

# Diseño de Lenguajes de Programación, Lab. 4

## RESULTADOS

Gabriel Brolo Tobar, carnet: 15105

### I. Comparación entre Expresiones Regulares.

A continuación, se muestran algunas RE (*regular expressions*) ingresadas al programa:

- a)  $ab^*ab^*$
- b)  $0^*10^*$
- c)  $(a^*|b^*)c$
- d)  $b+abc+$

Se procederá a mostrar los resultados obtenidos mediante la **generación directa del DFA** (*gd* de aquí en adelante), así como los obtenidos con la **minimización mediante el método de particiones** (*min* de aquí en adelante), además se mostrarán algunos casos en los que los resultados son distintos.

#### A. $ab^*ab^*$

*min:*

```
REGULAR EXPRESSION: ab*ab*
REGULAR EXPRESSION IN POSTFIX: ab*.a.b*.
SYMBOL LIST: [a, b]
STATES ([PARTITION]=STATE ID): {[1, 3]=1, [2, 4]=2, [0]=0}
TRANSITION TABLE ([STATE]=[SYMBOL]=[STATE])
WHERE STATE CORRESPONDS TO A STATE IN 'STATES':
{0={a=1}, 1={a=2, b=1}, 2={b=2}}
```

```
INITIAL STATE: [0]
```

```
FINAL STATES: [2]
```

*gd:*

```
REGULAR EXPRESSION: ab*ab*
REGULAR EXPRESSION IN POSTFIX: ab*.a.b*.
SYMBOL LIST: [a, b]
POSITIONS (POSITION_ID = SYMBOL): {1=a, 2=b, 3=a, 4=b, 5=#}
STATE MAP (STATE = FOLLOWPOS): {0=[2, 3], 1=[4, 5], 2=[1]}
TRANSITION TABLE: {0={a=1, b=0}, 1={b=1}, 2={a=0}}
```

```
INITIAL STATE: [2]
```

```
FINAL STATES: [1]
```

Ambos DFA contienen el mismo número de estados y las transiciones entre estos son iguales, *min* = *gd*.

#### B. $0^*10^*$

*min:*

```
REGULAR EXPRESSION: 0*10*
REGULAR EXPRESSION IN POSTFIX: 0*1.0*.
SYMBOL LIST: [0, 1]
STATES ([PARTITION]=STATE ID): {[0, 1]=0, [2, 3]=1}
TRANSITION TABLE ([STATE]=[SYMBOL]=[STATE])
WHERE STATE CORRESPONDS TO A STATE IN 'STATES':
{0={0=0, 1=1}, 1={0=1}}
```

```
INITIAL STATE: [0]
```

```
FINAL STATES: [1]
```

*gd:*

REGULAR EXPRESSION:  $0^*10^*$   
 REGULAR EXPRESSION IN POSTFIX:  $0^*1.0^*$ .  
 SYMBOL LIST: [0, 1]  
 POSITIONS (POSITION\_ID = SYMBOL): {1=0, 2=1, 3=0, 4=#}  
 STATE MAP (STATE = FOLLOWPOS): {0=[1, 2], 1=[3, 4]}  
 TRANSITION TABLE: {0={0=0, 1=1}, 1={0=1}}  
 INITIAL STATE: [0]  
 FINAL STATES: [1]

Ambos DFA contienen el mismo número de estados y las transiciones entre estos son iguales,  $min = gd$ .

### C. $(a^*|b^*)c$

*min:*

REGULAR EXPRESSION:  $(a^*|b^*)c$   
 REGULAR EXPRESSION IN POSTFIX:  $a^*b^*|c$ .  
 SYMBOL LIST: [a, b, c]  
 STATES ([PARTITION]=STATE ID): {[1]=1, [2]=2, [3]=3, [0]=0}  
 TRANSITION TABLE ([STATE]=[SYMBOL]=[STATE])  
 WHERE STATE CORRESPONDS TO A STATE IN 'STATES':  
 {0={a=1, b=2, c=3}, 1={a=1, c=3}, 2={b=2, c=3}, 3={}}  
 INITIAL STATE: [0]  
 FINAL STATES: [3]

