|  |
| --- |
|  |
| Prueba de Caja Blanca |
| ***“Plataforma de Contenidos Educativos Learning Content”*** |
|  |
|  |
| **Integrantes:**  Díaz Bautista Adriana Lizbeth  Mascasela Atupaña Brandon Raúl  Paredes Almachi Michelle Estefania  Pila Torres Johanna Jicela  **Fecha 2024-02-16** |

**BUSCAR PROBLEMAS**

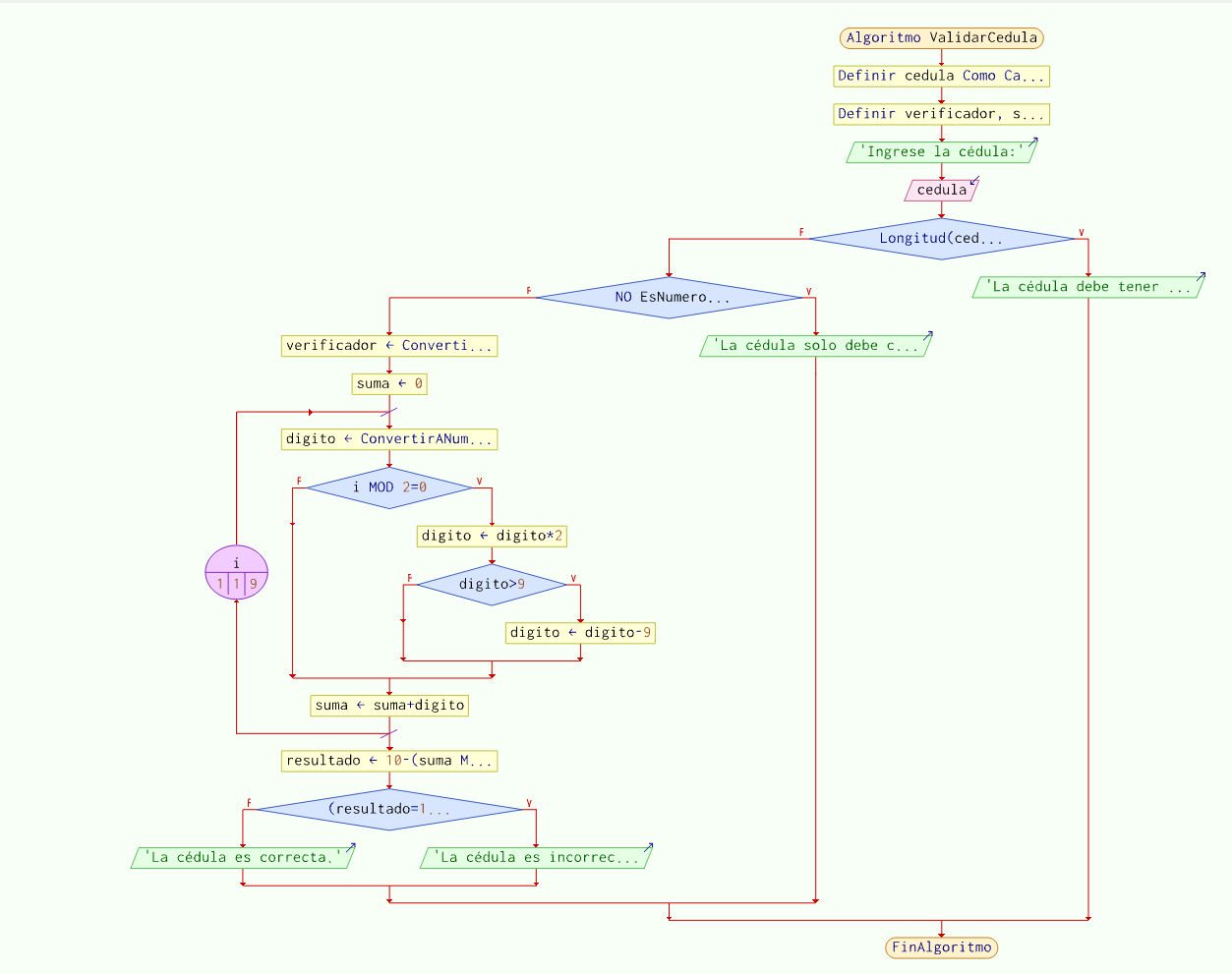
**CÓDIGO FUENTE**

|  |
| --- |
| export const validarCedula = (cedula) => {  if (cedula.length !== 10) {  return false;  } else {  if (!/^\d+$/.test(cedula)) {  return false;  } else {  const verificador = parseInt(cedula.charAt(9));  let suma = 0;  cedula.slice(0, 9).split('').forEach((digito, i) => {  digito = parseInt(digito);  if (i % 2 === 0) {  digito \*= 2;  if (digito > 9) {  digito -= 9;  }  }  suma += digito;  });  const resultado = 10 - (suma % 10);  if ((resultado === 10 && verificador !== 0) || resultado !== verificador) {  return false;  } else {  return true;  }  }  }  }; |

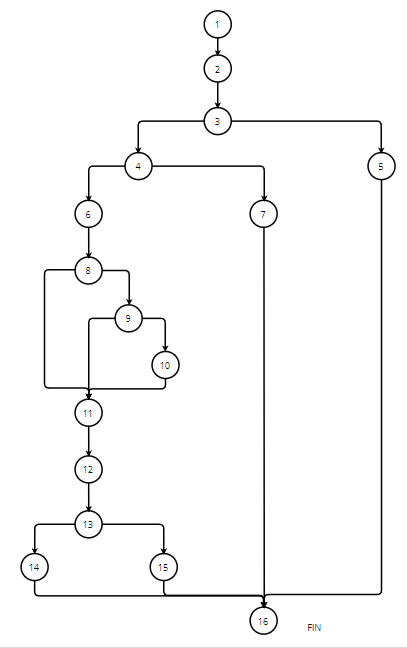
**PSUDOCÓDIGO**

|  |
| --- |
| Algoritmo ValidarCedula  Definir cedula Como Cadena  Definir verificador, suma, digito Como Entero  Escribir 'Ingrese la cédula:'  Leer cedula  Si Longitud(cedula)<>10 Entonces  Escribir 'La cédula debe tener 10 dígitos.'  