# http://vectorblog.org/wp-content/uploads/2013/06/Telehealth-cartoon-e1370345782337.jpgProyecto Final *Combinación de estructuras*

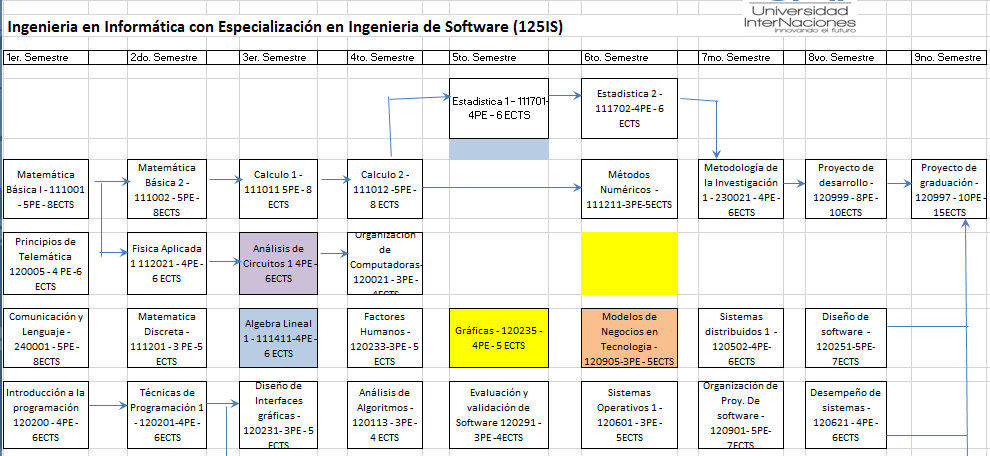
Objetivos del proyecto

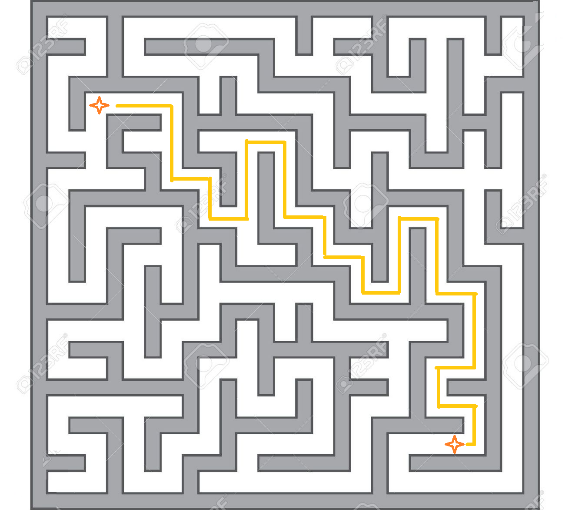
* Utilizar estructuras de datos para brindar una solución para administrar información.

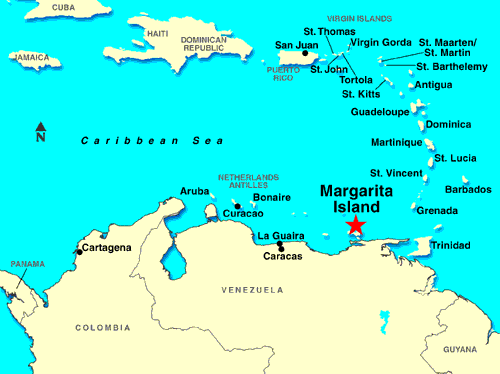
Recomendación: trabajar en grupos de dos integrantes.

Elegir uno de los siguientes proyectos:

1. Las carreras que ofrece una universidad tienen un plan de estudios que comprende varios cursos distribuidos en semestres. Estos cursos están interrelacionados de manera que unos cursos son prerrequisitos de otros. Por ejemplo, para inscribirse en Matemática básica 2, es necesario haber aprobado Matemática básica 1 en el semestre anterior. Desarrollar una aplicación de software que permita ingresar los cursos de los planes de estudio de varias carreras, especificando los prerrequisitos y genere una malla curricular similar a la que se muestra en la siguiente ilustración.



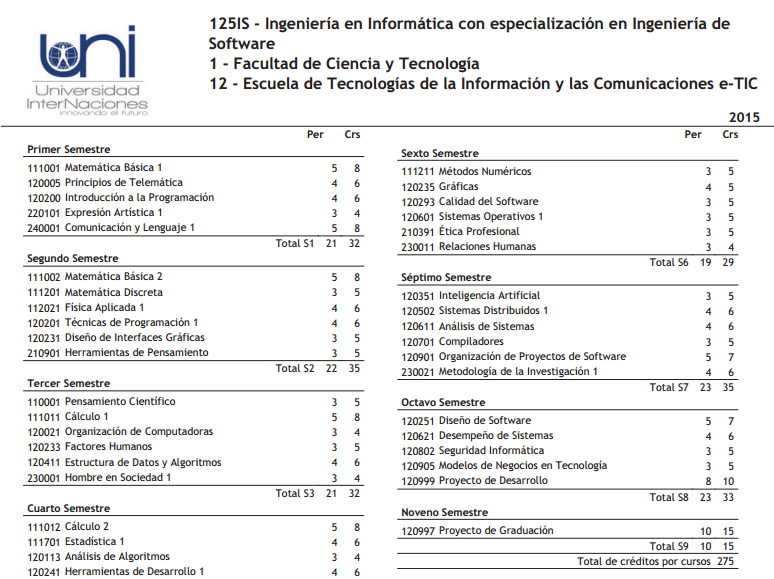
1. Encontrar la salida de un laberinto. El laberinto consta de espacios transitables y paredes que bloquean el paso. Desarrollar una aplicación de software que muestre un laberinto en pantalla, generado a partir de una carga de archivo; que permita al usuario seleccionar dos puntos en cualquier parte del laberinto y luego le muestre un camino que conecte ambos puntos.
2. En el mar Caribe hay varias islas, las cuales se pueden ir visitando en un velero. Pero por la distancia de las islas es necesario hacer paradas y reabastecer alimentos para la tripulación. Desarrolle un juego que permita partir de una isla e ir navegando hacia otras, considerando esta restricción de reabastecimiento de alimentos y la distancia entre cada isla. El objetivo del juego es recorrer todas las islas.



