

**ESTRUCTURAS de DATOS y ALGORITMOS – PARCIAL 1**

- 1- **Master chef.** Diseñe un struct *Ingrediente* en el que se ingrese producto, gramos, fecha de elaboración, y sus respectivos precios, para la preparación de un struct *Plato*, que tiene una receta, tiempo (hora-minuto), y un arreglo de 4 ingredientes. Además, el platillo contendrá 2 Condimentos, estructura tipo enum que tiene: Sal, Pimienta, Ajo, Oregano, Perjil, Aceite.
- Precargar dos Ingrediente con *strcpy*: Cebollas, 150gr. \$399.9, Caldos: 38gr. 150\$ el cubo, Morrón 82gr. \$456.
  - Crear dos Ingrediente leído con *getline*: Ej. Hongos, 25gr. \$345 y Pechuga 350gr. \$959.99
  - Crear un platillo a partir de esos ingredientes
  - Mostrar el plato y sus ingredientes.
  - Mostrar el costo final a gastar en dos platillos diferentes.
- 2- **BurgerVille**, un lugar donde los clientes registran pedidos o combos, a medida que ocurren se guardan en un arreglo '*Combito*'.

Se debe:

- Cargar un pedido en forma de cadena: *hamburguesa-gaseosa-tamaño*, con token separación "- ".  
Ejemplo: *BigCheddar-Coca-Grande* o *mcTremenda-Sprite-peque*. Use **strtok** para obtener la primera palabra antes del guion, (el token de hamburguesa).
- Determinar la longitud de la cadena token obtenida **strlen**
- Utilice la función **strcasecmp**, contra otra palabra: *cheddar*, y
  - verificar si está contenida dentro del primer token con **strstr**.
  - Anexar cheddar al token de la cadena. **strcat** Ej.: *BigcheddarCheddar*
- Pedir una vocal, y ver si la hamburguesa (el token) la contiene, avisar sino esta. **strchr**

