejecutar.

Se hizo la implementación de una TDA lista enlazada simple en la cual guardamos los terrenos obtenidos de nuestro archivo XML, para nuestro elemento terreno, obtiene distintos atributos, los cuales son: Nombre, dimensiones del terreno, coordenadas de inicio, coordenadas de finalización y posición del terreno con valor.

Las posiciones del terreno fueron almacenados en un TDA matriz ortogonal, en la cual se guardaron las posiciones del terreno con su respectivo valor.

Para la parte grafica se realizó la implementación de graphviz, con el cual presentamos al usuario el terreno de manera entendible y con una mejor presentación visual.

### Palabras clave

TDA: Tipo de dato abstracto

Nodo: Punto de intersección de datos.

XML: Lenguaje de marcas extensibles

### Abstract

*Project one of the Introduction to Programming and Computing 2 course aims to use the implementation of dynamic memory, implementing Abstract Data Types (ADT), in order to understand the internal process carried out by the computer to allocate a space in memory for any action to be executed.*

*The implementation of a simple link TDA list was made in which we save the lands obtained from our XML file, for our terrain element, it obtains different attributes, which are: Name, dimensions of the terrain, start coordinates, end coordinates and position of the land with value.*

*The terrain positions were stored in an orthogonal matrix TDA, in which the terrain positions with their respective value were saved.*

*For the graphic part, the implementation of graphviz was carried out, with which we presented the terrain to the user in an understandable way and with a better visual presentation.*

### Keywords

*ATD: Abstract data types*

*Node: Data intersection point.*

*XML: Extensible markup language*

## Introducción

El proyecto 1 se realizó usando tipos de datos abstractos, para su implementación en lenguaje Python se realizó con listas enlazadas simples, usando las clases listaTerrenos y la clase nodoTerreno, en la cual se almacenan los datos: Nombre, fila, columna, posición de inicio, posición de fin, matriz de posiciones.

Para la matriz de posiciones se usaron las clases encabezado, donde se almacenaron las listas de las posiciones con la cual se enlaza el valor, en la clase matriz, se realizan los métodos de inserción y de mostrar los terrenos almacenados.

Al querer mostrar un terreno, deberemos buscarlo por medio del nombre, si se encuentra almacenado se abrirá en formato .jpg, grafica generada por medio de graphviz.

## Desarrollo del tema

La Agencia Guatemalteca de Investigación Espacial (AGIE) ha diseñado un nuevo robot de exploración, llamado r2e2, que tiene la habilidad de explorar nuevos terrenos, este nuevo robot puede moverse en todo tipo de terrenos, sólo necesita más combustible para moverse en terrenos accidentados, y menos combustible en terrenos planos. El único problema con r2e2 es que solo puede moverse ortogonalmente, es decir, únicamente puede moverse en dirección Norte, Este, Sur y Oeste de su posición. R2e2 ha sido diseñado para comunicarse con el satélite Quetzal01 para tener una imagen del terreno que va a explorar, así que debe seleccionar la mejor ruta para llegar al punto de destino, r2e2 siempre debe seleccionar el camino que necesita el

mínimo uso de combustible para completar su exploración, sin embargo, los científicos que están experimentando con r2e2 necesitan un programa que calcule el camino que utilice la mínima cantidad de combustible. La máxima cantidad de combustible que r2e2 puede almacenar son 9,999 unidades. El terreno que r2e2 recibe desde el satélite Quetzal01 está dividido en una cuadrícula, donde cada celda de dicha cuadrícula tiene asignada la cantidad de combustible que r2e2 necesitará para pasar a través de esa celda, r2e2 también recibe las coordenadas de inicio y final del terreno de exploración, la imagen 1 muestra un ejemplo de la información recibida desde el satélite Quetzal01.

La Universidad de San Carlos de Guatemala cooperará desarrollando el algoritmo que r2e2 necesita para identificar el camino con menor consumo de combustible para moverse por el área de exploración y usted ha sido seleccionado para desarrollar esta tarea. La figura 2 ejemplifica el efecto del algoritmo sobre la información recibida por el satélite Quetzal01.

Para el correcto desarrollo del proyecto se implementaron TDA: Lista enlazada simple y matriz ortogonal. La cual los datos que fueron almacenados en los TDA fueron obtenidos por medio de un archivo XML

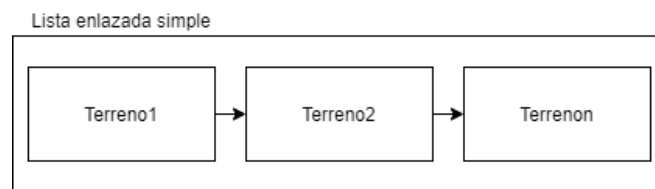


Figura 1. Manera de almacenar el terreno en la lista enlazada simple.

