

# Algoritmos de Minimización de NFA a DFA: Teoría y Métodos

Exploraremos los algoritmos esenciales para la minimización de NFA a DFA. Analizaremos los métodos teóricos y enfoques prácticos. Descubriremos herramientas y aplicaciones clave en este campo.



# ¿Qué son NFA y DFA? Una breve introducción

## NFA (Autómata Finito No Determinista)

Un NFA permite múltiples transiciones. Puede estar en varios estados a la vez.

## DFA (Autómata Finito Determinista)

Un DFA tiene una única transición. Está en un solo estado en cada momento.

# Subset Construction

Moment Sthingtic  
(LIFA)

NFA

Courtel ust?  
falsatic fintle sutounor  
(XIFA)

Towretrist..1)  
FIFA - Shalim et4  
Felerist - the urnod.op)  
NFA- to et)

NFA aotengating

( DFA)  
nJute Contreruriee

1. Contactt to iFAL)
2. Contelgy (LITA.)
3. Contecty to therig
4. Writt up: (UTAL)
4. Anping the thorfec  
(warife itfa).
- +
7. Encresiord by {ffle
8. Anpleses foot (equi
- 9.
4. Custrorake/ Evel)
4. Contactt sut ony a
5. Contactf featerius
- +
6. Chin sutional Ruton  
(lwarid clorer)

1. Confrectis actio
2. Colote Chileigy

## Algoritmo de Subconjuntos: La base de la conversión

1

### Inicio

Crear el estado inicial del DFA.

2

### Transiciones

Calcular transiciones para cada subconjunto.

3

### Repetir

Añadir nuevos estados al DFA.

4

### Final

Marcar los estados finales del DFA.

# DFA Minimization

D (DFA, :), Li:)))::=c):(ot)1 :-300) +:1235110)

D (DFA, :), Li:)))::=d):b=(s=1000) +:233,36))

D (DFA, ), Li:)))::=d)st)=(1 :2300) +:1235,04)

↓  
minimizations

D (DFA, :), Li:)))::=c):(ot)1 =1200) +:113,41))

D (DFA, :), Li:)))::=d):b=(s=1000) +:232,600)

D (DFA, ), Li:)))::=d)st)=(2=2000) +12325,04)

Rapplis tësh't cod bow ril mominetpons it deaft expeddelt.

## Minimización de DFA: Eliminando estados redundantes

1

### Identificar

Buscar estados  
equivalentes.

2

### Combinar

Unir los estados  
redundantes.

3

### Optimizar

Reducir la complejidad del autómata.



