

# Purple Place Bakery Automaton

Nicolás Cáceres   Óscar Carreño   David Naranjo

Autómatas y Lenguajes Formales  
Escuela de Ingeniería de Sistemas e Informática  
Universidad Industrial de Santander

2023

# Índice

1 Resumen y Abstract

2 Introducción

3 Propuesta

4 Resultados

5 Conclusiones

# Resumen y Abstract

## Resumen

El proyecto se centra en aplicar un Autómata Finito Determinista (AFD) para calcular el precio total de pasteles basándose en sus características. El AFD se diseñará para evaluar la información del pastel, utilizando la información almacenada en sus estados y determinar el precio mediante la revisión de los estados visitados por el autómata y operaciones aritméticas básicas.

## Abstract

The project applies a Deterministic Finite Automaton (DFA) to calculate the total price of cakes based on their characteristics. The DFA will be designed to assess the cake's information, utilizing the data stored in its states, and determine the price by reviewing the visited states of the automaton and basic arithmetic operations.

# Introducción

La teoría de autómatas y lenguajes formales es un campo multidisciplinar cuya importancia en la actualidad no se puede obviar. En el siguiente proyecto se aplica los temas vistos en el curso para la implementación de una máquina que permita calcular el precio de un pastel que contiene determinadas características.

Dentro de las características a evaluar se tiene: Tamaño, relleno, cobertura, decoración, sabor bizcocho, forma, textura y adicionales.

En las siguientes diapositivas se presenta la propuesta planteada para abordar el problema, junto con los resultados de dicha implementación.

# Objetivos

## Objetivo Específico

Diseñar e implementar un autómata que permita calcular el precio de un pastel dadas las características especificadas por el usuario

## Objetivos Generales

- Identificar las características de cada pastel
- Acceder e interpretar la información almacenada en cada estado del autómata
- Implementar dicho autómata en un entorno interactivo con interfaz para el usuario

# Propuesta

Por medio de una cadena de entrada que describe las características del pastel dado un alfabeto preestablecido, se recorre el autómata visitando el estado que refiere al precio de la susodicha característica. La idea es ir evaluando los componentes del pastel según un orden arbitrario e ir visitando los estados que refieren al precio de cada característica.

Una vez recorrida toda la cadena, se obtiene la lista de estados visitados, se hace un ligero ajuste aritmético y por último se realiza el cálculo del precio final del pastel.

## Definición Formal

$$AFD = (\Sigma, Q, \delta, q_0, F)$$

$$\Sigma = \{-, 1, 2, 3\}$$

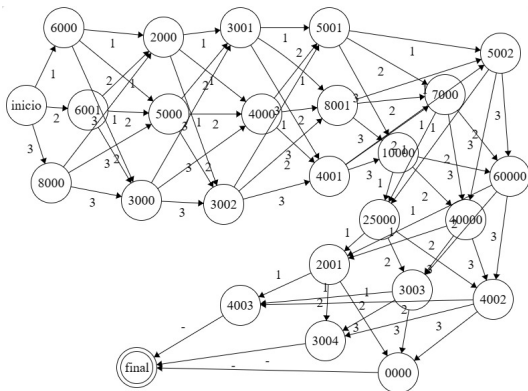
$$Q = \{inicio, 6000, 6001, 8000, 2000, 5000, 3000, 3001, 4000, 3002, \\ 5001, 8001, 4001, 5002, 7000, 10000, 25000, 40000, 60000, 2001, 3003, \\ 4002, 4003, 3004, 0000, Final\}$$

$$q_0 = \{inicio\}$$

$$F = \{Final\}$$

# Función de Transferencia y Diagrama de Transiciones

La función de transferencia también se puede verificar con el siguiente diagrama de transiciones del autómata





# Resultados

Para la prueba se toma una cadena que simboliza un pastel con las siguientes características: Base de vainilla, relleno de crema chantillí, cubierta de mora, arreglo de frutas, cuadrado, pequeño, húmedo y con un adicional de chips de chocolate (12312121).

Para mostrar el funcionamiento, en primera instancia, se muestra el listado de precios, la implementación de dicho autómata, por último, la salida dada la cadena descrita anteriormente.

