

## Ejercicios a realizar con ciclos, Bloque 2, 2020

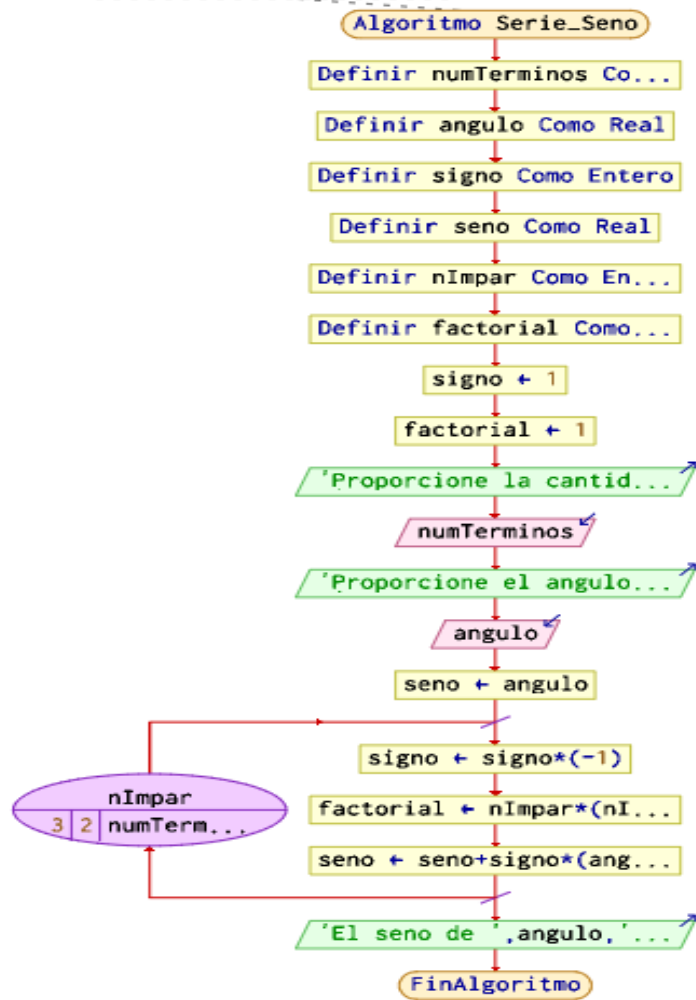
Usa el ciclo que consideres más adecuado salvo en el caso que se pida uno explícitamente. Se entregará tanto pseudocódigo como diagrama de flujo.

1. Escriba un algoritmo para calcular el seno de un ángulo proporcionado en radianes usando la siguiente fórmula. El usuario define cuántos términos se deben calcular, en el siguiente ejemplo se calcularon 3 términos, pero pueden ser más.

a.  $\text{Seno}(x) = x - (x^3/3!) + (x^5/5!) - (x^7/7!)$

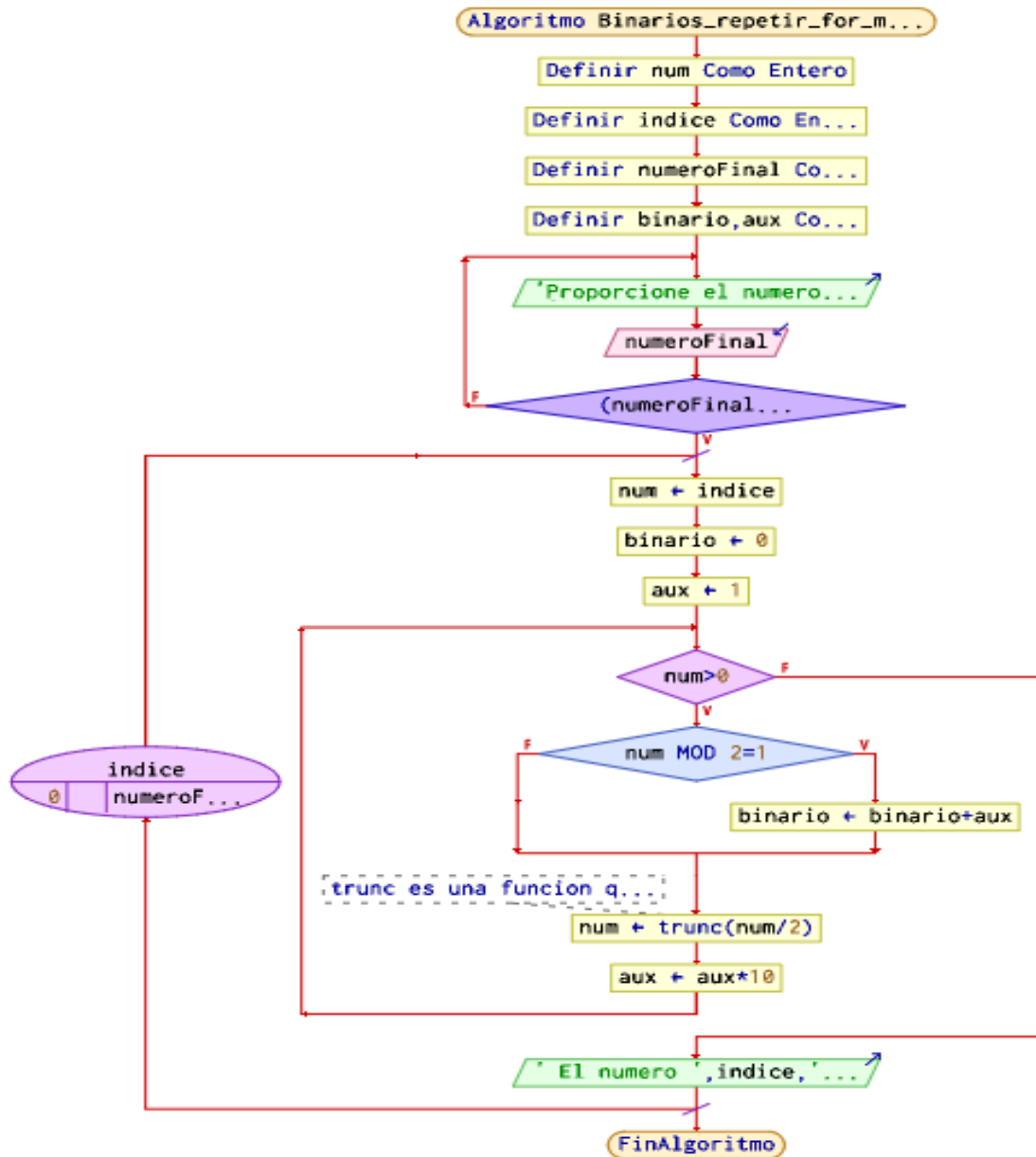
```
1  Algoritmo Serie_Seno //Serie de taylor es convergente y sirve para aproximar el ángulo
2      Definir numTerminos Como Entero;
3      Definir angulo Como Real;
4      Definir signo Como Entero;
5      Definir seno Como Real;
6      Definir nImpar Como Entero;
7      Definir factorial Como Real;;
8      signo ← 1;
9      factorial ← 1;
10     Escribir "Proporcione la cantidad de terminos que desea calcular en la serie";
11     Leer numTerminos;
12     Escribir "Proporcione el angulo que desea conocer";
13     Leer angulo;
14     seno ← angulo;
15     Para nImpar ← 3 con paso 2 hasta numTerminos * 2 + 1 Hacer
16         signo ← signo * (-1);
17         factorial ← nImpar * (nImpar - 1) * factorial;
18         seno ← seno + signo * (angulo ^ nImpar / factorial);
19     FinPara
20     Escribir "El seno de ", angulo, "es igual a:", seno;
21 FinAlgoritmo
```

