**ADA08: Arreglos Formato envío:** ADA08\_apellido\_nombre.zip

**Descripción:** Utilizando la herramienta PSeInt realice los siguientes algoritmos mediante Pseudocódigo y diagramas de flujo.

**Notas:**

* Deberá enviar el código (archivo .psc) de cada ejercicio con el nombre 01\_apellido.psc, 02\_apellido.psc,…10\_apellido.psc, donde los números representa la numeración de los ejercicios y apelido es su apellido paterno.
* Deberá exportar sus digramas de flujo al formato de imagen .png y posteriormente pegarlo en word en el ejercicio correspondiente, esto para que el formato a colores se mantenga.
* Tanto los ejercicios como el documento Word comprimirlo en un solo archivo: **ADA08\_apellido\_nombre.zip** y enviarlo a la plataforma.
* Recuerde lo siguiente: utilizar el perfil UADY, escribir comentarios a su código, utilizar nombres de variables de acuerdo a lo que vaya a almacenar.

1. Se requiere un algoritmo que reciba como entrada un arreglo unidimensional **ordenado** de enteros (posiblemente repetidos) y genere como salida otro arreglo con los números enteros, pero sin repeticiones. Realice el diagrama de flujo y el pseudocódigo.

**Algoritmo** repetidosOrdenados

*//Definimos variables*

**Definir** ingreso**,** arreglo**,** arregloSinRepeticiones**,** n**,** nSinRespeticiones**,** i**,** j**,** repetidos **Como** **Entero;**

**Definir** seRepite **Como** **Logico;**

repetidos = 0**;**

*//Pedimos el tamano del arreglo*

**Escribir** "Ingrese el tamano del arreglo ordenado"**;**

**Leer** n**;**

**Dimension** arreglo**[**n**];**

*//Se llena el arreglo*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** "Ingresa el valor para la posicion "**,** i **+** 1**,** ": "**;**

**Leer** ingreso**;**

**Si** i **=** 0 **Entonces**

arreglo**[**i**]** = ingreso**;**

**SiNo**

**Si** ingreso**<**arreglo**[**i**-**1**]** **Entonces**

**Escribir** "debes ingresar un valor igual o mayor que: "**,** arreglo**[**i**-**1**];**

i = i **-**1**;**

**SiNo**

arreglo**[**i**]** = ingreso**;**

seRepite = **Falso;**

**Para** j = 0 **Hasta** i **-** 1 **Con** **Paso** 1 **Hacer**

**Si** arreglo**[**i**]** **==** arreglo**[**j**]** **y** j**≠**i **Entonces**

seRepite = **Verdadero;**

**Fin** **Si**

**Fin** **Para**

**Si** seRepite **Entonces**

repetidos = repetidos **+** 1**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Si**

**Fin** **Para**

**Dimension** arregloSinRepeticiones**[**n**-**repetidos**];**

*//pasamos por el primer arreglo y guardamos los que no se repiten en el otro*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Si** i **=** 0 **Entonces**

arregloSinRepeticiones**[**0**]** = arreglo **[**0**];**

j=1**;**

**SiNo**

**Si** arreglo**[**i**]≠**arregloSinRepeticiones**[**j**-**1**]** **Entonces**

arregloSinRepeticiones**[**j**]** = arreglo**[**i**];**

j = j **+**1**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Para**

*//Escribimos el arreglo sin repeticiones*

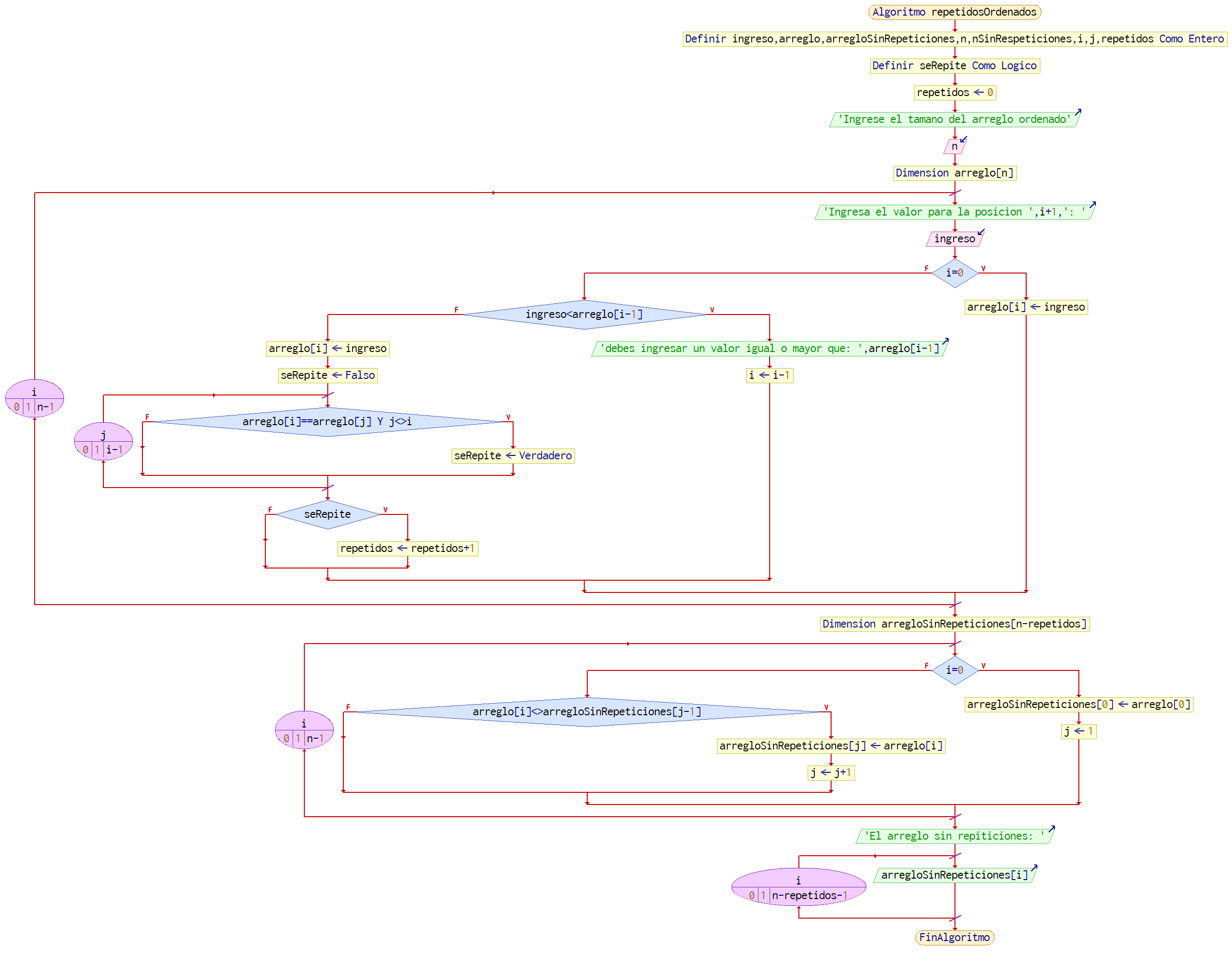
**Escribir** "El arreglo sin repiticiones: "**;**

**Para** i = 0 **Hasta** n **-** repetidos **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** arregloSinRepeticiones**[**i**];**

**Fin** **Para**

**FinAlgoritmo**



1. Se requiere un algoritmo que reciba como entrada un arreglo unidimensional **desordenado** de enteros (posiblemente repetidos) y genere como salida ese mismo arreglo, pero sin repeticiones. Realice el diagrama de flujo y el pseudocódigo.

