



Tarea 3

1. Objetivos

- Aplicar conceptos y nociones de estructuras de datos para la modelación de un problema complejo.
- Implementar interfaces y tipos genéricos para permitir el acceso a código externo.

2. Introducción

Estimado viajero, escucha mi historia... de los días del futuro pasado. Día... día aquel en que el general Zod-To triunfó. Triunfo... triunfo el de aquellos necios. Necios, sí, necios que bailan al son de la música - pff ‘‘*música*’’. Esos necios no saben la verdad, no saben NADA! Nada saben de aquel fatídico incidente... *incidente? No... tragedia. Sí! Tragedia!* De haber sabido, jamás - JAMÁS - hubiese programado esa maldita simulación... *Simul... simón... dice?* NO! BASTA! Esto ya quedó atrás. Atrás dejé esa maldita ciudad, ciudad del demonio. ARGH! NO! Vuelve al tema! Esta es la última oportunidad que tengo! Aquí va mi último intento.

Esta carta va dirigida a aquellos que desean saber la verdad, la pura y santa verdad. Aquel día, hace ya... argh, no recuerdo... maldita oscuridad... Aquel día de la simulación final no todo fue tan limpio como los Adrianinos (*false!* RoYgBiVianos!) creen... Durante los años he ido escribiendo la verdadera historia en las paredes de esta caverna. Confíen en el corazón de las cart... digo, de los colores! La verdad llegará a ustedes. Está todo aquí, todo! Ha llegado mi momento. *Momento... momentum.. mom* - NO! Ya no aguanto más! Oh, osado viajero! Por favor comunica al resto del planeta el conocimiento que adquieras de mis maravillosas escrituras! *Escritura... escribir.. teleserie* - NOOO! YA NO PUEDO MÁS!

Salva nuestro planeta! Sálvalo! Procura leer mis escrituras en todo su detalle! Una pequeña pérdida de información cambia el significado por completo! *comple... compil... compilar!*

Lord Enzo Sezentti IX



Luego de leer esta carta, quedas atónito. Decides aventurarte hacia las profundidades de la caverna. OH POR DIOS! ES CIERTO! Las escrituras de Enzo son maravillosas! Podrás comunicarle al resto del mundo la verdad... WAIT! SE HAN BORRADO PARTE DE LAS ESCRITURAS!



El fragmento más completo de las escrituras

3. Problema

Deberás elaborar un programa capaz de reconstruir las misteriosas escrituras de Lord Enzo, las cuales están representadas en una serie de puzzles que tendrás que completar con piezas que obedecen determinadas reglas. Será necesario idear la manera de que tu programa sea lo más eficiente posible, es decir, que encuentre las soluciones a los puzzles en un tiempo mínimo.

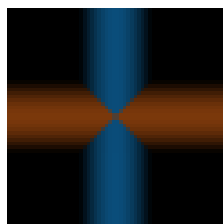
4. Especificaciones

Desde ahora en adelante las recónditas y misteriosas escrituras serán tratadas como si fuese un puzzle, compuesto por un conjunto de piezas que llamaremos Tiles. A continuación se presentan las características que deberás tomar en cuenta para desarrollar tu programa en base a estos elementos:

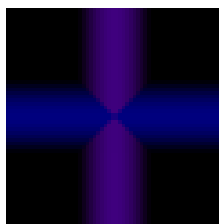
4.1. Tile

Los Tiles corresponden a las piezas que conforman un puzzle. Estas son básicamente una zona cuadrada atravesada por dos hilos de colores, uniendo los puntos medios de sus lados. Cada lado tiene un solo punto medio, por lo que solo puede tener un color asociado. Debido a esto, podemos representar los Tiles refiriendonos sólo a los colores asociados a cada uno de sus lados. Por ejemplo, si tenemos un Tile que es cruzado horizontalmente por un hilo rojo, y verticalmente por un hilo azul, podríamos decir que los lados izquierdo y derecho son rojos, y los lados superior e inferior son azules. Si escribimos los valores de Derecho, Superior, Izquierdo, Inferior de este tile, tenemos Rojo, Azul, Rojo, Azul, o para abreviar, **RARA**

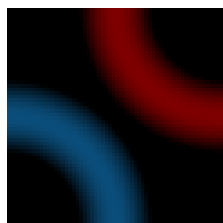
A continuación, algunos ejemplos siguiendo el mismo esquema.