Serie de taylor es con...



2. Realiza un algoritmo que reciba un número entero menor que 1024 y que imprima todos los números en binario desde cero hasta el número proporcionado.

```
1  Algoritmo Binarios_repetir_for_mientras
2      Definir num Como Entero;
3      Definir indice Como Entero;
4      Definir numeroFinal Como Entero;
5      Definir binario, aux Como Real;
6      Repetir
7          Escribir "Proporcione el numero que quiere convertir a binario (positivo menor a 1024)";
8          Leer numeroFinal;
9      Hasta Que (numeroFinal < 1024 y numeroFinal ≥ 0)
10     Para indice ← 0 hasta numeroFinal Hacer
11         num ← indice;
12         binario ← 0;
13         aux ← 1;
14         Mientras num > 0 Hacer
15             Si num mod 2 = 1 Entonces
16                 binario ← binario + aux;
17             FinSi
18             num ← trunc (num / 2); // trunc es una funcion que quita los decimales de una operación
19             aux ← aux * 10;
20         FinMientras
21         Escribir " El numero ", indice, " se escribe como ", binario, " en binario ";
22     FinPara
23 FinAlgoritmo
```



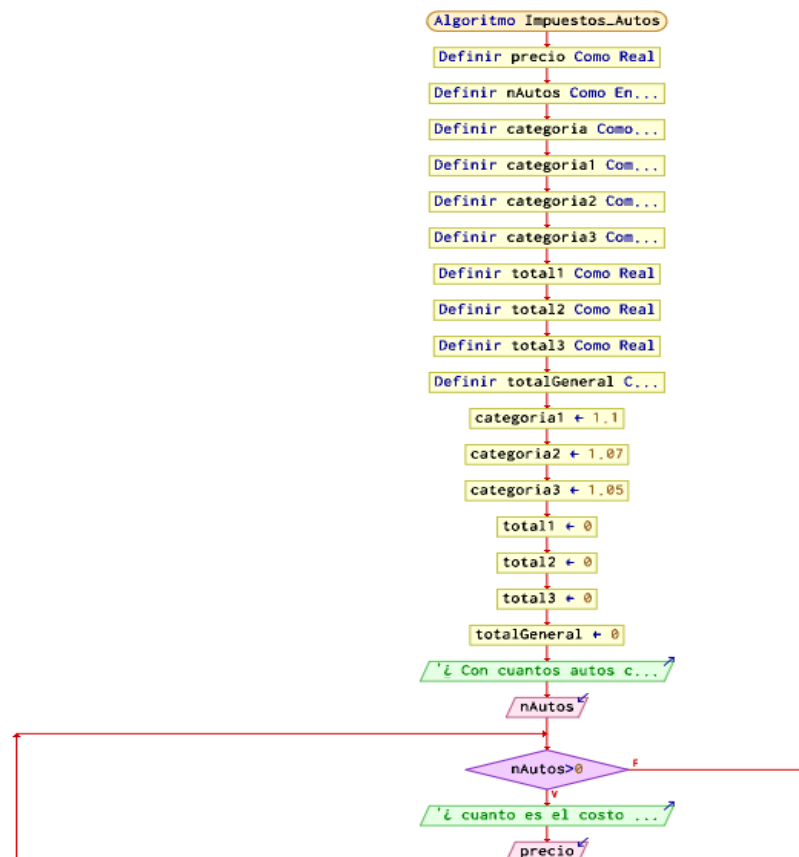
3. El gerente de una compañía automotriz desea determinar el impuesto que va a pagar por cada uno de los automóviles que posee, además del total que va a pagar por cada categoría y por todos los vehículos en total, basándose en la siguiente clasificación: Los vehículos de categoría 1 pagan 10% de su valor, los de categoría 2 pagan 7% de su valor, los de categoría 3 pagan 5%. Escriba un algoritmo que permita capturar los datos e imprimir los resultados. No se sabe cuántos de cada categoría hay. Usar ciclos MIENTRAS.