**Algoritmo** repetidosDesordenados

*//Definimos variables*

**Definir** arreglo**,** arregloSinRepeticiones**,** n**,** i**,** j**,**k**,** repetidos **Como** **Entero;**

**Definir** seRepite **Como** **Logico;**

repetidos = 0**;**

*//Pedimos el tamano del arreglo*

**Escribir** "Ingrese el tamano del arreglo"**;**

**Leer** n**;**

**Dimension** arreglo**[**n**];**

*//Se llena el arreglo*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** "Ingresa el valor para la posicion "**,** i **+** 1**,** ": "**;**

**Leer** arreglo**[**i**];**

**Si** i **≠** 0 **Entonces**

seRepite = **Falso;**

**Para** j = 0 **Hasta** i **-** 1 **Con** **Paso** 1 **Hacer**

**Si** arreglo**[**i**]** **==** arreglo**[**j**]** **y** j**≠**i **Entonces**

seRepite = **Verdadero;**

**Fin** **Si**

**Fin** **Para**

**Si** seRepite **Entonces**

repetidos = repetidos **+** 1**;**

**Escribir** repetidos**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Para**

**Dimension** arregloSinRepeticiones**[**n**-**repetidos**];**

*//pasamos por el primer arreglo y guardamos los que no se repiten en el otro*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Si** i **=** 0 **Entonces**

arregloSinRepeticiones**[**0**]** = arreglo **[**0**];**

j=1**;**

**SiNo**

seRepite=**Falso;**

**Para** k = 0 **Hasta** j**-**1 **Con** **Paso** 1 **Hacer**

**Si** arreglo**[**i**]==**arregloSinRepeticiones**[**k**]** **Entonces**

seRepite = **Verdadero;**

**Fin** **Si**

**Fin** **Para**

**Si** seRepite **==** **Falso** **Entonces**

arregloSinRepeticiones**[**j**]** = arreglo**[**i**];**

j = j **+** 1**;**

**FinSi**

**Fin** **Si**

**Fin** **Para**

*//Escribimos el arreglo sin repeticiones*

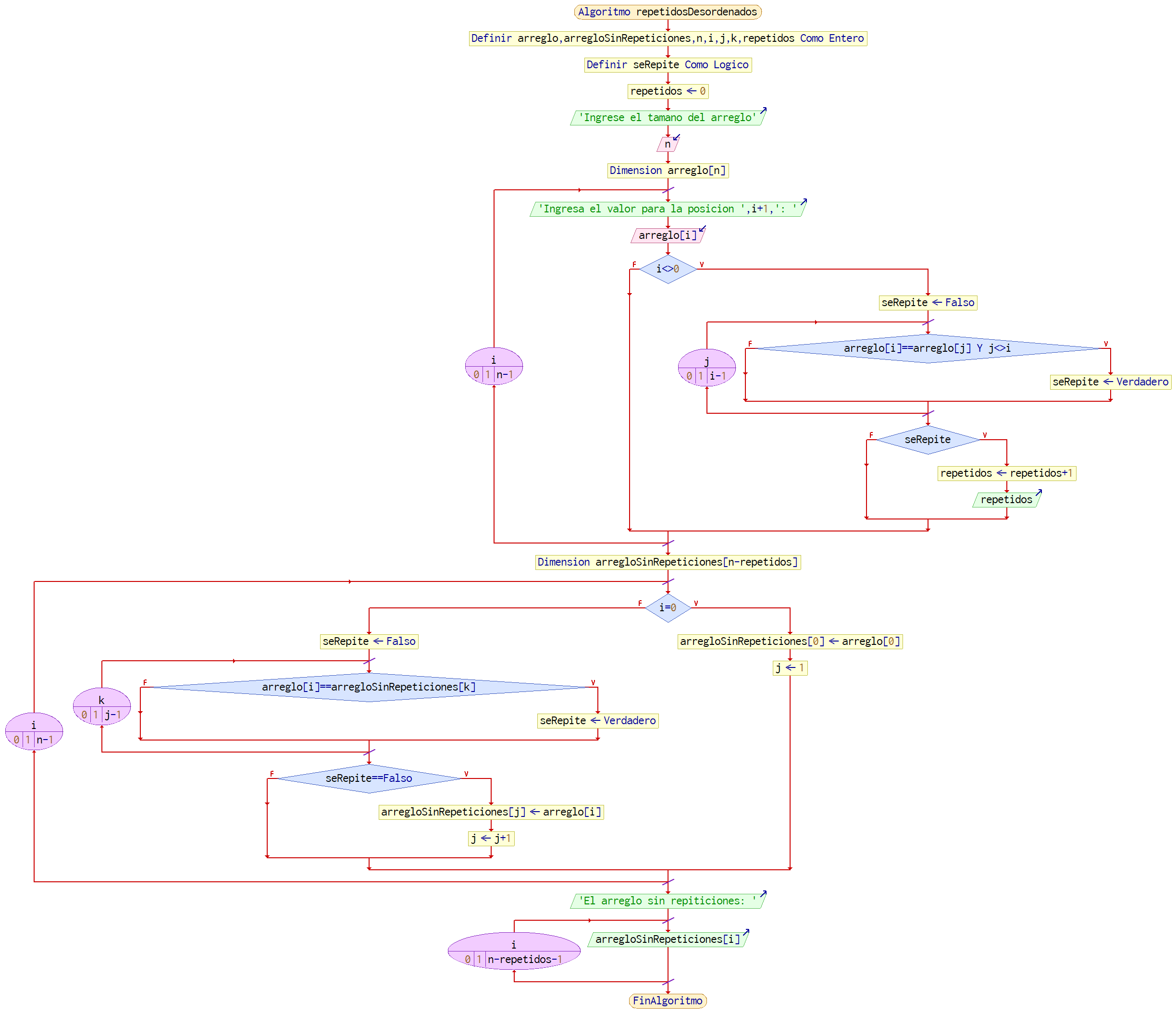
**Escribir** "El arreglo sin repiticiones: "**;**

**Para** i = 0 **Hasta** n **-** repetidos **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** arregloSinRepeticiones**[**i**];**

**Fin** **Para**

**FinAlgoritmo**



1. Suponga que en una elección hubo 12 candidatos (con identificadores 1, 2, 3, …, 12). Por otra parte, los votos para cada candidato se teclean de manera desorganizada como se muestra a continuación:

1 5 7 5 1 12 10 7 1 7 5 8 1 5 -1

Nota: El final de los datos está dado por -1

Construya un algoritmo (proporcione el diagrama de flujo y el pseudocódigo) que pueda proporcionar la siguiente información:

1. El número de votos de cada candidato al final de cada elección
2. El candidato ganador, el número de votos que obtuvo y el porcentaje correspondiente del total de la elección. Suponga que el candidato ganador no empató en número de votos con otro candidato.