A continuación se muestran más detalles de cada alternativa.

## Opción 1. Malla curricular

Recomendación: descargar el pensum de su carrera en Universidad InterNaciones y analizar qué información guardará por cada curso.



**Fuente**: Pensum de Ingeniería en Informática con especialización en Ingeniería de software, de Universidad InterNaciones, 2017. Disponible en: <http://uni.edu.gt/wp-content/uploads/2017/05/125IS-IngInformaticaEspSoftware-Pensum1.pdf>

Preguntas para analizar:

1. Por el momento enfoquémonos en una única carrera. ¿Qué información guardaría su programa sobre cada curso?
2. Además de esta información, ¿qué necesita para saber la información sobre los prerrequisitos?
3. Ahora enfocándonos en la parte gráfica. ¿Qué datos adicionales le podrían ayudar a dibujar la malla curricular a partir de los datos del pensum y prerrequisitos correspondientes?
4. ¿Qué estructura(s) de datos le serán útiles para desarrollar esta aplicación? ¿De qué manera la(s) utilizará?
5. Si se desea guardar los pensum de varias carreras, ¿qué cambio le tendría qué hacer al programa?
6. ¿Qué pantallas propone para esta aplicación? Colocar prototipos y las relaciones entre dichos prototipos (a partir de qué pantalla se llega a otra pantalla, si es que son varias pantallas).

## Opción 2. Encontrando caminos en un laberinto

Recomendación: Comenzar por un laberinto simple pequeño, que únicamente conste de espacios transitables y paredes.

Preguntas para analizar:

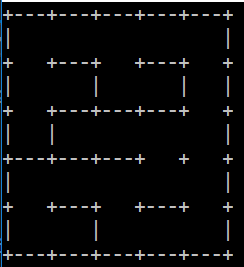
1. ¿Qué formato propone para el archivo que definirá el laberinto?

* Quisiéramos incluir la posibilidad de generar un laberinto aleatorio y encontrar el camino en ese laberinto. Si vemos que es muy complicado entonces guardaremos en un archivo de texto un código que nos indique que clase de celda es cada celda del laberinto.

1. Considerando que el programa debe encontrar el camino dentro del laberinto, ¿qué estructura(s) de datos considera que le será útil para desarrollar esta aplicación? ¿Cómo le será útil dicha(s) estructura(s)?

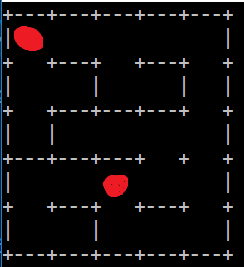
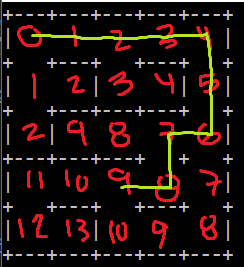
* Hemos considerado 2 posibilidades, una es utilizar arreglos y la otra es utilizar un grafo.
* Esta estructura nos ayudará a encontrar el camino entre 2 puntos del laberinto ya que podremos "navegar" por esta estructura hasta encontrar el punto destino.

1. Coloque un ejemplo de un laberinto (de preferencia simple) y la estructura que lo representará internamente en el programa.



* Este laberinto está representado por un arreglo bidimensional de objetos Celda, los cuales tienen referencias a sus celdas vecinas.

1. Identifique el camino para ir de un punto a otro (de ejemplo) y señale el camino en la estructura.



Para encontrar el camino entre dos puntos utilizaríamos el algoritmo de Dijkstra, el cual además nos da el camino más corto entre 2 puntos. Como se muestra en la imagen, el algoritmo numera las celdas de acuerdo a su distancia del punto de partida. Luego es cuestión de movernos del punto de destino hacia otra celda, siempre escogiendo la de menor distancia, y formaremos el camino.

1. ¿Qué algoritmo utilizará para encontrar (de manera automática) dicho camino?

* El algoritmo del Maestro Dijkstra.

1. Conociendo el camino, ¿cómo lo mostrará visualmente en el programa?

* Utilizaremos Unity, ya que tiene un sin fin de funciones gráficas que nos ayudarán a mostrar el camino. Una de estas funciones se llama LineRenderer, la cual genera una línea entre los puntos que le demos, así que será cuestión de ir generando una nueva línea entre cada dos celdas que vayamos recorriendo.

## Opción 3. Recorrido de islas