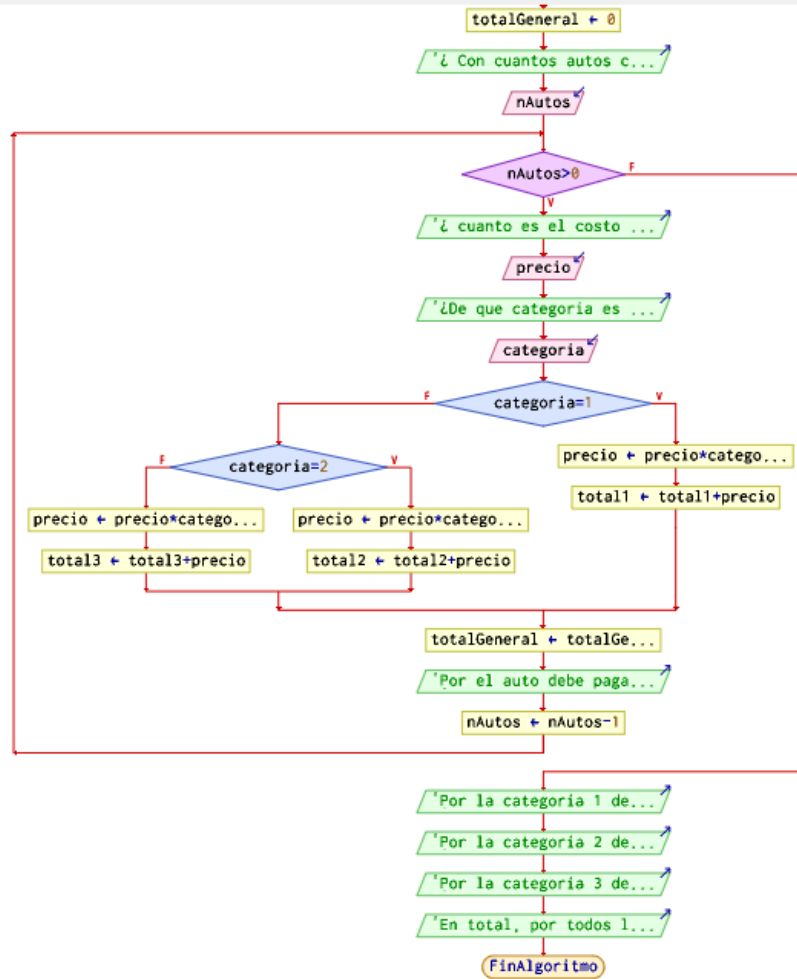
```
1  Algoritmo Impuestos_Autos
2      Definir precio Como Real;
3      Definir nAutos Como Entero;
4      Definir categoria Como Entero;
5      Definir categoria1 Como Real;
6      Definir categoria2 Como Real;
7      Definir categoria3 Como Real;
8      Definir total1 Como Real;
9      Definir total2 Como Real;
10     Definir total3 Como Real;
11     Definir totalGeneral Como Real;
12     categoria1 ← 1.1;
13     categoria2 ← 1.07;
14     categoria3 ← 1.05;
15     total1 ← 0;
16     total2 ← 0;
17     total3 ← 0;
18     totalGeneral ← 0;
19     Escribir "¿ Con cuantos autos cuenta?";
20     Leer nAutos;
21     Mientras nAutos > 0 Hacer
22         Escribir "¿ cuanto es el costo del auto?";
23         Leer precio;
24         Escribir "¿De que categoria es su auto?";
25         Leer categoria;
26         Si categoria = 1 Entonces
27             precio ← precio * categoria1;
28             total1 ← total1 + precio;
29         SiNo
30             Si categoria = 2 Entonces
31                 precio ← precio * categoria2;
32                 total2 ← total2 + precio;
33             Si categoria = 3 Entonces
34                 precio ← precio * categoria3;
35                 total3 ← total3 + precio;
```

```

34
35     SiNo
36         precio ← precio * categoria1;
37         total3 ← total3 + precio;
38     .....
39
40     FinSi
41 FinSi
42 totalGeneral ← totalGeneral + precio;
43 Escribir "Por el auto debe pagar", precio;
44 nAutos ← nAutos - 1;
45 FinMientras
46 Escribir "Por la categoría 1 debe pagar $", total1;
47 Escribir "Por la categoría 2 debe pagar $", total2;
48 Escribir "Por la categoría 3 debe pagar $", total3;
49 Escribir "En total, por todos los autos debes pagar $", totalGeneral;
50 FinAlgoritmo