**Algoritmo** candidatos3

*//Definicion e inicializacion de variables*

**Definir** candidatos**,** i**,** candidato**,** votosCandidato**,** votosTotales**,** vecesMayor **Como** **Entero;**

**Definir** porcentaje **Como** **Real;**

**Definir** faltanVotos**,** hayEmpate **Como** **Logico;**

**Dimension** candidatos**[**12**];**

faltanVotos = **Verdadero;**

hayEmpate = **Falso;**

votosCandidato = 0**;**

votosTotales = 0**;**

vecesMayor = 0**;**

**Para** i=0 **Hasta** 11 **Con** **Paso** 1 **Hacer**

candidatos**[**i**]** = 0**;**

**Fin** **Para**

*//Solicitamos los votos por el id del candidato*

**Escribir** "Para contar los votos, ingrese el identificador del candidato, cuando termine ingrese -1."**;**

**Mientras** faltanVotos **Hacer**

**Escribir** **Sin** **Saltar** "Id del candidato:"**;**

**Leer** candidato**;**

*//si es -1 deiamos de contar votos*

**Si** candidato **=** **-**1 **Entonces**

faltanVotos = **Falso;**

**SiNo**

*//Si dan un candidato inexistente, avisar y no contar el voto*

**Si** candidato**≥**13 **o** candidato**≤**0 **Entonces**

**Escribir** "Ese candidato no existe, sólo hay candidatos 1-12."**;**

**SiNo**

*//contamos el voto*

candidato = candidato **-**1**;**

candidatos**[**candidato**]** = candidatos**[**candidato**]** **+** 1**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Mientras**

*//Buscamos al mayor, su total, porcentaie*

**Para** i=0 **Hasta** 11 **Con** **Paso** 1 **Hacer**

votosTotales = votosTotales **+** candidatos**[**i**];**

**Escribir** i**,** " "**,** candidatos**[**i**],** " "**,** votosTotales**;**

**Si** i**==**0 **Entonces**

votosCandidato = candidatos**[**i**];**

candidato = i **+** 1**;**

**SiNo**

**Si** candidatos**[**i**]** **>** votosCandidato **Entonces**

votosCandidato = candidatos**[**i**];**

candidato = i **+** 1**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Para**

*//Buscamos empates*

**Para** i=0 **Hasta** 11 **Con** **Paso** 1 **Hacer**

**Si** candidatos**[**i**]** **==** votosCandidato **Entonces**

vecesMayor = vecesMayor **+** 1**;**

**Si** vecesMayor **≥**2 **Entonces**

hayEmpate = **Verdadero;**

i = 12**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Para**

*//revisamo que al menos haya un voto*

**Si** votosTotales **==** 0 **Entonces**

**Escribir** "Nadie voto, entonces nadie gano :(."**;**

**SiNo**

*//Imprimimos los resultados en caso de que no haya empate*

**Si** hayEmpate **Entonces**

**Escribir** "Hubo empate, los votos se deben hacer de nuevo."**;**

**SiNo**

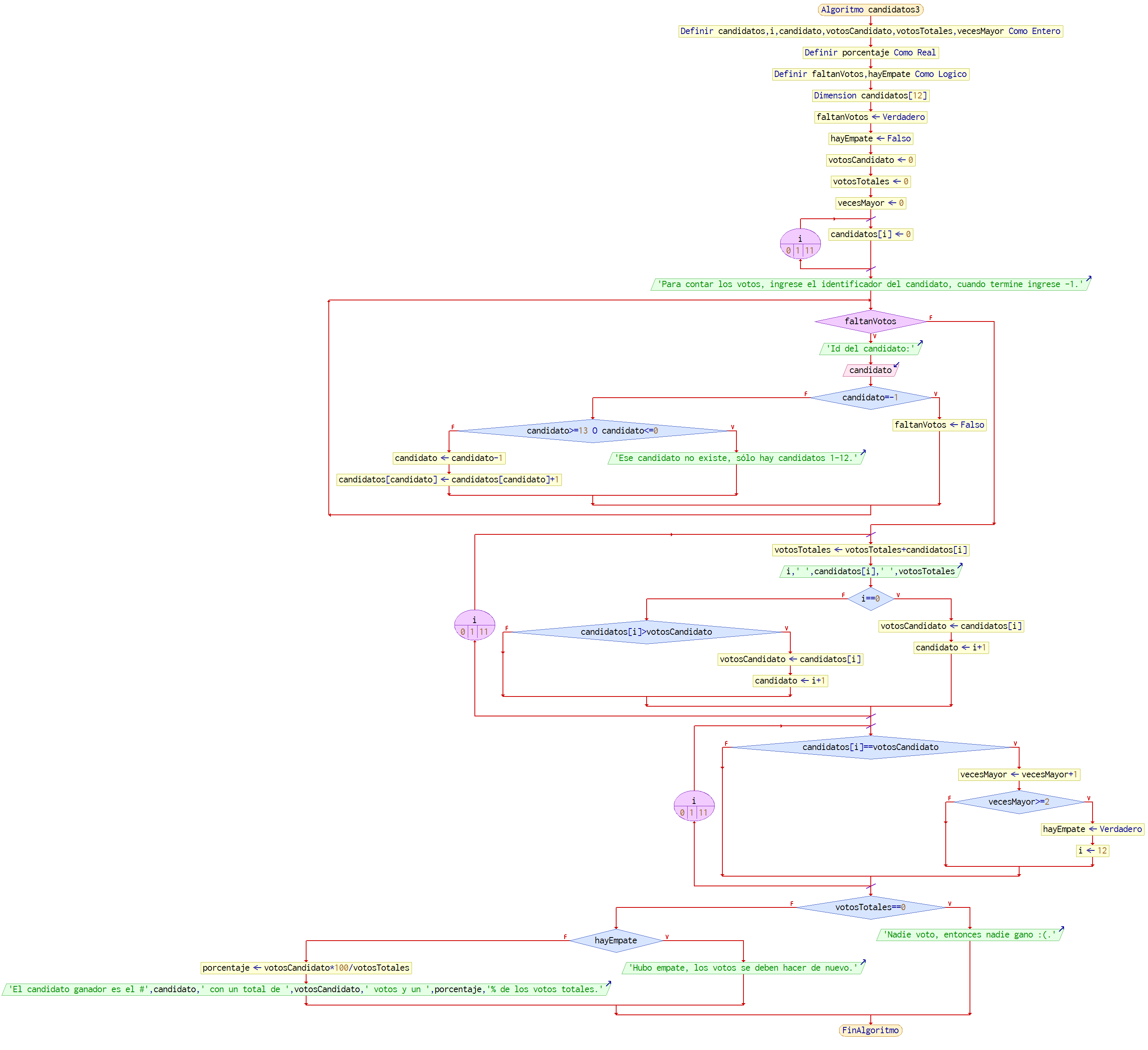
porcentaje = votosCandidato**\***100**/**votosTotales**;**

**Escribir** "El candidato ganador es el #"**,** candidato**,** " con un total de "**,** votosCandidato**,** " votos y un "**,** porcentaje**,** "% de los votos totales."**;**

**Fin** **Si**

**FinSi**

**FinAlgoritmo**



1. Se requiere un algoritmo para almacenar en un arreglo unidimensional los primeros N números primos de M números proporcionados por el usuario, valide que M > N. Al final imprima el arreglo de los números primos. Represéntelo mediante el pseudocódigo y el diagrama de flujo.

