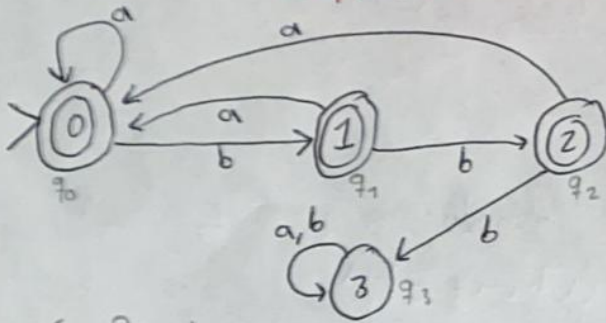


Expresiones Regulares.

①



Expresión Regular:

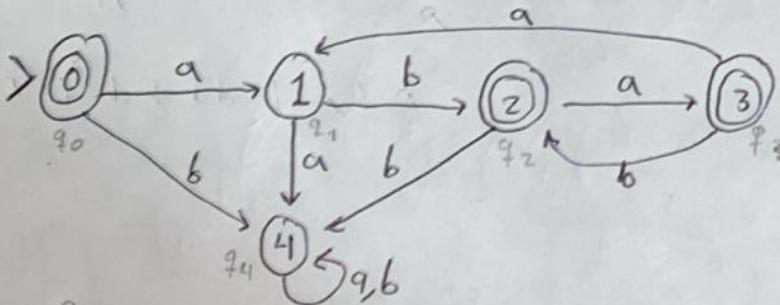
$$\frac{q_0 a q_0}{a^*}$$

$$\frac{q_0 a q_1 y q_0}{b(\lambda + a)}$$

$$\frac{q_0 a q_2 y q_0}{b(b + ba^*)}$$

$$\rightarrow a^* + b(\lambda + a) + b(b + ba^*) //$$

②



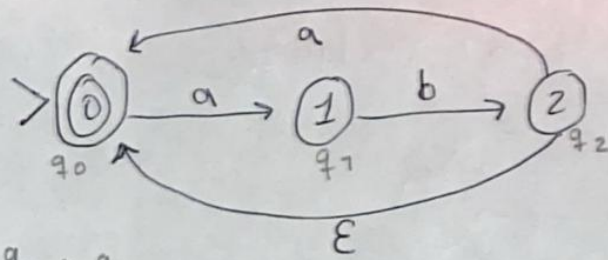
$$\frac{q_0 a q_0}{\lambda}$$

$$\frac{q_0 a q_2}{ab}$$

$$\frac{q_0 a q_3 y q_2}{ab a(\lambda + b)}$$

$$\rightarrow \lambda + ab + ab a(\lambda + b) //$$

③

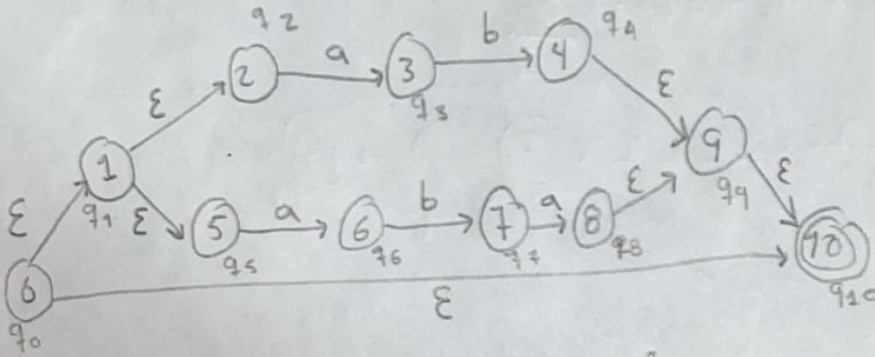


$$\frac{q_0 \alpha q_0}{\lambda}$$

$$\frac{q_0 \alpha q_0 \text{ pasando por } q_1 \text{ y } q_2}{(ab(a+\epsilon))^*}$$

$$\rightarrow \lambda + (ab(a+\epsilon))^*$$

④



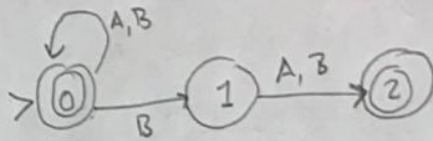
$$\frac{q_0 \alpha q_{10}}{\epsilon}$$

$$\frac{q_0 \alpha q_{10} \text{ pasando por } q_1, 2, 3, 4, 9}{\epsilon \epsilon a b \epsilon \epsilon}$$

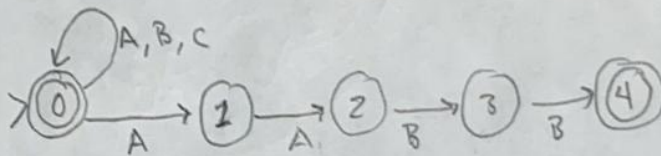
$$\frac{q_0 \alpha q_{10} \text{ pasando por } q_1, 5, 6, 7, 8, 9}{\epsilon \epsilon a b a \epsilon \epsilon}$$

$$\rightarrow \epsilon + \epsilon \epsilon a b (\epsilon \epsilon + a \epsilon \epsilon)$$