```

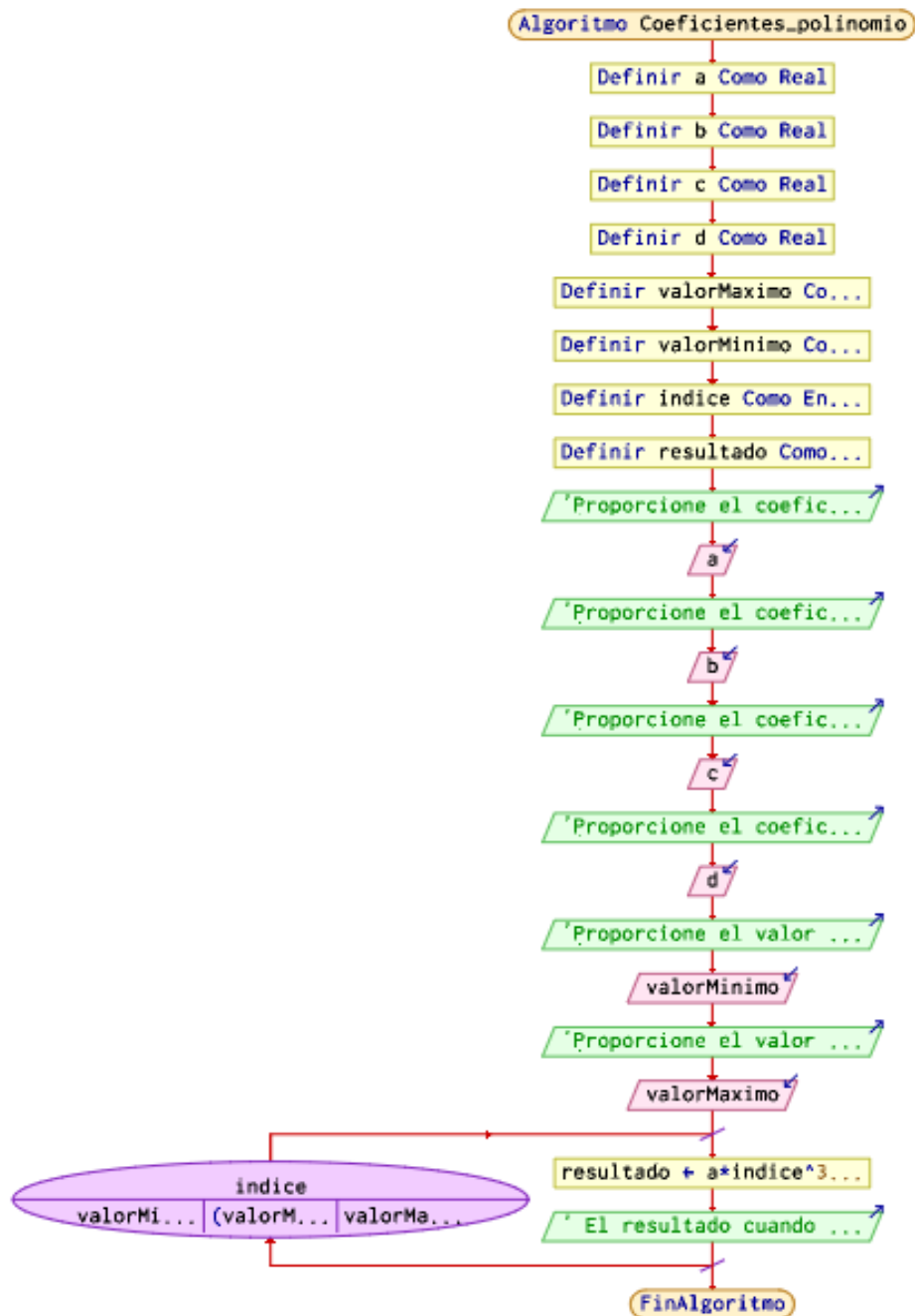




4. Escribir un algoritmo que permita capturar los coeficientes del polinomio  $AX^3 + BX^2 + CX + D$ , además de un valor mínimo para X y uno máximo. El algoritmo deberá generar una tabla con el valor del polinomio evaluado en 10 valores igualmente espaciados entre el mínimo y el máximo. Además, deberá imprimir si hubo un cambio de signo en ese rango. (Sugerencia: checa tu trabajo con [www.desmos.com](http://www.desmos.com))

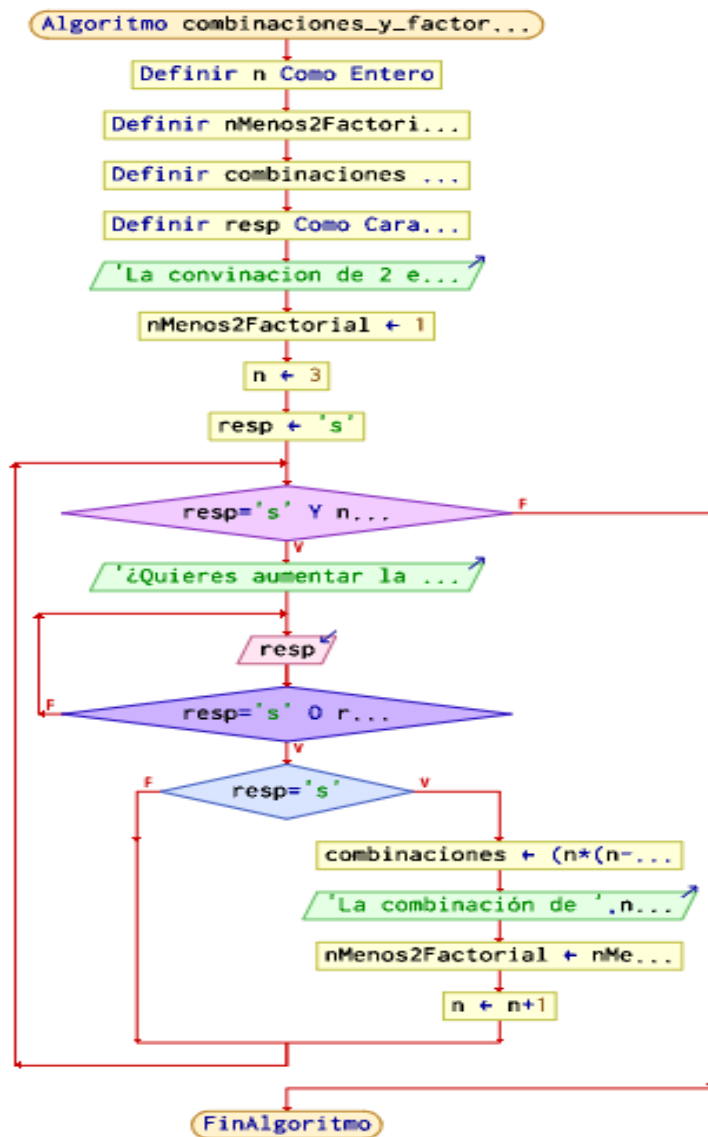
```
1  Algoritmo Coeficientes_polinomio
2      Definir a Como Real;
3      Definir b Como Real;
4      Definir c Como Real;
5      Definir d Como Real;
6      Definir valorMaximo Como Real;
7      Definir valorMinimo Como Real;
8      Definir indice Como Entero;
9      Definir resultado Como Real;
10     Escribir "Proporcione el coeficiente de término cúbico";
11     Leer a;
12     Escribir "Proporcione el coeficiente de término cuadrático";
13     Leer b;
14     Escribir "Proporcione el coeficiente de término lineal";
15     Leer c;
16     Escribir "Proporcione el coeficiente de término independiente";
17     Leer d;
18     Escribir "Proporcione el valor mínimo a evaluar";
19     Leer valorMinimo;
20     Escribir "Proporcione el valor máximo a evaluar";
21     Leer valorMaximo;
22     Para indice ← valorMinimo con paso (valorMaximo - valorMinimo)/9 hasta valorMaximo Hacer
23         resultado ← a * indice ^ 3 + b * indice ^ 2 + c * indice + d;
24
25         Escribir " El resultado cuando x = ", indice, " es igual a ", resultado;
26     FinPara
27 FinAlgoritmo
```