**SubAlgoritmo** esPrimo **<-** validarPrimo **(** numeroAValidar **)**

**Definir** raizV **Como** **Real;**

**Definir** esPrimo **Como** **Logico;**

**Definir** i **Como** **Entero;**

raizV = trunc**(**rc**(**numeroAValidar**));**

esPrimo = **Verdadero;**

**Si** numeroAValidar**≤**1 **Entonces**

esPrimo = **Falso;**

**SiNo**

**Para** i = 2 **Hasta** raizV **Con** **Paso** 1 **Hacer**

*//Si el resto es 0 entonces es divisible por algún número*

*//entonces no es un numero primo*

**Si** numeroAValidar**%**i **==** 0 **Entonces**

esPrimo = **Falso;**

i = raizV**;**

**Fin** **Si**

**Fin** **Para**

**Fin** **Si**

**Fin** **SubAlgoritmo**

**Algoritmo** mPrimosN

*//Definimos variables*

**Definir** n**,** m**,** numerosArray**,** primosArray**,** i**,** contadorPrimos**,** j **Como** **Entero;**

contadorPrimos = 0**;**

*//pedimos la cantidad de números primos y los numeros a revisar*

**Escribir** **Sin** **Saltar** "¿Cuántos números tiene?"**;**

**Leer** m**;**

n=m**+**1**;**

**Mientras** n**>**m **Hacer**

**Escribir** **Sin** **Saltar** "¿Cuántos primeros primos quiere?"**;**

**Leer** n**;**

**Si** n**>**m **Entonces**

**Escribir** "Debe pedir menos primos que el tamaño de los numeros que tienes, repita el ingreso de valor."**;**

**Fin** **Si**

**Fin** **Mientras**

**Dimension** numerosArray**[**m**];**

*//Guardamos los números del usuario*

**Para** i = 0 **Hasta** m**-**1 **Con** **Paso** 1 **Hacer**

**Escribir** **Sin** **Saltar** "Ingrese el valor para el número "**,** i **+** 1**,** ":"**;**

**Leer** numerosArray**[**i**];**

**Si** validarPrimo**(**numerosArray**[**i**])** **Entonces**

contadorPrimos = contadorPrimos **+** 1**;**

**Fin** **Si**

**Fin** **Para**

**Dimension** primosArray**[**contadorPrimos**];**

j = 0**;**

**Para** i = 0 **Hasta** m**-**1 **Con** **Paso** 1 **Hacer**

**Si** validarPrimo**(**numerosArray**[**i**])** **Entonces**

primosArray**[**j**]** = numerosArray**[**i**];**

j = j **+** 1**;**

**Si** j **=** contadorPrimos **Entonces**

i = m**;**

**Fin** **Si**

**Fin** **Si**

**Fin** **Para**

**Si** contadorPrimos**<**n **Entonces**

**Escribir** "No hubo "**,** n**,** " primos para tus números pero te presento los primeros "**,** contadorPrimos**,** " primos disponibles."**;**

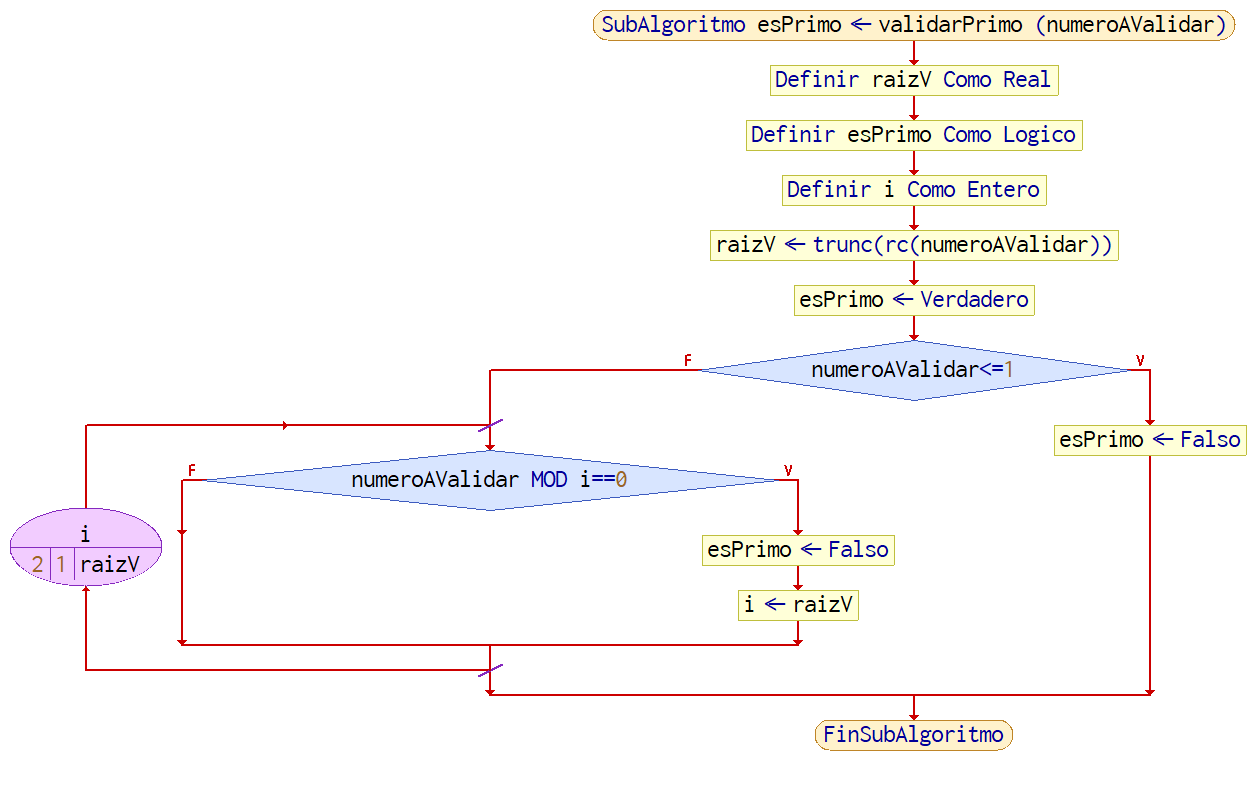
**FinSi**

**Para** i = 0 **Hasta** contadorPrimos**-**1 **Con** **Paso** 1 **Hacer**

**Escribir** i**+**1**,**"-. "**,** primosArray**[**i**];**

**Fin** **Para**

**FinAlgoritmo**



1. En un arreglo unidimensional se almacenan las calificaciones de N alumnos. Construya un algoritmo (proporcione el diagrama de flujo y pseudocódigo) que permita calcular e imprimir lo siguiente:
   1. El promedio general del grupo.
   2. Número de alumnos aprobados y número de alumnos reprobados (Si el alumno sacó una calificación menor a 70 se le considera reprobado).
   3. Porcentaje de alumnos aprobados y reprobados.
   4. Número de alumnos cuya calificación fue mayor a 80.

**Algoritmo** calificaciones

*//Definimos variables*

**Definir** n**,** i**,** aprobados**,** reprobados**,** mayor80 **Como** **Entero;**

**Definir** porctjAprobados**,** porctjReprobados**,** promedio**,** calificacion**,** calificacionesArreglo **Como** **Real;**

promedio = 0**;**

mayor80 = 0**;**

aprobados = 0**;**

reprobados = 0**;**

*//Pedimos la cantidad n de alumnos*

**Escribir** "¿Cuántos alumnos son?"**;**

**Leer** n**;**

**Dimension** calificacionesArreglo**[**n**];**

*//por cada alumno pedimos calificacion*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** "Ingresa la calificación del alumno #"**,**i**+**1**;**

**Leer** calificacion**;**

promedio = **((**promedio **\*** i**)** **+** calificacion**)** **/** **(**i**+**1**);**

