

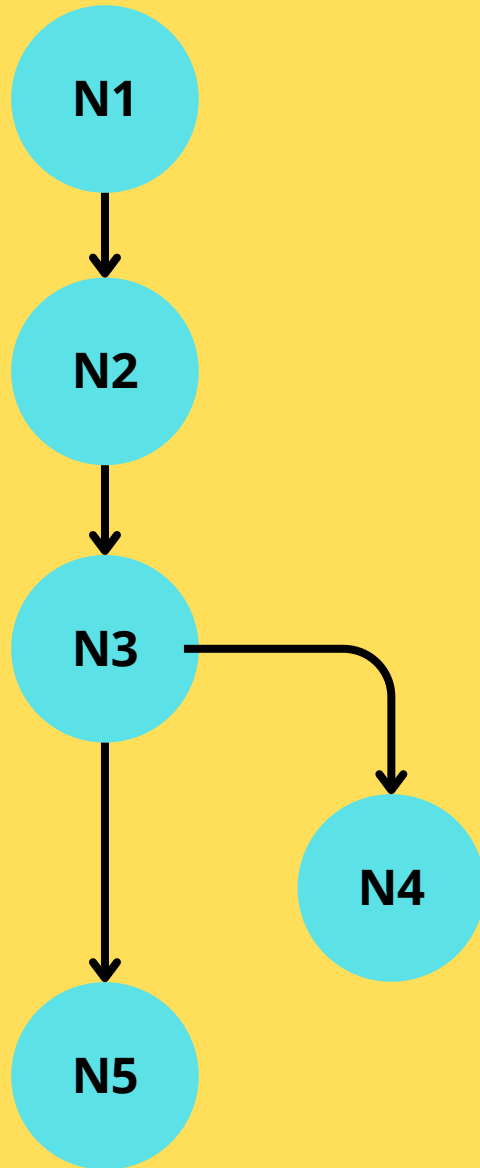
## Método: conectarBD

```
(1) public Connection conectarBD() {  
    (2)  Connection conn = null;  
        (3)  try {  
Class.forName ("com.mysql.DriverManager").newInstance ();  
        String url = "jdbc:mysql://127.0.0.1/test?  
        user=lopes&password=123";  
        conn = DriverManager. getConnection(url) ;  
        }(4)catch (Exception e) { }  
        (5) return conn; }
```

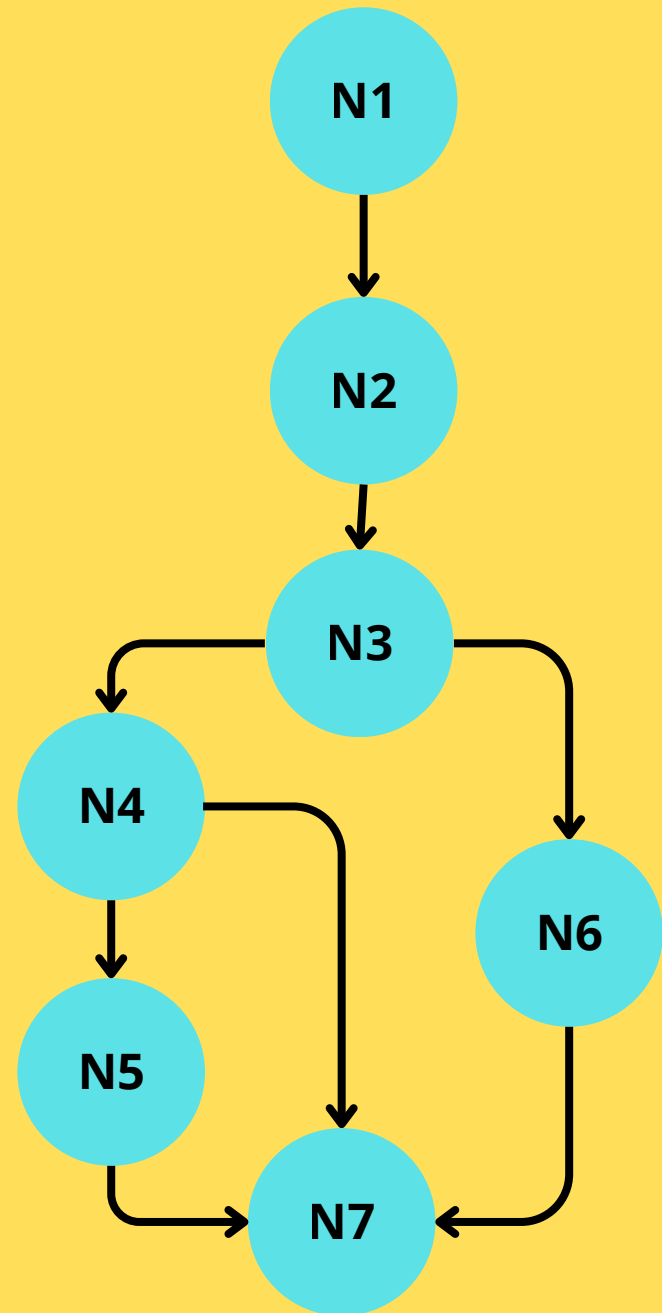
## Método: verificarUsuario

```
(1) public boolean verificarUsuario (String login, String senha) {  
    (2) String sql = "";  
    Connection conn = conectarBD() ;  
    sql += "select nome from usuarios ";  
    sql += "where login = " + "'" + login + "'";  
    sql += " and senha = " + "'" + senha + "'";  
    (3) try {  
        Statement st = conn. createStatement ();  
        ResultSet rs = st.executeQuery (sql) ;  
        (4)if (rs.next () ) {  
            (5) result = true;  
            (5)nome = rs.getString("nome");}  
        } (6) catch (Exception e) { }  
        (7) return result; }  
}
```

## Método: conectarBD



## Método: verificarUsuario



# Complexidade ciclomática

fórmula:

$$M = E - N + 2P$$

Método: conectarBD

Método: verificarUsuario

$$E = 5, N = 5, P = 1$$

$$E = 8, N = 7, P = 1$$

$$M = 5 - 5 + 2 \times 1$$

$$M = 8 - 7 + 2 \times 1$$

$$M = 0 + 2$$

$$M = 1 + 2$$

$$M = 2$$

$$M = 3$$

## Caminhos básicos

### Método: conectarBD

**P1: N1 → N2 → N3 → N5**  
(conexão bem-sucedida, sem exceção)

**P2: N1 → N2 → N3 → N4 → N5**  
(exceção no try, cai no catch e retorna)

### Método: verificarUsuario

**P1: N1 → N2 → N3 → N4(verdadeiro) → N5 → N7**  
(fluxo normal, registro encontrado)

**P2: N1 → N2 → N3 → N4(falso) → N7**  
(fluxo normal, nenhum registro)

**P3: N1 → N2 → N3 → N6 → N7**  
(exceção no try)