*gd:*

REGULAR EXPRESSION:  $(a^*|b^*)c$   
 REGULAR EXPRESSION IN POSTFIX:  $a^*b^*|c$ .  
 SYMBOL LIST: [a, b, c]  
 POSITIONS (POSITION\_ID = SYMBOL): {1=a, 2=b, 3=c, 4=#}  
 STATE MAP (STATE = FOLLOWPOS): {0=[1, 2, 3], 1=[4]}  
 TRANSITION TABLE: {0={a=0, b=0, c=1}, 1={}}  
 INITIAL STATE: [0]  
 FINAL STATES: [1]

En esta ocasión,  $gd$  proporciona el DFA mínimo, sin embargo,  $min$  proporciona el mismo DFA generado por la transformación del NFA a DFA.

### D. $b+abc+$

*min:*

REGULAR EXPRESSION:  $b+abc+$   
 REGULAR EXPRESSION IN POSTFIX:  $bb^*.a.b.c.c^*$ .  
 SYMBOL LIST: [a, b, c]  
 STATES ([PARTITION]=STATE ID): {[5, 6]=2, [4]=1, [0, 1, 2, 3]=0}  
 TRANSITION TABLE ([STATE]=[SYMBOL]=[STATE])  
 WHERE STATE CORRESPONDS TO A STATE IN 'STATES':  
 {0={b=0}, 1={c=2}, 2={c=2}}  
 INITIAL STATE: [0]  
 FINAL STATES: [2]

*gd:*

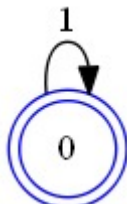
REGULAR EXPRESSION:  $b+abc+$   
 REGULAR EXPRESSION IN POSTFIX:  $bb^*.a.b.c.c^*$ .  
 SYMBOL LIST: [b, a, c]  
 POSITIONS (POSITION\_ID = SYMBOL): {1=b, 2=b, 3=a, 4=b, 5=c, 6=c, 7=#}  
 STATE MAP (STATE = FOLLOWPOS): {0=[2, 3], 1=[4], 2=[5], 3=[6, 7], 4=[1]}  
 TRANSITION TABLE: {0={a=1, b=0}, 1={b=2}, 2={c=3}, 3={c=3}, 4={b=0}}  
 INITIAL STATE: [4]  
 FINAL STATES: [3]

En esta ocasión, min proporciona un DFA erróneo, la implementación del algoritmo no fue eficaz para esta regexp, sin embargo, gd proporciona el DFA correcto y mínimo. La diferencia es que min, tiene una transición menos, resultado que una partición no fue generada correctamente.

## II. Comparación con un compañero

Se comparó la expresión regular:  $((1?)^*)^*$  con el compañero José Custodio.

Los DFA obtenidos fueron los mismos. En la generación proveída por José, el autómata es:



En mi generación directa:

```
REGULAR EXPRESSION: ((1?)*)*
REGULAR EXPRESSION IN POSTFIX: 1ε|**
SYMBOL LIST: [1]
POSITIONS (POSITION_ID = SYMBOL): {1=1, 2=#}
STATE MAP (STATE = FOLLOWPOS): {0=[1, 2]}
TRANSITION TABLE: {0={1=0}}
INITIAL STATE: [0]
FINAL STATES: [0]
```

Y en la minimización:

```
REGULAR EXPRESSION: ((1?)*)*
REGULAR EXPRESSION IN POSTFIX: 1ε|**
SYMBOL LIST: [1]
STATES ([PARTITION]=STATE ID): {[0, 1]=0}
TRANSITION TABLE ([STATE]={ [SYMBOL]=[STATE] })
WHERE STATE CORRESPONDS TO A STATE IN 'STATES':
{0={1=0}}
INITIAL STATE: [0]
FINAL STATES: [0]
```

Los cuales, ambos, producen el mismo autómata mostrado en la imagen de José.