**Si** calificacion**≥**70 **Entonces**

aprobados = aprobados **+** 1**;**

porctjAprobados = aprobados **\*** 100 **/** n**;**

**Si** calificacion **>** 80 **Entonces**

mayor80 = mayor80 **+** 1**;**

**Fin** **Si**

**SiNo**

reprobados = reprobados **+** 1**;**

porctjReprobados = reprobados **\*** 100 **/** n**;**

**Fin** **Si**

calificacionesArreglo**[**i**]** = calificacion**;**

**Fin** **Para**

*//Imprimimos resultados*

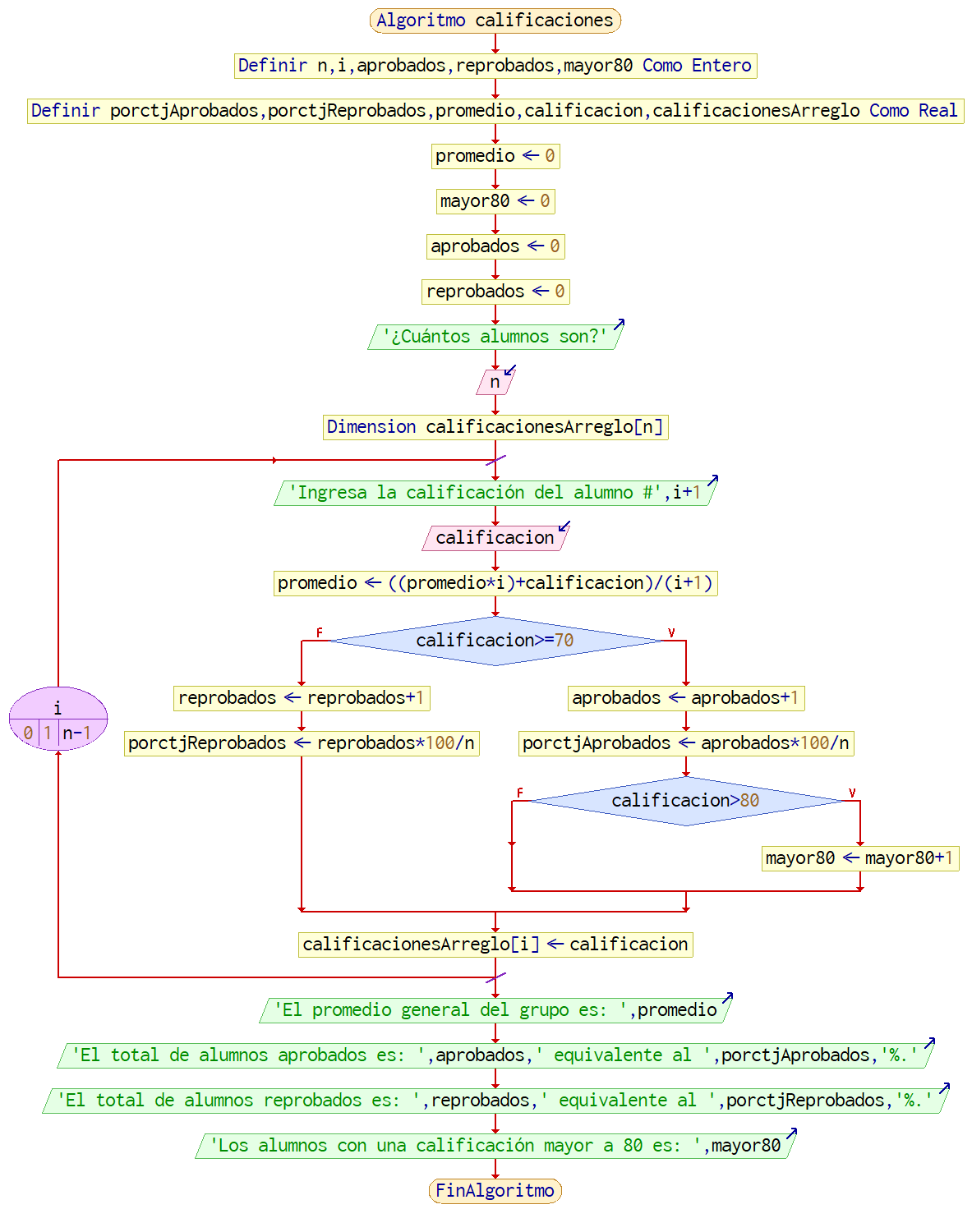
**Escribir** "El promedio general del grupo es: "**,** promedio**;**

**Escribir** "El total de alumnos aprobados es: "**,** aprobados**,** " equivalente al "**,** porctjAprobados**,** "%."**;**

**Escribir** "El total de alumnos reprobados es: "**,** reprobados**,** " equivalente al "**,** porctjReprobados**,** "%."**;**

**Escribir** "Los alumnos con una calificación mayor a 80 es: "**,** mayor80**;**

**FinAlgoritmo**



1. Dado un arreglo unidimensional de tipo entero que contiene las calificaciones de un grupo de alumnos que presentaron examen de ingreso para una universidad, construya un algortimo (proporcione el diagrama de flujo y pseudocódigo) que calcule lo siguiente:
   1. La media aritmética. Esta se calcula como la suma de los elementos entre el número de elementos.
   2. La varianza. Esta se calcula como la suma de los cuadrados de las desviaciones de la media, entre el número de elementos.
   3. La desviación estándar. Esta se calcula como la raíz cuadrada de la varianza.
   4. La moda. Esta se calcula obteniendo el número que más frecuencia tiene.

**Algoritmo** examenIngreso

*//Definimos variables*

**Definir** n**,** i**,** j**,** calificacion**,** repeticiones**,** repeticionesP**,** posicionModa **Como** **Entero;**

**Definir** promedio**,** calificaciones**,** varianza**,** desviacionE **Como** **Real;**

promedio = 0**;**

varianza = 0**;**

*//Pedimos la cantidad n de alumnos*

**Escribir** "Ingresa la cantidad de alumnos"**;**

**Leer** n**;**

**Dimension** calificaciones**[**n**];**

*//por cada alumno pedimos calificacion*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** "Ingresa la calificación del alumno #"**,**i**+**1**;**

**Leer** calificacion**;**

promedio = **((**promedio **\*** i**)** **+** calificacion**)** **/** **(**i**+**1**);**

calificaciones**[**i**]** = calificacion**;**

**Fin** **Para**

repeticionesP = 0**;**

*//Iteramos para obtener la suma de los cuadrados y la posicion de la moda*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

repeticiones = 0**;**

varianza = varianza **+** **(**calificaciones**[**i**]-**promedio**)↑**2**;**

**Para** j = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Si** calificaciones**[**i**]==**calificaciones**[**j**]** **Y** i**≠**j **Entonces**