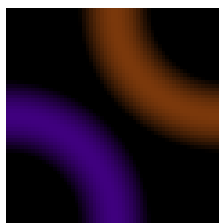
NCNC



AMAM



RRCC



NNMM

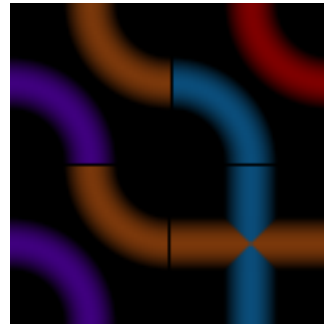
4.2. Puzzle

El puzzle consiste en un grilla rectangular que contiene algunos Tiles ya puestos en algunas celdas, junto con un set de Tiles que deberá ser usado para restaurarlo.

Usted deberá posicionar en cada celda uno de estos tiles para llenar la grilla. Las reglas son simples: por cada tile en una posición dada, los colores de sus lados deben calzar con los colores de cada uno de sus vecinos. No se considerará correcto llenar un Puzzle de Tiles sin tener en cuenta los colores de las piezas adyacentes:

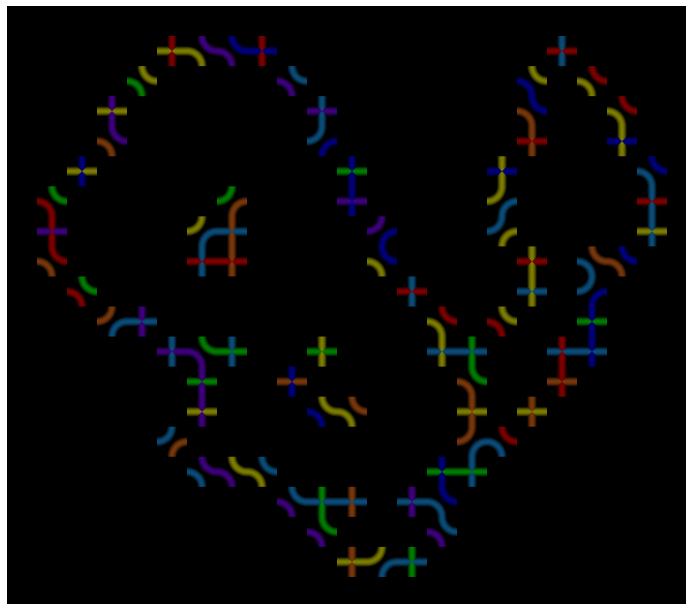


¡Muy bien hecho!



¡INCORRECTO!

A continuación un ejemplo de un puzzle con siete colores:



¿QUIÉN ES ESE POKÉMON?!

5. LibreríaT3

Se ha elaborado una librería que permite la completa interacción con los puzzles, los Tiles y la generación de imágenes. Esta contiene las clases Puzzle y Tile, junto con tres interfaces que usted deberá implementar en sus estructuras de datos. A continuación se presentan los métodos y campos de cada interfaz:

- **ILista<T>**: Representa una lista de un tipo genérico T.
 - **T this[int index]**: Permite acceso al elemento i-ésimo de la lista mediante lo siguiente:
 - **get**: Entrega el elemento i-ésimo de la lista.
 - **set**: Asigna el elemento i-ésimo de la lista.
 - **int Count**: Indica cuantos elementos contiene esta lista
 - **void Add(T item)**: Agrega *item* al final de la lista
 - **bool Contains(T item)**: Retorna *true* si la lista contiene a *item*
 - **bool Remove(T item)**: Elimina el primer elemento igual a *item*. Retorna *true* si se consiguió eliminar.
- **IQueue<T>**: Representa una cola de un tipo genérico T.
 - **int Count**: Indica cuantos elementos hay en cola.
 - **void Enqueue(T item)**: Añade *item* al final de la cola.
 - **T Dequeue()**: Entrega y elimina el primer elemento de la cola.
- **IStack<T>**: Representa una pila de un tipo genérico T.
 - **int Count**: Indica cuantos elementos hay apilados.
 - **void Push(T item)**: Pone *item* encima de la pila.
 - **T Pop()**: Entrega y elimina el elemento de más arriba de la pila.