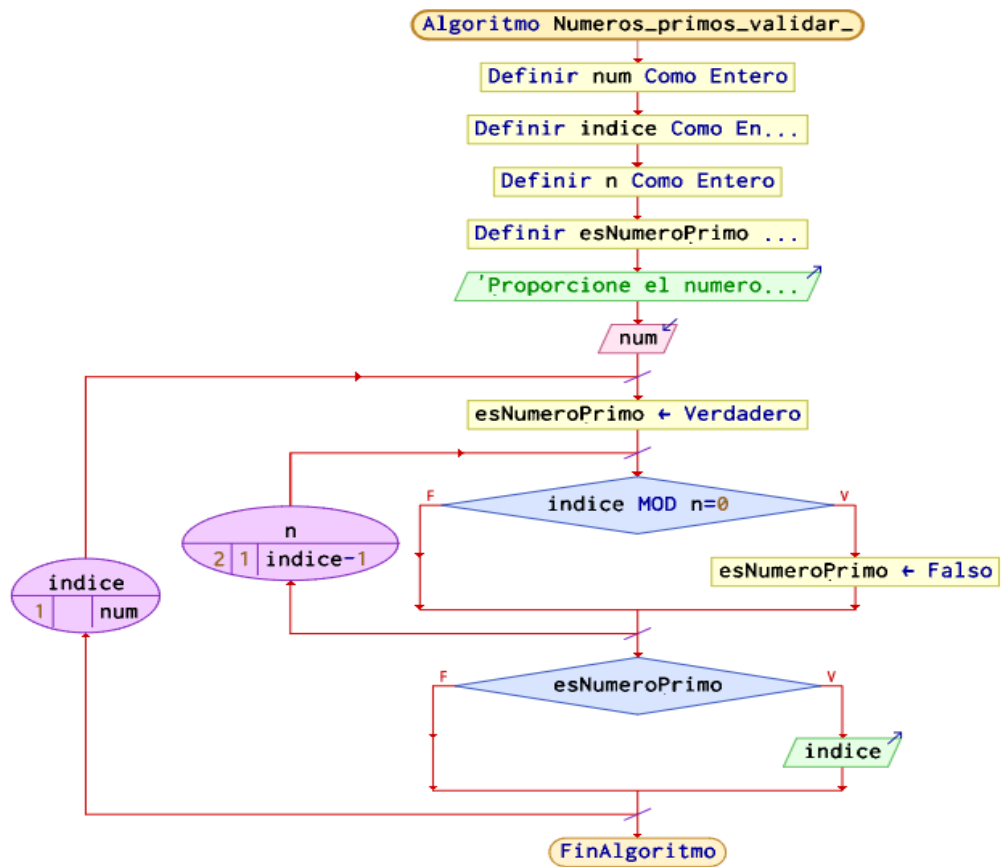
5. El número de combinaciones de  $n$  elementos tomados de 2 en 2 se puede calcular con la fórmula  $C(n,2)=n!/(2!(n-2)!)$ . Escriba un algoritmo que imprima  $C(1,2)$  después  $C(2,2)$  luego  $C(3,2)$  así sucesivamente hasta que el usuario diga que se detenga o se llegue a  $C(14,2)$ .

```
1  Algoritmo combinaciones_y_factoriales
2      Definir n Como Entero;
3      Definir nMenos2Factorial Como Entero;
4      Definir combinaciones Como Entero;
5      Definir resp Como Caracter;
6      Escribir "La convinacion de 2 elementos tomados de 2 en 2 es igual a 1";
7      nMenos2Factorial ← 1;
8      n ← 3;
9      resp ← "s";
10     Mientras resp = "s" y n ≤ 14 Hacer
11         Escribir "¿Quieres aumentar la cantidad de elemntos? s/n";
12         Repetir
13             Leer resp;
14         Hasta Que resp = "s" o resp = "n";
15         Si resp = "s" Entonces
16             combinaciones ← (n * (n- 1) * nMenos2Factorial)/ (2 * nMenos2Factorial);
17             Escribir "La combinación de ", n, " elementos tomados de 2 en 2 es igual a ", combinaciones;
18             nMenos2Factorial ← nMenos2Factorial * (n - 2);
19             n ← n + 1;
20         FinSi
21     FinMientras
22 FinAlgoritmo
```



6. Determinar si un número es primo se puede hacer verificando que no tiene divisores distintos de uno menores que él. Escriba un algoritmo que reciba un número entero y que imprima los números primos menores o iguales a él.

```
1  Algoritmo Numeros_primos_validar_  
2      Definir num Como Entero;  
3      Definir indice Como Entero;  
4      Definir n Como Entero;  
5      Definir esNumeroPrimo Como Logico;  
6      Escribir "Proporcione el numero para saber si hay numeros primos iguales o menores";  
7      Leer num;  
8      Para indice<- 1 Hasta num Hacer  
9          esNumeroPrimo =Verdadero;  
10         Para n<- 2 con paso 1 Hasta indice -1 Hacer  
11             Si indice % n = 0 Entonces  
12                 esNumeroPrimo<- Falso;  
13             FinSi  
14         FinPara  
15         Si esNumeroPrimo Entonces  
16             Escribir indice;  
17         FinSi  
18     FinPara  
19 FinAlgoritmo  
20
```