repeticiones = repeticiones **+** 1**;**

**Fin** **Si**

**Fin** **Para**

**Si** repeticiones**>**repeticionesP **Entonces**

repeticionesP = repeticiones**;**

posicionModa = i**;**

**Fin** **Si**

**Fin** **Para**

varianza = varianza**/**n**;**

desviacionE = rc**(**varianza**);**

*// for(int i=0;i<n;i++){*

*// s=0;*

*// for(int j=0;j<n;j++){*

*// if(A[i]==A[j] & i<>j){*

*// s=s+1;*

*// }*

*// }*

*// if(s>=m){*

*// m=s;//guardamos el mayor por el momento*

*// a=i;*

*// }*

*// }*

*//Imprimimos resultados*

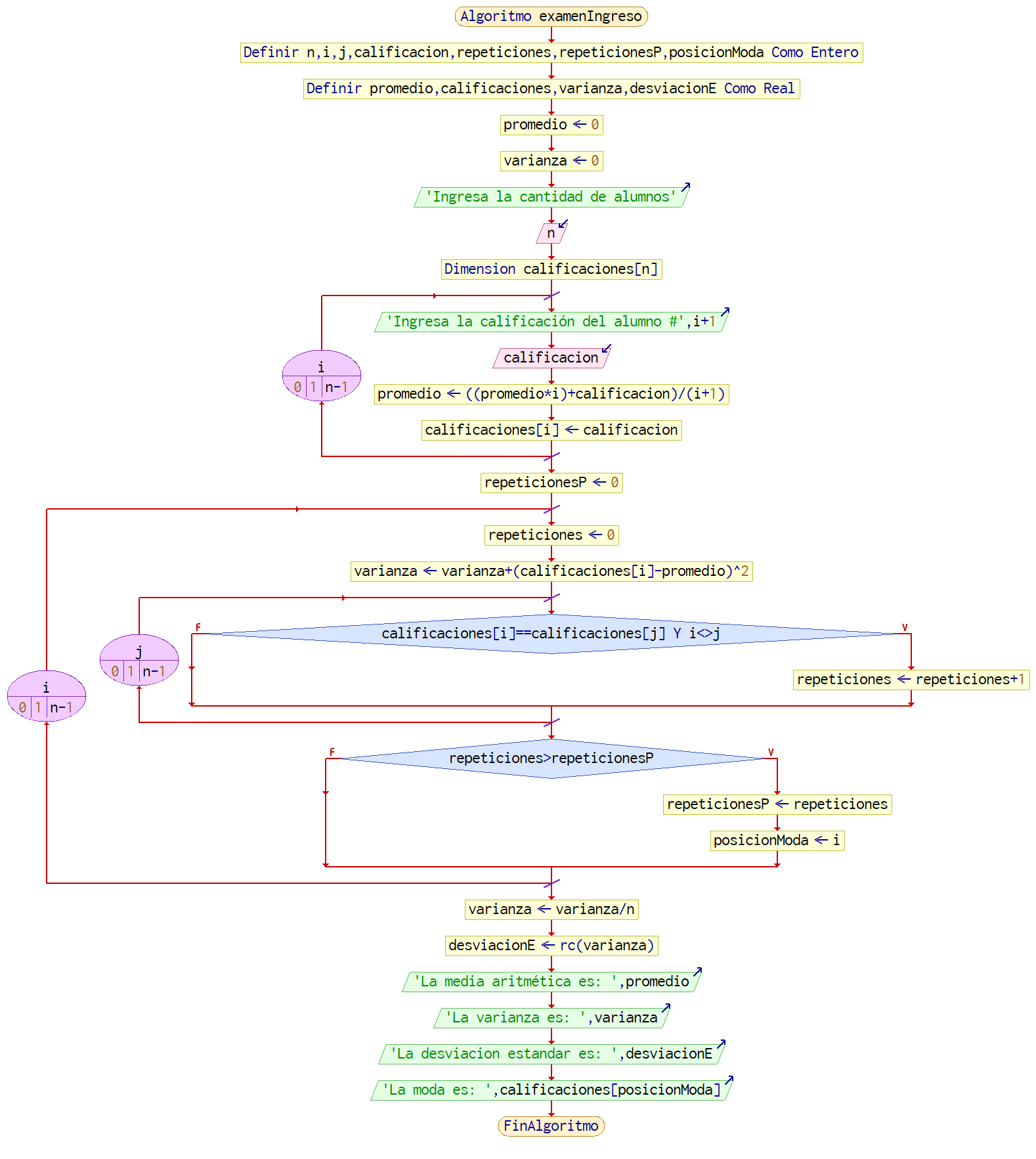
**Escribir** "La media aritmética es: "**,** promedio**;**

**Escribir** "La varianza es: "**,** varianza**;**

**Escribir** "La desviacion estandar es: "**,** desviacionE**;**

**Escribir** "La moda es: "**,** calificaciones**[**posicionModa**];**

**FinAlgoritmo**



1. En un arreglo unidimensional se ha almacenado el número total de toneladas de cereales cosechadas durante cada mes del año anterior. Construya un algoritmo (proporcione el diagrama de flujo y pseudocódigo) que proporcione la siguiente información:
   1. El promedio anual de toneladas cosechadas.
   2. ¿Cuántos meses tuvieron una cosecha superior al promedio anual?
   3. ¿Cuántos meses tuvieron una cosecha inferior al promedio anual?
   4. ¿Cuál fue el mes en el que se produjeron mayor número de toneladas?.

**Algoritmo** cosechasDeCereal

*//Definimos variables*

**Definir** cosechas**,** promedio**,** toneladas **Como** **Real;**

**Definir** mayorPromedio**,** menorPromedio**,** mes**,** i **,** masToneladas **Como** **Entero;**

**Definir** meses **Como** **Caracter;**

mayorPromedio = 0**;**

menorPromedio = 0**;**

promedio = 0**;**

**Dimension** cosechas**[**12**];**

**Dimension** meses**[**12**];**

meses**[**0**]** = "Enero"**;**

meses**[**1**]** = "Febrero"**;**

meses**[**2**]** = "Marzo"**;**

meses**[**3**]** = "Abril"**;**

meses**[**4**]** = "Mayo"**;**

meses**[**5**]** = "Junio"**;**

meses**[**6**]** = "Julio"**;**

meses**[**7**]** = "Agosto"**;**

meses**[**8**]** = "Septiembre"**;**

meses**[**9**]** = "Octubre"**;**

meses**[**10**]** = "Noviembre"**;**

meses**[**11**]** = "Diciembre"**;**

*//Llenamos las cosechas con las toneladas obtenidas*

**Para** i = 0 **Hasta** 11 **Con** **Paso** 1 **Hacer**

**Escribir** **Sin** **Saltar** "Ingrese las toneladas obtenidas en el mes de "**,** meses**[**i**];**

**Leer** toneladas**;**

*//calculamos el promedio*

promedio = **((**promedio **\*** i**)** **+** toneladas**)** **/** **(**i**+**1**);**