SiNo  Si NO EsNumero[cedula] Entonces  Escribir 'La cédula solo debe contener números.'  SiNo  verificador <- ConvertirANumero(Subcadena(cedula,10,1))  suma <- 0  Para i<-1 Hasta 9 Con Paso 1 Hacer  digito <- ConvertirANumero(Subcadena(cedula,i,1))  Si i MOD 2=0 Entonces  digito <- digito\*2  Si digito>9 Entonces  digito <- digito-9  FinSi  FinSi  suma <- suma+digito  FinPara  resultado <- 10-(suma MOD 10)  Si (resultado=10 Y verificador<>0) O resultado<>verificador Entonces  Escribir 'La cédula es incorrecta.'  SiNo  Escribir 'La cédula es correcta.'  FinSi  FinSi  FinSi  FinAlgoritmo |

**DIAGRAMA DE FLUJO**



**GRAFO**



**RUTAS**

**R1:** 1, 2, 3, 4, 6, 8, 9, 11, 12, 13, 14, 16

**R2:** 1, 2, 3, 4, 6, 8, 11, 12, 13, 14, 16

**R3:** 1, 2, 3, 4, 6, 8, 11, 12, 13, 15, 16

**R4:** 1, 2, 3, 4, 6, 8, 9, 10, 11, 12, 13, 14, 16

**R5:** 1, 2, 3, 4, 6, 8, 9, 10, 11, 12, 13, 15, 16

**R6:** 1, 2, 3, 4, 7, 16

**R7:** 1, 2, 3, 5, 16

**COMPLEJIDAD CICLOMÁTICA**

Se puede calcular de las siguientes formas:

* V(G) = número de nodos predicados(decisiones)+1

V(G)=5+1=6

* V(G) = A – N + 2

V(G)= 20 – 16 + 2 = 6

DONDE:

**P:** Número de nodos predicado

**A:** Número de aristas

**N:** Número de nodos