

Investigación de Operaciones

II Semestre de 2020

Proyecto 04: Series Deportivas

Estudiantes:

- **Michelle Alvarado Zúñiga – 2017163102**
- **Luis Molina Juárez - 2015047814**

 **Dynamic
Programming**

Tabla de Contenidos

1.	Descripción del problema.....	3
1.1	Descripción general	3
1.2	Características del programa.....	3
2.	Diseño del programa	3
2.1	Decisiones de diseño de la estructura del código	3
2.2	Diseño del archivo de carga.....	4
2.3	Diseño de interfaz gráfica	4
2.4	Algoritmos usados	Error! Bookmark not defined.
3.	Componentes externos	5
4.	Manual de Usuario	5
4.1	Compilación y ejecución	5
4.2	Uso.....	5
5.	Objetivos alcanzados	6
6.	Link del repositorio	6

1. Descripción del problema

1.1 Descripción general

Este proyecto es una aplicación de interfaz gráfica creada en Angular6+ utilizando HTML5 Y CSS3, el cual corresponde a una nueva funcionalidad del Proyecto 00 donde la opción del algoritmo Series Deportivas estará disponible.

1.2 Características del programa

El programa deberá permitir resolver el problema de Series Deportivas, siendo la interfaz gráfica lo más flexible posible permitiéndole al usuario proporcionar los siguientes datos:

- Número máximo de juegos (n), se debe suponer que $n \leq 11$
- Probabilidad del equipo **A** de ganar en casa (p_h), en dónde la probabilidad del equipo **B** de ganar de visita es $q_r = 1 - p_h$
- Probabilidad del equipo **A** de ganar de visita (p_r), en dónde la probabilidad del equipo **B** de ganar de local es $q_h = 1 - p_r$

2. Diseño del programa

2.1 Decisiones de diseño de la estructura del código

Por requisito ya establecido, la ejecución de un algoritmo se debe hacer por medio de un componente de angular. Sin embargo, para separar el control de la interfaz gráfica y el algoritmo correspondiente como tal, se decide que el componente de angular será el encargado de controlar la interfaz gráfica y todos los eventos que esta genere y, además, usará los servicios de una clase (específica por algoritmo) que le dará los resultados correspondientes del algoritmo en ejecución. Esto con el fin de separar responsabilidades y no tener código en el componente de Angular que maneje la interfaz y ejecute la lógica del algoritmo.

2.2 Diseño del archivo de carga

El equipo de trabajo decide que usará archivos JSON para guardar y cargar los datos del usuario. La razón es que typescript, el lenguaje detrás del framework de Angular, es muy flexible con este tipo de archivos y tiene funciones muy útiles para leer y crear en este formato. Más específicamente se decide que en el archivo se guardará:

- El máximo número de juegos.
- La probabilidad p_h .
- La probabilidad p_r .
- Una lista de booleanos correspondiente al formato de juego en donde cada elemento indica si el equipo A juega o no en casa.

```
{
  "maxNumberOfGames": 7,
  "ph": 57,
  "pr": 49,
  "format": [
    {"home": true},
    {"home": true},
    {"home": false},
    {"home": false},
    {"home": false},
    {"home": true},
    {"home": true}
  ]
}
```

2.3 Diseño de interfaz gráfica

Para la interfaz gráfica que se mostrará para la ejecución del algoritmo se decide dividir la pantalla en dos secciones, la sección izquierda permitirá configurar los parámetros de este. La sección derecha permitirá visualizar la tabla donde se visualiza la ejecución del algoritmo. Todas las decisiones de diseño del Proyecto 00 se toman en cuenta en este programa.

3. Componentes externos

Para la creación de algunos componentes del programa se han usado componentes externos a nuestro desarrollo. Dentro de estos están incluidos:

- **plus minus number picker**: este componente se tomó como base y fue modificado para seleccionar la capacidad de la mochila y la cantidad de objetos. Creado por Toby, un usuario de CodePen que puso a disponibilidad el componente en: <https://codepen.io/tobyj/pen/vGbpPj>
- **Bootstrap**: esta librería se incluyó por su clase de estilo *modal*, principalmente para mostrar un mensaje de confirmación para cuando el usuario quiera navegar a otro algoritmo o al menú principal, en caso de que.
- **Flaticon**: se usaron algunos íconos proporcionados por Flaticon, los créditos se encuentran en la página principal del programa.