Fuente: Elaboración propia

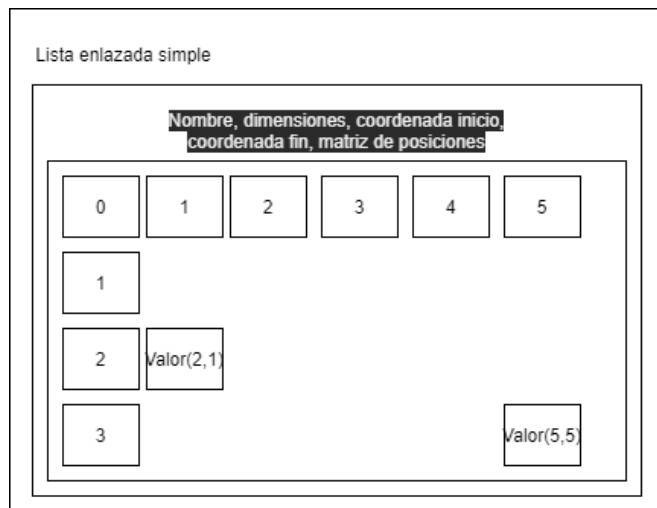


Figura 2. Manera de almacenar las posiciones en el nodo de la lista enlazada simple.

Fuente: Elaboración propia

El archivo XML tiene el siguiente formato ya definido.

```
<terrenos>
  <terreno nombre="terreno1">
    <dimension>
      <m>3</m>
      <n>3</n>
    </dimension>
    <posicioninicio>
      <x>1</x>
      <y>1</y>
    </posicioninicio>
    <posicionfin>
      <x>3</x>
      <y>3</y>
    </posicionfin>
    <posicion x="1" y="1">1</posicion>
    <posicion x="1" y="2">1</posicion>
    <posicion x="1" y="3">5</posicion>
    <posicion x="2" y="1">4</posicion>
    <posicion x="2" y="2">1</posicion>
    <posicion x="2" y="3">4</posicion>
    <posicion x="3" y="1">3</posicion>
    <posicion x="3" y="2">1</posicion>
    <posicion x="3" y="3">1</posicion>
  </terreno>
</terrenos>
```

Ingreso 1. Formato del archivo XML.

Fuente: Elaboración proporcionada.

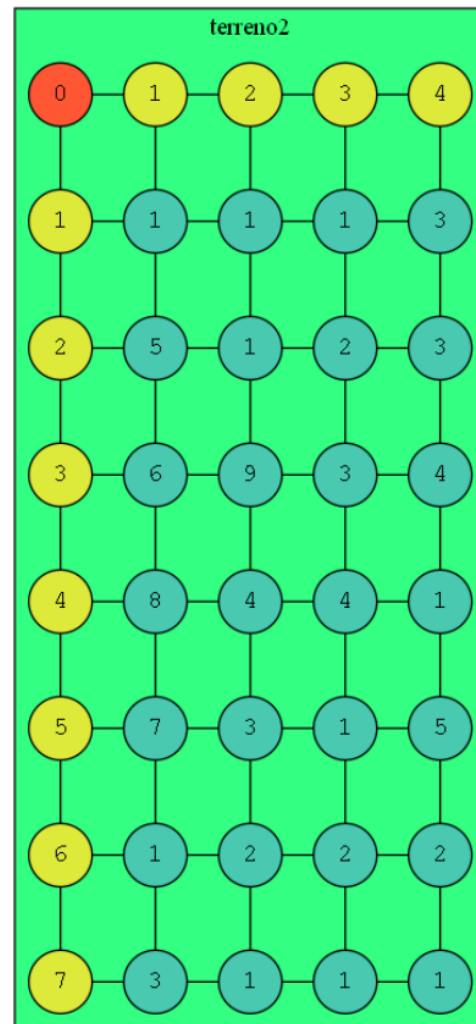


Figura 3. Ejemplo de la generación del terreno por medio de graphviz.

Fuente: Elaboración propia

La respectiva solución al proyecto tiene como fin el implementar algoritmos de recorrido del camino más corto, usando el algoritmo dijkstra. Con la cual podemos calcular la cantidad mínima de combustible que puede ser consumido.

## Conclusiones

Exploración R2E2 nos brinda una idea mucho más clara y concisa, la forma en la cual se hace el uso de la memoria dinámica.

Como estudiantes de la carrera de ingeniería en ciencias y sistemas es de vital importancia tener dichos conocimientos, los cuales son pieza importante para nuestro crecimiento como futuros Ingenieros. Esto se debe a que se nos presentan problemas que pueden asemejarse en el ámbito laboral, el cual nosotros debemos dar soluciones eficientes sin sobrecargar los sistemas.

## Referencias bibliográficas

\*Algoritmos para la ruta más corta

[http://www.udb.edu.sv/udb\\_files/recursos\\_guias/informatica-ingenieria/programacion-iv/2019/ii/guia-10.pdf](http://www.udb.edu.sv/udb_files/recursos_guias/informatica-ingenieria/programacion-iv/2019/ii/guia-10.pdf)

\* Estructuras de datos en Python

[https://www.youtube.com/watch?v=kvmKDwrBEE\\_M&list=PLCh9J2\\_B\\_crLJN8L\\_1aVfJL-kjfEiazvv&ab\\_channel=CesarRamos](https://www.youtube.com/watch?v=kvmKDwrBEE_M&list=PLCh9J2_B_crLJN8L_1aVfJL-kjfEiazvv&ab_channel=CesarRamos)

\* Grafos: Implementación algoritmo Dijkstra

<https://yoandroide.xyz/grafos-implementacion-algoritmo-dijkstra-en-java/>

\*Graphviz

<https://graphviz.org/documentation/>

\*TDA: Tipo de datos abstractos

<https://users.dcc.uchile.cl/~bebustos/apuntes/cc30a/TDA/>