Las clases Puzzle y Tile se encuentran completamente documentadas. Para ver sus métodos, asegúrese de posicionar el XML de documentación¹ en la misma carpeta que la librería. Cuando Visual Studio le muestre los métodos y propiedades de cada objeto, junto a ellos aparecerá su descripción y como usarlos.

Los puzzles y sus tiles correspondientes estarán almacenados en los archivos con extensión .doge² que les fueron entregados. Estos corresponden a instancias serializadas de la clase Puzzle, las cuales usted deberá des-serializar para poder cargarlas en su programa.

Una vez creada la instancia de Puzzle, utilizando los métodos **GetTiles** y **GetFixed** podrá obtener el set de tiles del puzzle, y sus tiles prefijados, respectivamente. Con esta información deberá buscar el resto de tiles necesarios para resolver el puzzle. Encontrada la solución tendrás que entregarle al método **Print** la lista de todos los Tiles que conforman el puzzle resuelto. Hecho esto, se creará una imagen de la misteriosa escritura de Lord Enzo.

¹El que se llama igual que la librería

²Wow, such name, so mystery

6. Bonus

6.1. Do you have what it takes? (10 %)

Las historias cuentan que en los tiempos de las tablas Fadić existían unos legendarios sabios que manejaban las olvidadas artes de una misteriosa lengua llamada EHF-CHARP. Según las escrituras de Enzo, estos no eran solo una leyenda... Demuestra a tus ayudantes qué tan Jedi puedes ser escribiendo tu algoritmo en F#. Es importante notar que esto aplica SOLO para el algoritmo de solución, tus clases y estructuras de datos deben ser creadas en C#. No cumplir con esto afectará gravemente tu nota.

6.2. Show them who's boss (10 décimas máximo)

Las diez mejores tareas, seleccionadas según cuánto tarden en resolver los puzzles, serán premiadas con décimas extras: 10 para el primer lugar, 9 para el segundo, 8 para el tercero, y así. Adicionalmente, al primer lugar se le entregará un premio sorpresa.

7. Restricciones y alcances

- Tu programa debe ser desarrollado en el lenguaje C#, sobre .Net Framework 4.5.
- El entorno de desarrollo a utilizar debe ser Microsoft Visual Studio 2013.
- El ayudante puede bonificar o castigar tu puntaje³ de la tarea, si le parece adecuado. Se recomienda ordenar el código.
- No subas las carpetas bin, obj ni el archivo .suo⁴, hay un descuento de un punto en tu nota por hacerlo.
- Junto con tu tarea deberás subir un archivo ReadMe donde irá especificado todo lo que tu programa hace o no, junto con las cosas que hayas asumido. En caso de no subir este archivo se aplicará un descuento en tu nota.
- Queda estrictamente prohibido el uso de cualquier librería externa, sin contar LibreriaT3, que trabaje con estructuras de datos; usted deberá implementar sus propias estructuras. Esto incluye las estructuras de datos que vienen con .NET. Los arreglos no califican como estructura de datos.

8. Entrega

- **Fecha/hora:** 12 de octubre de 2014 / 23:59.
- **Lugar:** En repositorios SVN personales, al interior del directorio Tarea 3.

Debe subir la solución de Visual Studio completa en código, sin ejecutables. Tareas que no compilen o no cumplan con las restricciones señaladas en este enunciado tendrán la calificación mínima (1.0).

³ ± 5 décimas.

⁴Es un archivo oculto.