*//Obtenemos el mes con más toneladas de cosecha*

**Si** i**==**0 **Entonces**

mes = 0**;**

**SiNo**

**Si** toneladas**>**cosechas**[**mes**]** **Entonces**

mes = i**;**

**FinSi**

**Fin** **Si**

cosechas**[**i**]** = toneladas**;**

**Fin** **Para**

*//comparamos con el promedio para saber cuantos son mayores o menores*

**Para** i = 0 **Hasta** 11 **Con** **Paso** 1 **Hacer**

**Si** cosechas**[**i**]>**promedio **Entonces**

mayorPromedio = mayorPromedio **+** 1**;**

**SiNo**

menorPromedio = menorPromedio **+** 1**;**

**Fin** **Si**

**Fin** **Para**

*//Imprimimos los resultados*

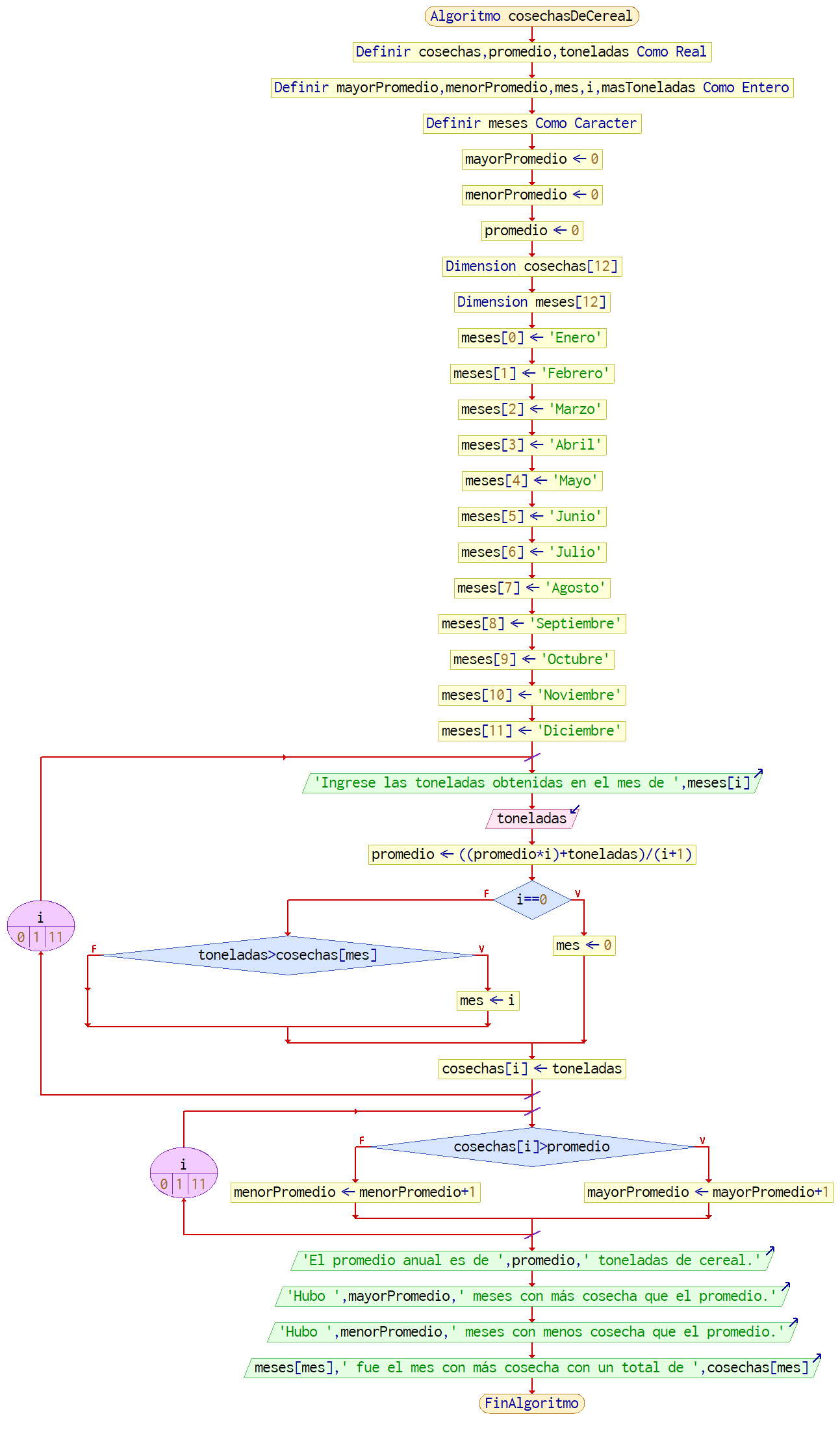
**Escribir** "El promedio anual es de "**,** promedio**,** " toneladas de cereal."**;**

**Escribir** "Hubo "**,** mayorPromedio**,** " meses con más cosecha que el promedio."**;**

**Escribir** "Hubo "**,** menorPromedio**,** " meses con menos cosecha que el promedio."**;**

**Escribir** meses**[**mes**],** " fue el mes con más cosecha con un total de "**,** cosechas**[**mes**];**

**FinAlgoritmo**



1. Realizar un algoritmo que dado un arreglo de N elementos, imprima la posición y el valor del elemento mayor almacenado en él. Suponga que todos los elementos del vector son diferentes. Represéntelo mediante el pseudocódigo y el diagrama de flujo.

**Algoritmo** mayorDeUnArreglo

*//Definimos variables*

**Definir** arreglo**,** n**,** posicionMayor**,** i **Como** **Entero;**

posicionMayor = 0**;**

*//Pedimos el tamano del arreglo*

**Escribir** "Ingresa el tamano del arreglo"**;**

**Leer** n**;**

**Dimension** arreglo**[**n**];**

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** **Sin** **Saltar** "Ingresa el valor del elemento "**,** i**+**1**;**

**Leer** arreglo**[**i**];**

**Si** i **==** 0 **Entonces**

posicionMayor = i**;**

**SiNo**

**Si** arreglo**[**i**]** **>** arreglo**[**posicionMayor**]** **Entonces**

posicionMayor = i**;**

**Fin** **Si**

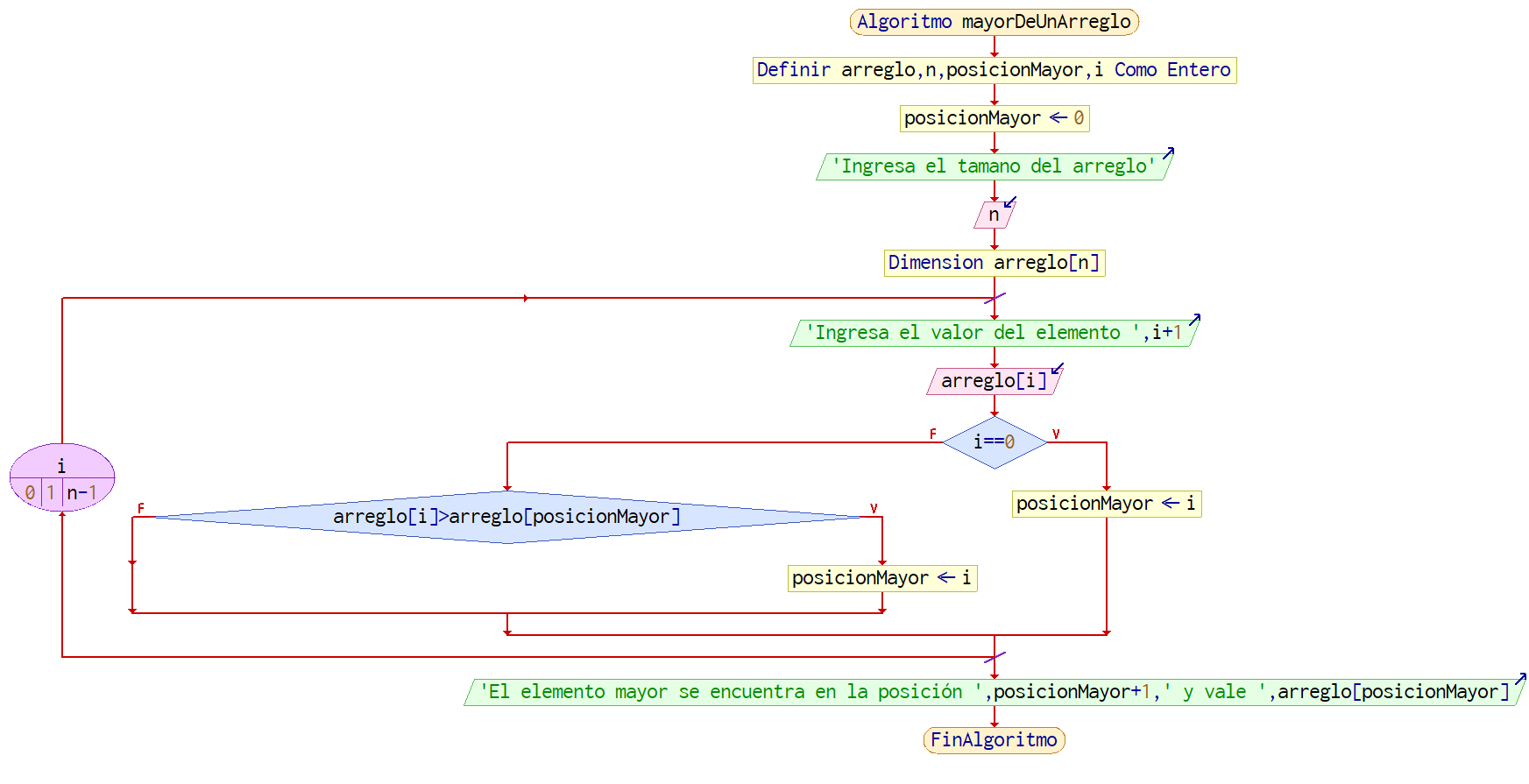
**Fin** **Si**

**Fin** **Para**

*//Escribimos la posicionMayor y lo que vale<*

**Escribir** "El elemento mayor se encuentra en la posición "**,** posicionMayor**+**1**,** " y vale "**,** arreglo**[**posicionMayor**];**

**FinAlgoritmo**



1. Realizar un algoritmo para calcular el promedio de N valores almacenados en un arreglo. Determinar además cuantos son mayores que el promedio, imprimir el promedio, el número de datos mayores que el promedio y proporcionar una lista de los valores que son mayores al promedio. Represente en pseudocódigo y diagrama de flujo.