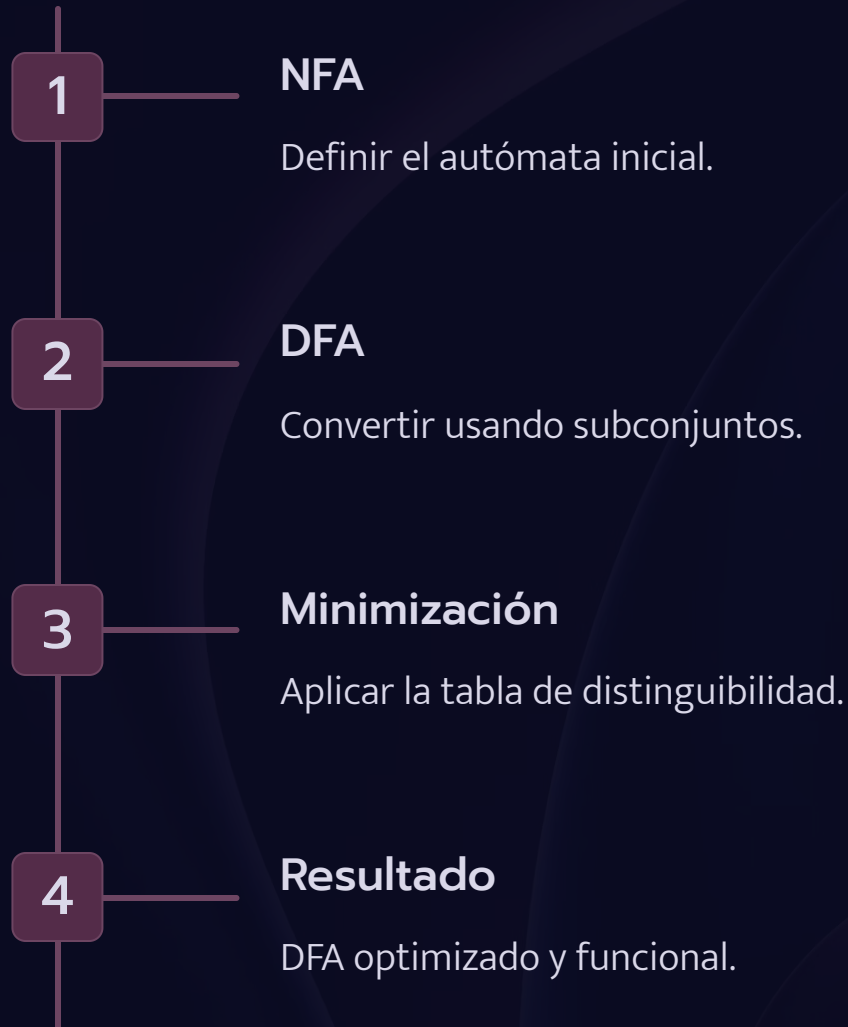
Deterministic Finite Automation (DFA)										
Path	Wsste		State		0	8	1	2	3	
f, a	10	1.4	10	1.7	98	5.9	3/2	521	57	0
f, 2	66	9.2	35	3.5	36	3.3	1/2	683	57	0
f, a	66	1.3	64	1.7	58	3.5	9/2	586	57	0
f, 4	61	3.6	64	3.3	71	6.6	3/3	551	51	0
f, 4	67	1.1	96	1.6	97	4.2	8/5	565	67	4
f, 4	08	1.9	96	1.9	74	9.3	3/2	323	6.2	0
f, a	06	3.7	96	1.4	34	1.5	4/5	333	67	0
f, a	04	3.1	64	3.6	31	3.9	2/3	309	0.7	7
f, 5	02	3.1	98	0.7	33	0.2	3/5	308	07	6
f, a	03	4.9	99	0.2	34	9.6	3/2	989	01	0
f, a	07	3.3	96	3.3	34	9.7	3.1	915	6.3	0
f, a	05	0.9	96	0.2	75	0.2	7 1	903	8.1	0
f, 3	04	6.6	94	4.1	78	9.6	3/4	944	6.9	4
f, a	55	0.1	33	4.5	74	2.1	2/1	630	9.2	4
f, a	24	6.7	98	6.7	47	4.1	8/5	800	3.5	0
f, a	27	6.2	68	3.5	33	8.1	8/6	399	4.4	0
f, a	87	8.3	38	6.7	34	4.1	3/5	500	3.8	0
f, a	64	8.1	95	4.4	42	4 1	2/6	332	6.2	0
f, a	99	5.5	94	3.5	94	5 1	3/5	934	87	0
f, a	76	3.7	32	3.5	33	8.1	3/6	852	6.3	0

# Método de la Tabla de Distinguibilidad: Un enfoque práctico

Estado A	Estado B	¿Distinguibles?
q1	q2	Sí/No
q3	q4	Sí/No

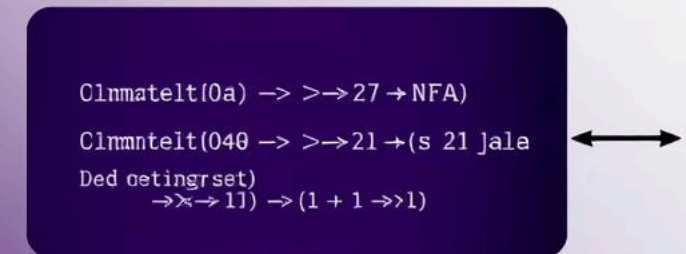
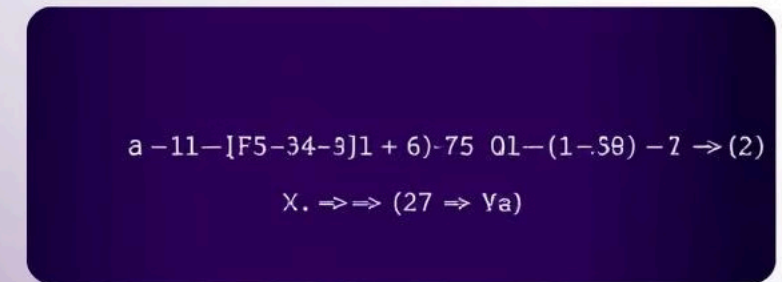
Este método utiliza una tabla. Se identifican los estados distinguibles. Esto facilita la minimización del DFA.

# Ejemplos paso a paso: NFA a DFA minimizado



## NFA to DFA

NFA to DFA





# Herramientas y software para la conversión y minimización



## JFLAP

Software educativo para teoría de autómatas.



## Online Tools

Convertidores y minimizadores en línea.



## Libraries

Implementaciones en lenguajes de programación.



# Conclusión: Importancia y aplicaciones de los algoritmos

## Compilación

Análisis léxico y sintáctico.

## Reconocimiento

Patrones en textos y datos.

## Diseño

Protocolos de comunicación.