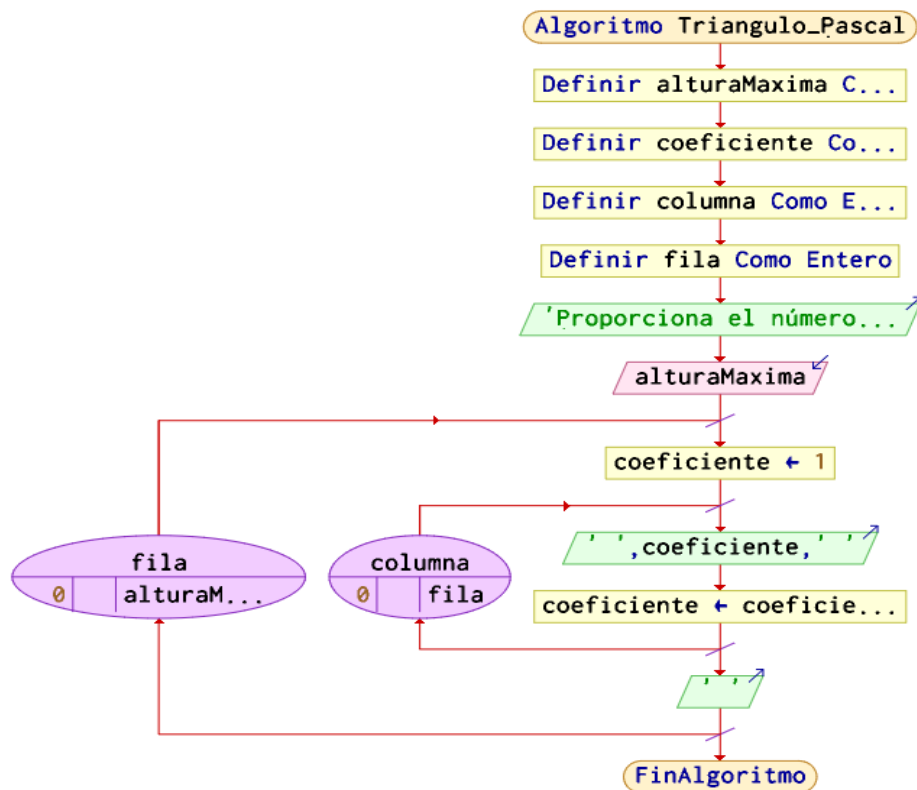


7. Escribir un programa que permita visualizar el triángulo de Pascal. El usuario proporciona un valor n, el cual representa la altura máxima del triángulo.

```

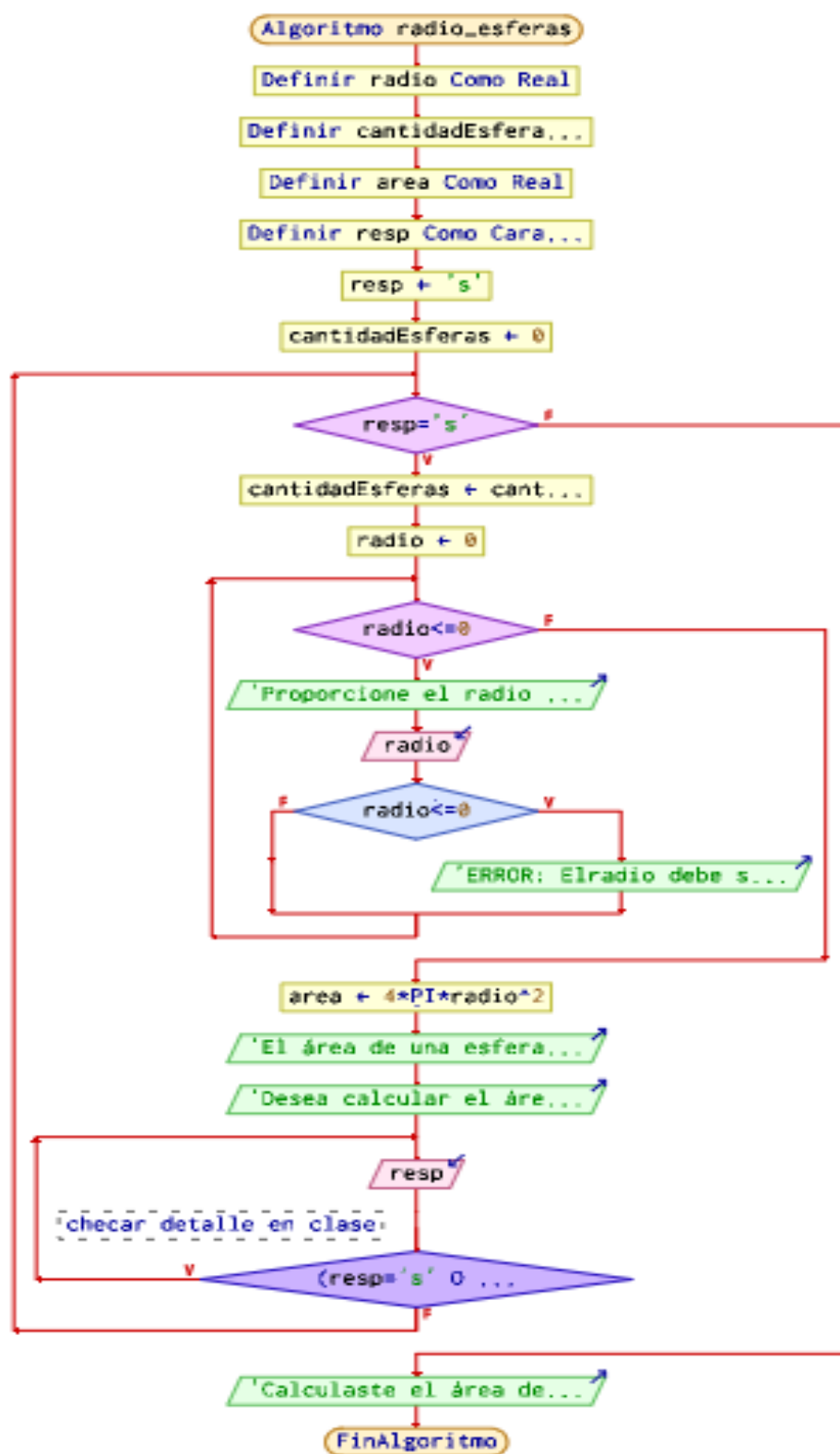
1  Algoritmo Triangulo_Pascal
2      Definir alturaMaxima Como Entero;
3      Definir coeficiente Como Entero;
4      Definir columna Como Entero;
5      Definir fila Como Entero;;
6      Escribir "Proporciona el número que representa la altura maxima del triangulo de Pascal";
7      Leer alturaMaxima;
8      Para fila← 0 Hasta alturaMaxima -1 Hacer
9          coeficiente← 1;
10         Para columna← 0 hasta fila hacer
11             Escribir " ", coeficiente, " " Sin Saltar;
12             coeficiente← coeficiente *(fila - columna)/(columna + 1);
13         FinPara
14         Escribir " ";
15     FinPara
16 FinAlgoritmo