**Algoritmo** mayorMenorPromedio

*//Definimos variables*

**Definir** arreglo**,** n**,** i**,** mayorPromedio**,** menorPromedio **Como** **Entero;**

**Definir** promedio **Como** **Real;**

mayorPromedio = 0**;**

menorPromedio = 0**;**

promedio = 0**;**

*//Pedimos el tamano del arreglo*

**Escribir** **Sin** **Saltar** "Ingresa el tamano del arreglo"**;**

**Leer** n**;**

**Dimension** arreglo**[**n**];**

*//Se llena el arreglo*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** **Sin** **Saltar** "Ingresa el valor para la posicion "**,**i **+** 1**,** ": "**;**

**Leer** arreglo**[**i**];**

promedio = promedio **+** arreglo**[**i**];**

**Fin** **Para**

*//calculamos el promedio*

promedio = promedio **/** n**;**

*//Contar los mayores del promedio e imprimilos*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Si** arreglo**[**i**]>**promedio **Entonces**

**Escribir** **Sin** **Saltar** arreglo**[**i**],** ", "**;**

mayorPromedio = mayorPromedio **+** 1**;**

**Fin** **Si**

**Fin** **Para**

**Si** mayorPromedio**>**1 **Entonces**

**Escribir** " son mayores que el promedio."**;**

**SiNo**

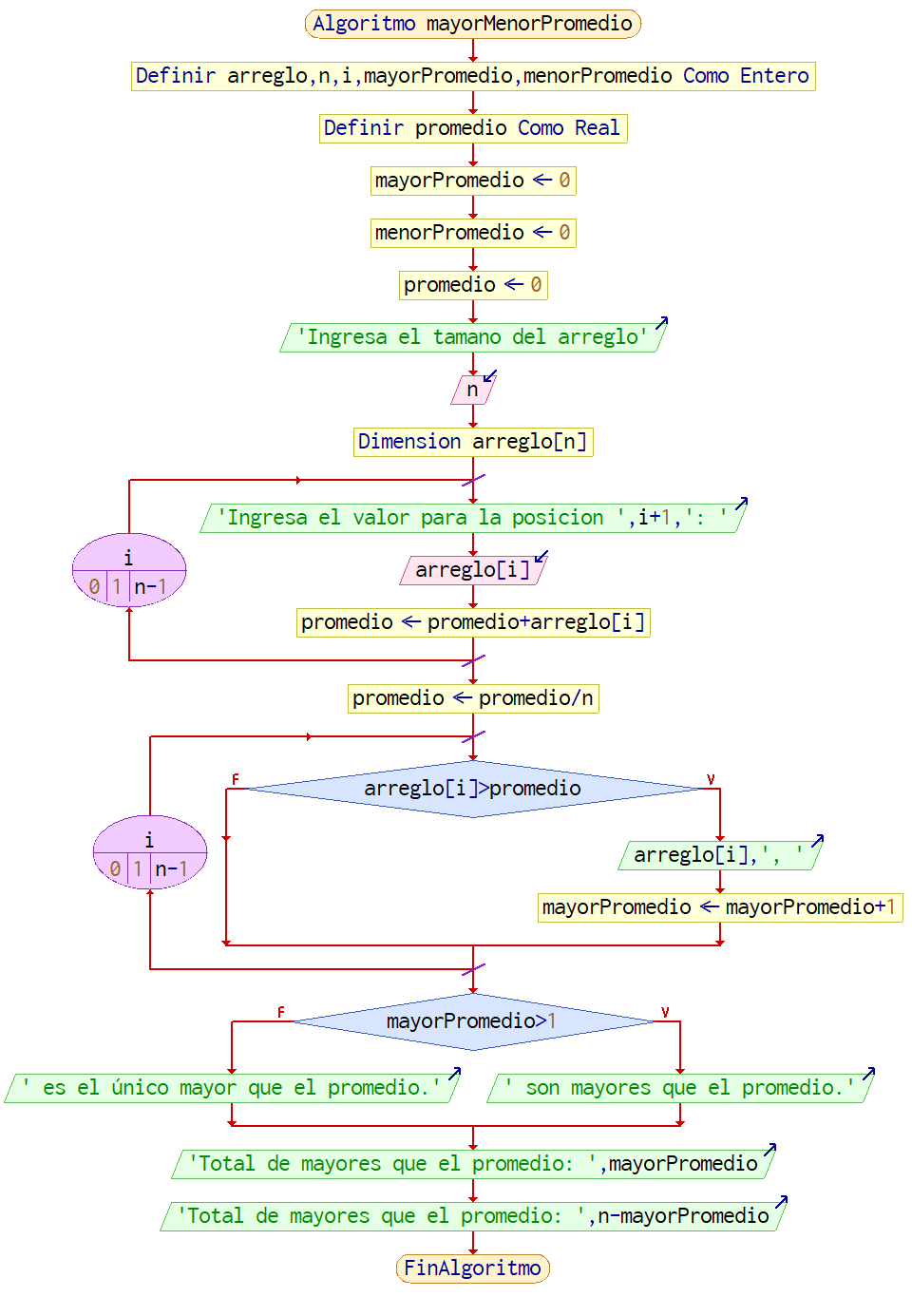
**Escribir** " es el único mayor que el promedio."**;**

**Fin** **Si**

**Escribir** "Total de mayores que el promedio: "**,** mayorPromedio**;**

**Escribir** "Total de mayores que el promedio: "**,** n**-**mayorPromedio**;**

**FinAlgoritmo**



1. Diseñe un algoritmo que lea dos arreglos A y B de N elementos cada uno y multiplique el primer elemento de A con el último elemento de B y luego el segundo elemento de A por el penúltimo elemento de B y así sucesivamente hasta llegar al último elemento de A por el primer elemento de B. El resultado de la multiplicación almacenarlo en un arreglo C. Represéntelo mediante el diagrama de flujo y el pseudocódigo.

**Algoritmo** abc

*//Definimos*

**Definir** A**,** B**,** C**,** N**,** i **Como** **Entero;**

*//Pedimos el tamano de los arreglos*

**Escribir** "Ingresa el tamano de los arreglos (A y B)."**;**

**Leer** N**;**

**Dimension** A**[**N**];**

**Dimension** B**[**N**];**

**Dimension** C**[**N**];**

*//se llena el arreglo A*

**Escribir** "Ingresa ahora los valores para el arreglo A"**;**

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** **Sin** **Saltar** "Ingresa el valor de la posición "**,** i **+** 1**,** ": "**;**

**Leer** A**[**i**];**

**Fin** **Para**

*//se llena el arreglo B*

**Escribir** "Ingresa ahora los valores para el arreglo B"**;**

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** **Sin** **Saltar** "Ingresa el valor de la posición "**,** i **+** 1**,** ": "**;**

**Leer** B**[**i**];**

**Fin** **Para**

*//se calculan los valores del arreglo C*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

C**[**i**]** = A**[**i**]** **\*** B**[**N**-**i**-**1**];**

**Fin** **Para**

*//Mostramos el arreglo C*

**Para** i = 0 **Hasta** n **-** 1 **Con** **Paso** 1 **Hacer**

**Escribir** "El valor de la posicion "**,** i **+** 1**,** " en el arreglo C es "**,** C**[**i**],**"."**;**

**Fin** **Para**

**FinAlgoritmo**