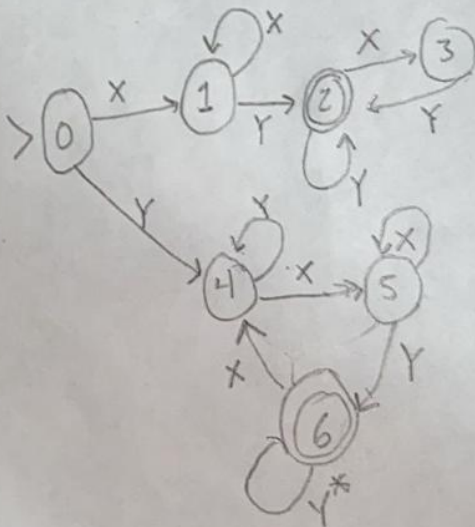
① $(A|B)^*B(A|B)$



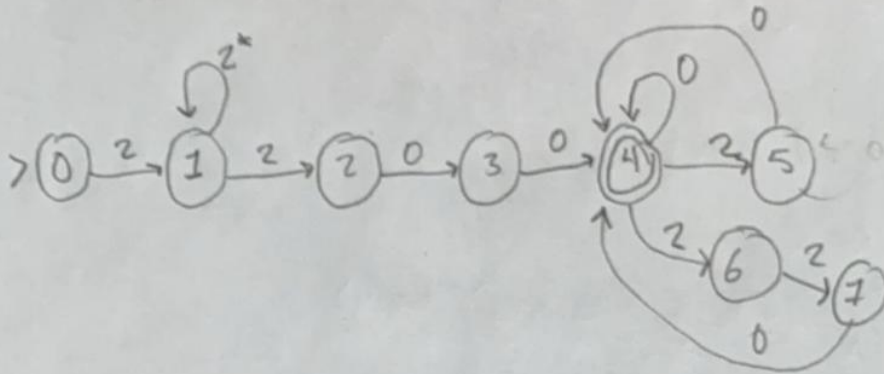
② $(A|B|C)^*AABBB$



③ $XX^*Y(XYY^*)|YY^*(XX^*YY^*)^*$



④ $22^*200(0^*2|220)^*$



Hacer un formulario para ingresar información personal de una persona y debe validar los campos por medio de unas expresiones regulares, obligatoriamente tiene que tener un campo para ingresar un E-mail y lo debe validar, igualmente el nombre no debe contener números, cada campo debe ser validado de forma exitosa por medio de expresiones regulares. Mínimo 5 campos.

The image shows a web application window titled "Expresiones Regulares". It contains a form with five input fields: "Nombres" (containing "Deyvis Saul"), "Apellidos" (containing "Benito Medrano"), "Correo:" (containing "deyvis@"), "Edad:" (containing "20"), and "Telefono:" (containing "4617-2036"). A "Guardar" button is located to the right of the "Telefono:" field. A modal message box is displayed over the "Correo:" field, titled "Message", with an information icon and the text "El correo es invalido". An "OK" button is at the bottom of the message box. Below the form fields is a large, empty rectangular box.

Expresiones Regulares

Nombres: Deyvis Saul

Apellidos: Benito Medrano

Correo: deyvis@

Edad: 20


Telefono: 4617-2036

Guardar

Message

El correo es invalido

OK


— □ ×

Expresiones Regulares

Nombres:

Apellidos:

Correo:

Edad:

Telefono:

Nombres: Deyvis Saul
 Apellidos: Benito Medrano
 Correo: deyvis@gmail.com
 Edad: 20
 Telefono: 4617-2036

Códigos de validaciones:

```
public class Validaciones {
```

```
// Expresión regular para Validaciones
```

```
private static final String caracteresNombre = "[a-zA-ZáéíóúÁÉÍÓÚñÑ\\s-]+$";
```

```
private static final String caracterEmail="[a-zA-Z0-9_%.+]+@[a-zA-Z0-9+\\.[a-zA-Z]{2,}$";
```

```
private static final String caracterEdad="^(?:[0-9]|[1-9][0-9]|1[0-3][0-9])$";
```

```
private static final String caracterTelefono="^[0-9]{4}([\\s-]?[0-9]{4})$";
```

```
public Validaciones() {
```

```
}
```

```
public boolean validarNombre(String nombre){
```

```

Pattern patron= Pattern.compile(caracteresNombre);

for(int i=0; i<nombre.length(); i++){
    char character= nombre.charAt(i);
    String characterString= String.valueOf(character);

    if(!patron.matcher(characterString).matches()){
        return false;
    }
}
return true;
}

```

```

public boolean validarCorreo(String correo){
    Pattern patron= Pattern.compile(caracterEmail);

    for(int i=0; i<correo.length(); i++){
        char character=correo.charAt(i);
        String characterString = String.valueOf(character);

        if(!patron.matcher(correo).matches()){
            return false;
        }

    }
    return true;
}

```

```

public boolean ValidarEdad(String edad){
    Pattern patron=Pattern.compile(caracterEdad);

```

```
    Matcher comparacion= patron.matcher(edad);
```

```
    return comparacion.matches();
```

```
}
```

```
public boolean validarTelefono(String telefono){
```

```
    Pattern patron=Pattern.compile(caracterTelefono);
```

```
    Matcher comparacion= patron.matcher(telefono);
```

```
    return comparacion.matches();
```

```
}
```

```
}
```