```



8. Escribir un algoritmo, que pida el radio (dato de tipo real) de una esfera. En caso de que el radio sea menor o igual que 0, muestre el mensaje: "ERROR: El radio debe ser mayor que cero.". Que pida nuevamente el radio mientras el radio introducido sea incorrecto. Que escriba en pantalla: "El área de una esfera de radio <radio> es: <área>". Que pregunte "¿Desea calcular el área de otra esfera? s/n". Que repita todo el proceso si la respuesta es "s", en otro caso, que termine el ciclo y que imprima cuántas veces se calculó el área de una esfera. Nota: Área de una esfera =  $4 * \pi * \text{radio}^2$ . Utilice solo ciclos mientras.

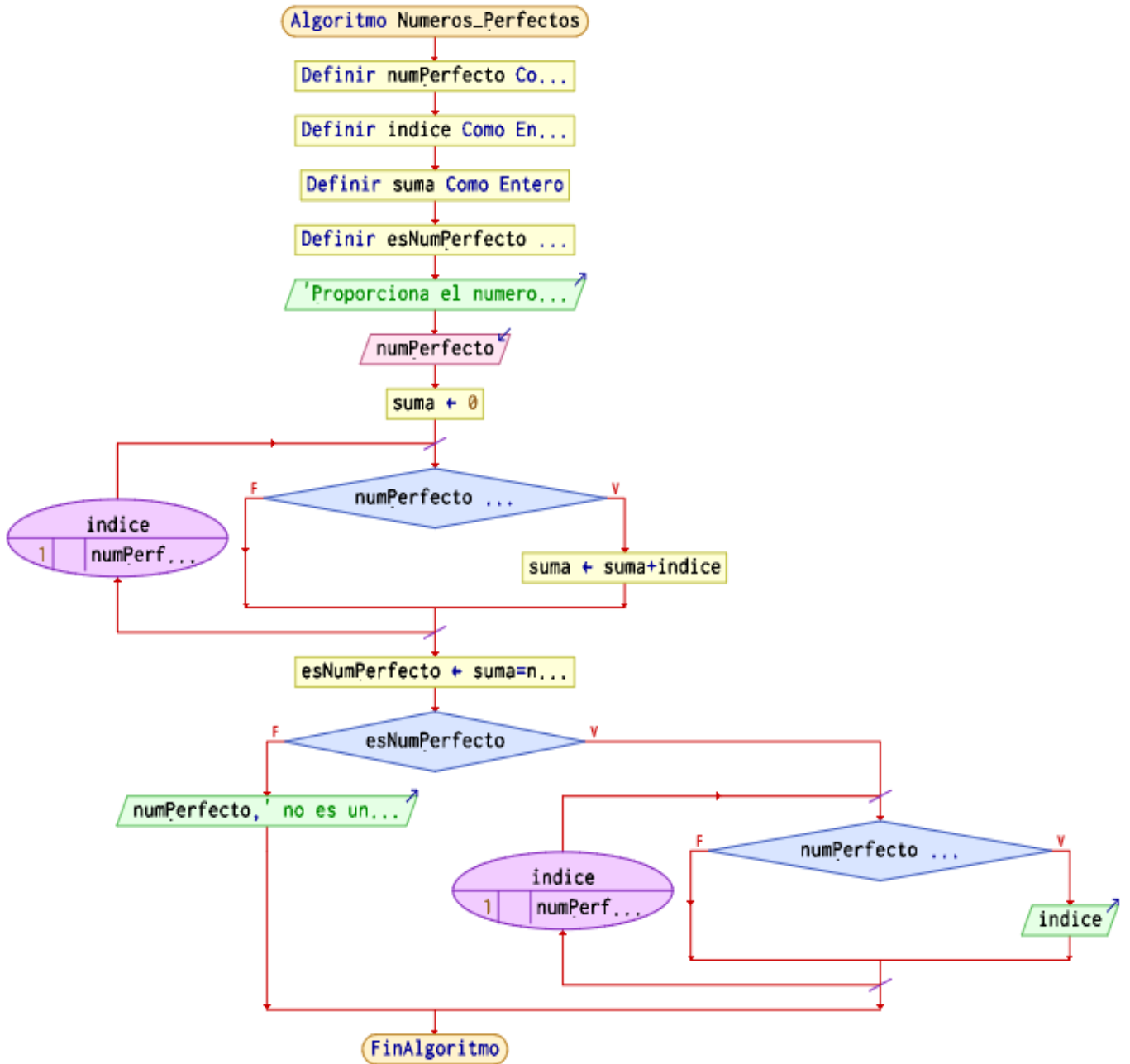
```
1  Algoritmo radio_esferas
2      Definir radio Como Real;
3      Definir cantidadEsferas Como Entero;
4      Definir area Como Real;
5      Definir resp Como Caracter;
6      resp ← "s";
7      cantidadEsferas ← 0;
8      Mientras resp = "s" Hacer
9          cantidadEsferas ← cantidadEsferas + 1;
10         radio ← 0;
11         Mientras radio ≤ 0
12             Escribir "Proporcione el radio de la esfera que desea calcular";
13             Leer radio;
14             Si radio ≤ 0 Entonces
15                 Escribir "ERROR: Elradio debe ser mayor que 0";
16             FinSi
17         FinMientras
18         area ← 4 * PI * radio ↑ 2;
19         Escribir "El área de una esfera de radio ", radio, " es: ", area;
20         Escribir "Desea calcular el área de otra esfera? s/n";
21         Hacer
22             Leer resp;
23         Mientras Que (resp = "s" o resp = "n") // checar detalle en clase
24         FinMientras
25         Escribir "Calculaste el área de ", cantidadEsferas, " esferas ";
26 FinAlgoritmo
```





