



**INGENIERÍA** | **DEPARTAMENTO**  
**UNIVERSIDAD DE LOS ANDES** | **DE COMPUTACIÓN**  
**MÉRIDA VENEZUELA** | **ESCUELA DE SISTEMAS**

**Primera Entrega del Proyecto: "Dots and  
Boxes" .**

Nombre de equipo: "Codemakers"

7 de octubre de 2016

## 1. Nombre del equipo

"Codemakers"

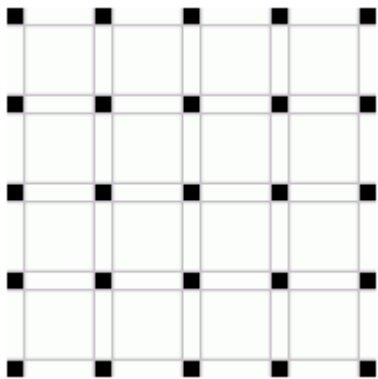
## 2. Integrantes y correos electrónicos

- Javier Mérida Morales. *Email:* [javiermerida05@gmail.com](mailto:javiermerida05@gmail.com)  
*C.I:* V 24.162.551
- Jesús Daniel Rojas. *Email:* [Jesusdanielru1@gmail.com](mailto:Jesusdanielru1@gmail.com)  
*C.I:* V 25.806.551
- Juan Diego Morón. *Email:* [juandiegp17@gmail.com](mailto:juandiegp17@gmail.com)  
*C.I:* V 23.390.971
- Marco Alejandro Sandoval. *Email:* [kada9001@gmail.com](mailto:kada9001@gmail.com)  
*C.I:* V 24.350.394

### 3. ¿Qué se piensa hacer?

Se plantea el desarrollo de un videojuego llamado "Dots and Boxes" ("*Puntos y cajas*"), basado en un popular juego recreativo a lápiz y papel del mismo nombre; creado por el matemático francés François Édouard Anatole Lucas. El juego a desarrollar será para ser jugado por un jugador contra el computador (*player vs computer*).

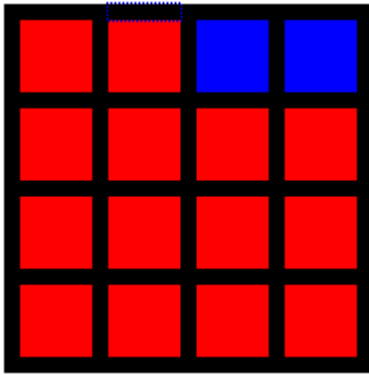
La esencia del juego consiste en dar una matriz de cuadrados con bordes transparentes, a modo de tablero, en donde los vertices de los cuadrados representan puntos que pueden ser unidos.



Ejemplo de matriz de cuadrados de dimensiones 4 x 4.

Los jugadores (en nuestro caso dos: *player vs computer*), tienen jugadas por turnos, donde una jugada consiste en unir mediante líneas verticales u horizontales dos puntos adyacentes en el tablero. Un jugador que completa el cuarto lado de un cuadrado gana cierta cantidad puntos y le corresponde una nueva tirada. El juego termina cuando no hay más líneas que se pueden colocar y todos los cuadrados están llenos por los jugadores.

El ganador del juego es el jugador con más puntos.



Ejemplo de una partida donde el jugador que marca azul pierde contra el que marca rojo (14 pts rojo contra 2 puntos azul).

**El objetivo principal del proyecto es el desarrollo de un oponente (*bot*), capaz de derrotar y dificultar las jugadas del jugador humano.**

Resumiendo las reglas del videojuego:

1. Toque en medio de dos puntos para marcar una línea (vertical u horizontal). Su objetivo es completar una caja de cuatro lados.
2. El jugador que completa una caja de cuatro lados tiene la oportunidad de jugar una vez más.
3. Completar tantas casillas como sea posible.
4. El jugador que complete mayor cantidad cuadrados, gana el juego.

## 4. ¿Cómo se justifica este proyecto en la asignatura?

Según Kenneth Rosen\*, la estructura de datos Árbol, se puede utilizar para analizar distintos tipos de juegos, como las damas o el ajedrez, debido a que cada jugador conoce los movimientos del otro, y por lo tanto no hay componentes de azar en el juego.

En nuestro caso, en "Dots and boxes", usaremos árboles para implementar el videojuego, dado que el *bot*, es decir, el contrincante del jugador humano, debe buscar el camino más eficiente para no perder, o en otras palabras, el camino más eficiente para cerrar un cuadro. Por otra parte, deberá "engañar" al usuario para que este deje libre un solo lado por marcar de un cuadro, cosa que hará el *bot* en el siguiente turno. Esta búsqueda de caminos hará uso de la estructura de datos referida arriba, un árbol, ya que, partiendo de una jugada (*raíz*), se conocen todas las posibles próximas jugadas (*hijos*).

El proyecto sirve como una herramienta para asentar los conocimientos impartidos en el aula de clase, y cuando de asentar conocimientos se trata, la mejor y más efectiva forma de lograrlo es a través de la práctica. Y si la práctica se hace didáctica, donde se aprende, practica y estudia, disfrutando del proceso; ¡mucho mejor!.

---

\* Kenneth H. Rosen, *Matemática Discreta y sus aplicaciones*. Quinta edición. McGraw-Hill. 2004. Colombia.

## 5. ¿Cómo se piensa resolver "Dots and boxes"?

Debido a la temática del juego y a la forma en la cual será programado, este proporciona un usuario virtual (*bot*), inteligente en cierta forma, el cual tiene como fin ser un buen contrante. Naturalmente, tratará de ganar sobre el jugador. En consecuencia, el *bot* debe evaluar la cantidad de combinaciones y jugadas posibles a realizarse. Es por esto que el conjunto de posibilidades será estructurado en árboles, específicamente en árboles estructurados con listas. Se elige esta estructura de datos, debido a que para simular cualquier tipo de agrupación, ya sean variaciones, permutaciones o combinaciones, los árboles son realmente útiles.

El árbol, vendrá dado por la raíz, la cual contendrá la jugada actual que se acaba de dar en el último turno, y cada uno de los hijos de esta raíz, contendrán cada posible jugada a tomar por el *bot*. De esta forma, seleccionará la mejor jugada.

La dificultad que el "usuario virtual" le proporcionará al juego, dependerá de la cantidad de sub-jugadas que el mismo tendrá la capacidad de analizar.

Las herramientas a utilizar en el desarrollo:

- Material de la asignatura: Libro *Tejiendo Algoritmos*, apuntes de clase.
- Lenguaje de programación C++.
- Librería Qt para el manejo de interfaz gráfica.