4. Manual de Usuario

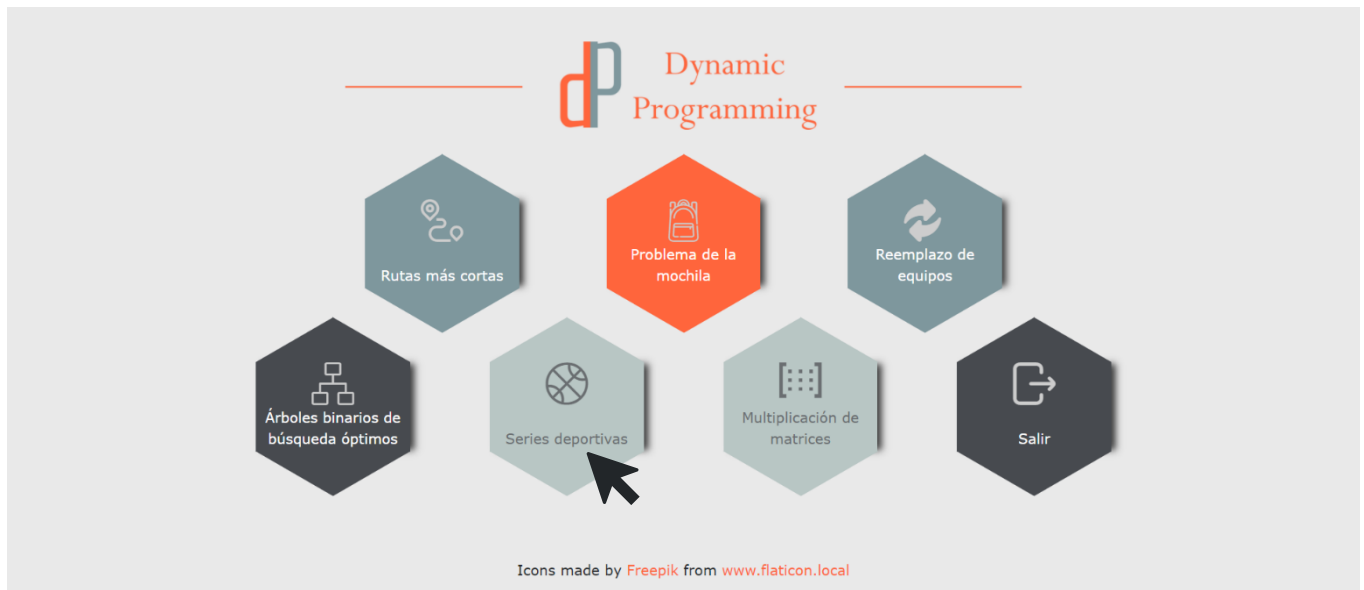
4.1 Compilación y ejecución

Al ser un proyecto en angular se necesita realizar una serie de pasos para poder compilar y ejecutar el programa. Se debe:

1. Abrir la carpeta que contiene el proyecto de Angular, en este caso se llama IC6400-PRY y abrir un comando de línea de Windows en este directorio.
2. Ejecutar el comando *npm install*. Este comando instalará todo lo necesario para correr el proyecto. Como requisito, se debe tener instalado *node.js* (por el comando *npm*) y Angular CLI.
3. Una vez terminado el proceso que ejecuta el comando, se debe ejecutar *ng serve* para poder lanzar el servidor.
4. Se debe acceder desde un navegador a la dirección: *localhost:4200*.

4.2 Uso

El uso de este programa es muy sencillo, se le mostrará el menú principal y será posible seleccionar cualquiera de las opciones. En este caso seleccionará *Reemplazo de equipos*.



Se le mostrará una interfaz que le permitirá:

The screenshot shows the 'Series deportivas' interface. On the left, there are input fields for 'n:' (set to 7), 'ph:' (57%), 'pr:' (49%), 'qr:' (43%), and 'qh:' (51%). Below these are toggle switches for 'Configuración del formato de juego' for seven games, alternating between 'Casa' and 'Visita'. A red button at the bottom left is labeled 'Ejecutar algoritmo'. On the right, there is a table for 'Equipo B' with columns 0 to 4. The first row is for 'Equipo A' with values 0, 1, 2, 3, 4. The table contains numerical values, with the bottom-right cell (4,4) highlighted in dark blue and containing the value 0.578. Annotations with arrows point to the 'Cargar archivo' and 'Guardar archivo' buttons at the top right, the 'Ejecutar algoritmo' button, and the solution table.

Opciones de guardar y cargar archivos

Modificar los datos del algoritmo.
Ejecutar el algoritmo

Visualizar tabla de solución

		Equipo B				
		0	1	2	3	4
Equipo A	0	-	1	1	1	1
	1	0	0.57	0.8151	0.9057	0.9519
	2	0	0.3249	0.5651	0.732	0.8398
	3	0	0.1592	0.3581	0.5413	0.7114
	4	0	0.078	0.2152	0.4011	0.578

5. Objetivos alcanzados

Se alcanzaron los siguientes objetivos:

- Se cargan y guardan archivos.
- Se logra establecer el número máximo de juegos, la probabilidad p_h y p_r .
- Se calculan las probabilidades q_h y q_r .
- Se logra realizar el trabajo extra opcional 1.
- Se logra visualizar la tabla de ejecución con los valores y colores correctos.

6. Link del repositorio

El proyecto se almacenó en un repositorio de GitHub, disponible en el siguiente enlace:

<https://github.com/LuisMJ1197/IC6400-ProyectoSemestral>