9. Un número es perfecto si “la suma de sus divisores excepto el mismo es igual al propio número”.  
Haga un algoritmo para calcular e imprimir los números perfectos menores o iguales que N.

```
1  Algoritmo Numeros_Perfectos
2  Definir numPerfecto Como Entero;
3  Definir indice Como Entero;
4  Definir suma Como Entero;
5  Definir esNumPerfecto Como Logico;
6  Escribir "Proporciona el numero que deseas saber si es perfecto";
7  Leer numPerfecto;
8  suma ← 0;
9  Para indice ← 1 Hasta numPerfecto -1 Hacer
10     Si numPerfecto % indice = 0 Entonces
11         suma ← suma + indice;
12     FinSi
13 FinPara
14 esNumPerfecto ← suma = numPerfecto;
15 Si esNumPerfecto Entonces
16     Para indice ← 1 Hasta numPerfecto -1 Hacer
17         Si numPerfecto % indice = 0 Entonces
18             Escribir indice;
19         FinSi
20     FinPara
21 SiNo
22     Escribir numPerfecto, " no es un número perfecto";
23 FinSi
24 FinAlgoritmo
```



10. El departamento de policía de Mérida tiene los datos de las infracciones levantadas en la ciudad, la cual está dividida en 3 zonas. De cada infracción se tiene lo siguiente:

- Número de placa
- Zona de la infracción (A, B, C)
- Límite de velocidad en la zona en km/hr (A:90, B:100, C:110)
- Velocidad a la que iba el vehículo en km/hr.

Se asignan 100 pesos de multa por cada km/hr excedido del límite. Escriba un algoritmo que permita capturar un número no determinado de multas en un día. La salida para cada captura será la siguiente:

Placa	Zona	Limite	Velocidad	Multa
XXX	XXX	XXX	XXX	XXX

¿Otra infracción? s/n

y que al finalizar imprima la siguiente tabla:

#### TOTAL DE MULTAS EN MERIDA

Zona 1- Total de multas: XX	Monto: XXX
Zona 2- Total de multas: XX	Monto: XXX
Zona 3- Total de multas: XX	Monto: XXX

```

1  Algoritmo Policia_del_Diablo_infracciones
2      Definir cantidadMultas1 Como Entero;
3      Definir CantidadMultas2 Como Entero;
4      Definir cantidadMultas3 Como Entero;
5      Definir montoTotal1 Como Real;
6      Definir montoTotal2 Como Real;
7      Definir montoTotal3 Como Real;
8      Definir limite1 Como Entero;
9      Definir limite2 Como Entero;
10     Definir limite3 Como Entero;
11     Definir limite Como Entero;
12     Definir placa Como Caracter;
13     Definir zona Como Caracter;
14     Definir velocidad Como Real;
15     Definir multa Como Real;
16     Definir resp Como Caracter;
17
18     cantidadMultas1← 0;
19     CantidadMultas2← 0;
20     cantidadMultas3← 0;
21     montoTotal1← 0;
22     montoTotal2← 0;
23     montoTotal3← 0;
24     limite1← 90;
25     limite2← 100;
26     limite3← 110;
27

```

```
27
28     Repetir
29         Escribir "Proporcione la placa del auto";
30         Leer placa;
31         Repetir
32             Escribir "Proporcione la zona: A, B o C";
33             Leer zona;
34             zona← Mayusculas(zona);
35         Hasta Que zona ="A" o zona = "B" o zona = "C"
36         Escribir "Proporcione la velocidad del auto";
37         Leer velocidad;
38         Si zona = "A" Entonces
39             limite← limite1;
40         SiNo
41             Si zona ="B" Entonces
42                 limite← limite2;
43             SiNo
44                 limite← limite3;
45             FinSi
46         FinSi
47         multa← (velocidad - limite) * 100;
48         Si zona= "A" Entonces
49             cantidadMultas1← cantidadMultas1 + 1;
50             montoTotal1← montoTotal1 + multa;
51         SiNo
52             Si zona ="B" Entonces
53                 CantidadMultas2← CantidadMultas2 + 1;
54                 montoTotal2← montoTotal2 + multa;
55             SiNo
```

```

55         SiNo
56             cantidadMultas3 ← cantidadMultas3 + 1;
57             montoTotal3 ← montoTotal3 + multa;
58         FinSi
59     FinSi
60     Escribir "Placa | Zona | Limite | Velocidad | Multa";
61     Escribir placa, " | ", zona, " | ", limite, " | ", velocidad, " | ", multa;
62     Repetir
63         Escribir "Hay alguna otra infracción oficial? s/n";
64         Leer resp;
65     Hasta Que resp = "s" o resp = "n"
66 Hasta Que resp = "n"
67 Escribir "Zona A - Total de multas: ", cantidadMultas1, "    Monto: ", montoTotal1;
68 Escribir "Zona B - Total de multas: ", cantidadMultas2, "    Monto: ", montoTotal2;
69 Escribir "Zona C - Total de multas: ", cantidadMultas3, "    Monto: ", montoTotal3;
70 FinAlgoritmo

```

goritmo Policia\_del\_Diablo\_inf...

Definir cantidadMultas...

Definir cantidadMultas...

Definir cantidadMultas...

Definir montoTotal1 Co...

Definir montoTotal2 Co...

Definir montoTotal3 Co...

Definir limite1 Como E...

Definir limite2 Como E...

Definir limite3 Como E...

Definir limite Como En...

Definir placa Como Car...

Definir zona Como Cara...

Definir velocidad Como...

Definir multa Como Real

Definir resp Como Cara...

cantidadMultas1 + @

cantidadMultas2 + @

cantidadMultas3 + @

montoTotal1 + @

montoTotal2 + @

montoTotal3 + @