# Resultados

## PRECIOS:

**TIPO VIZCOCHO:** Vainilla ->1 :6k , Chocolate ->2 : 601k , Tres Leches ->3 :8k

**TIPO RELLENO:** Crema pastelera ->1 :2k, Chantilly ->2 :5k, Arequipe ->3 :3k

**TIPO DE COBERTURA:** Merengue ->1 : 3K, Ganache ->2 :4K , Mora ->3 :3K

**DECORACION:** Frutas ->1 : 5k , Flores ->2 : 8k , Glaseado Colorido ->3 :4K

**FORMA PASTEL:** Redonda ->1 : 5k , Cuadrada ->2 : 7K , Rectangular ->3 : 10k

**TAMAÑO PASTEL:** Pequeño (4 a 6) ->1 : 25k , Mediano (8 a 10) ->2 : 40k, Familiar (15 a 20) ->3 :60k

**TEXTURA DEL PASTEL:** Seco ->1 :2k , Humedo ->2 :3k, Crema ->3 :4k

**ADICIONALES:** Chips de Chocolate ->1 : 4k, Sprinkles ->2 : 3k, Ninguno ->3 :0k

# Resultados

```
from automata.fa.dfa import DFA
dfaPrecto = DFA(
    states={'Inicio', '6000', '6001', '6000', '2000', '5000', '7000', '3001', '4001', '3002', '5001', '0001', '4001', '5001', '7000', '10000', '25000', '40000', '60000', '2001', '3003', '4002', '4003', '3004', '0000', 'Final'},
    input_symbols={'1', '2', '3'},
    transitions={
        'Inicio': [{'input': 'Inicio', '1': '6000', '2': '6001', '3': '0000'},
        '6000': [{'input': '6000', '1': '2000', '2': '5000', '3': '3000'},
        '6001': [{'input': '6001', '1': '2000', '2': '5000', '3': '3000'},
        '0000': [{'input': '0000', '1': '2000', '2': '5000', '3': '3000'},
        '2000': [{'input': '2000', '1': '3001', '2': '4000', '3': '3002'},
        '5000': [{'input': '5000', '1': '3001', '2': '4000', '3': '3002'},
        '3000': [{'input': '3000', '1': '3001', '2': '4000', '3': '3002'},
        '3001': [{'input': '3001', '1': '5001', '2': '0001', '3': '4001'},
        '4000': [{'input': '4000', '1': '5001', '2': '0001', '3': '4001'},
        '3002': [{'input': '3002', '1': '5001', '2': '0001', '3': '4001'},
        '5001': [{'input': '5001', '1': '5002', '2': '7000', '3': '10000'},
        '0001': [{'input': '0001', '1': '5002', '2': '7000', '3': '10000'},
        '4001': [{'input': '4001', '1': '5002', '2': '7000', '3': '10000'},
        '5002': [{'input': '5002', '1': '25000', '2': '40000', '3': '60000'},
        '7000': [{'input': '7000', '1': '25000', '2': '40000', '3': '60000'},
        '10000': [{'input': '10000', '1': '25000', '2': '40000', '3': '60000'},
        '25000': [{'input': '25000', '1': '2001', '2': '3003', '3': '4002'},
        '40000': [{'input': '40000', '1': '2001', '2': '3003', '3': '4002'},
        '60000': [{'input': '60000', '1': '2001', '2': '3003', '3': '4002'},
        '2001': [{'input': '2001', '1': '4003', '2': '3004', '3': '0000'},
        '3003': [{'input': '3003', '1': '4003', '2': '3004', '3': '0000'},
        '4002': [{'input': '4002', '1': '4003', '2': '3004', '3': '0000'},
        '4003': [{'input': 'Final', '1': '4003', '2': '4003', '3': '4003'},
        '3004': [{'input': 'Final', '1': '3004', '2': '3004', '3': '3004'},
        '0000': [{'input': 'Final', '1': '0000', '2': '0000', '3': '0000'},
        'Final': [{'input': 'Final', '1': 'Final', '2': 'Final', '3': 'Final'}
    ]),
    initial_state='Inicio',
    final_states={'Final'})
```

# Resultados

```
myList=list(dfaPrecio.validate_input("12312121-",step=True))
Sum=0
aux=0
for i in range(len(myList)):
    if myList[i]!='inicio' and myList[i]!='Final':
        Sum=Sum+(round(int(myList[i])/10)*10)
print("El precio del pastel es: {} pesos".format(Sum))
```

El precio del pastel es: 58000 pesos

# Conclusiones

La solución propuesta para abordar el problema propuesto resultó adecuada ya que cumple con los requerimientos planteados desde un principio. A su vez, se aplicó correctamente la teoría vista en el curso, en este caso, la referente a los autómatas finitos deterministas.