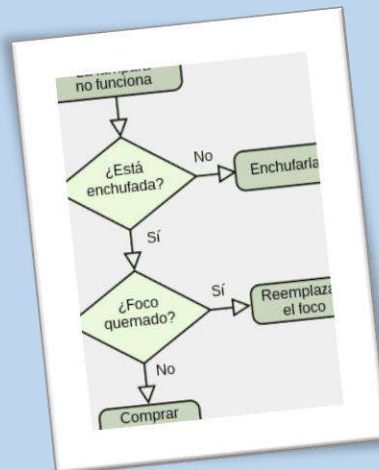




C5 Programación_Explorador_Virtual_L2

Oscar Ricardo Jurado Zambrano



PROFESOR: Miguel Eduardo Guerrero Ibarra

Monitora: Dayana hurtado.

ACTIVIDAD

FORO 11 – FASE ENTRENAMIENTO

San Juan de Pasto, 13 junio 2025

FORO 11 – FASE ENTRENAMIENTO

Estimados Campistas,

Bienvenidos a este espacio donde deberán montar la evidencia del trabajo realizado con respecto a los algoritmos en pseudocódigo en PSEint.

6 punto.

6. Revise el archivo adjunto con nombre Taller_Errores.psc encuentre sus errores, descríbalos, corrija los y presente la evidencia del código corregido

```
1  Algoritmo CodigoSinErrores
2      // Declaración de arreglos y variables
3      Dimension nombres[5]
4      Dimension edades[5]
5      Definir suma, i, n, j Como Entero;
6      Definir promedio, edad_mayor, edad_menor Como Real;
7      Definir respuesta, nombre_busqueda Como Cadena;
8      Definir encontrado Como Logico;
9
10     suma ← 0;
11     edad_mayor ← 0;
12     edad_menor ← 9999;
13
14     Escribir "Ingrese la cantidad de estudiantes (máximo 5):";
15     Leer n;
16
17     Si n > 5 Entonces
18         Escribir "Error: El máximo de estudiantes es 5";
19         n ← 5;
20     FinSi
21
22     Para i ← 1 Hasta n Hacer
23         Escribir "Ingrese el nombre del estudiante ", i, ":";
24         Leer nombres[i];
25         Escribir "Ingrese la edad del estudiante ", i, ":";
26         Leer edades[i];
27         suma ← suma + edades[i];
28
29         Si edades[i] > edad_mayor Entonces
30             edad_mayor ← edades[i];
31         FinSi
32
33         Si edades[i] < edad_menor Entonces
34             edad_menor ← edades[i];
35         FinSi
36     FinPara
37
38     promedio ← suma / n;
39
40     Escribir "El promedio de edades es: ", promedio;
41     Escribir "La edad mayor es: ", edad_mayor;
42     Escribir "La edad menor es: ", edad_menor;
43
44     Escribir "Listado de estudiantes mayores de edad:";
45     Para i ← 1 Hasta n Hacer
46         Si edades[i] ≥ 18 Entonces
47             Escribir nombres[i], " - ", edades[i];
48         FinSi
49     FinPara
50
51     Escribir "¿Desea buscar un estudiante? (si/no)";
52     Leer respuesta;
53
54     Si respuesta = "si" Entonces
55         Escribir "Ingrese el nombre a buscar:";
56         Leer nombre_busqueda;
```

```
54     Si respuesta = "si" Entonces
55         Escribir "Ingrese el nombre a buscar:";
56         Leer nombre_busqueda;
57         encontrado ← Falso;
58         Para j ← 1 Hasta n Hacer
59             Si nombres[j] = nombre_busqueda Entonces
60                 Escribir "Estudiante encontrado:";
61                 Escribir "Nombre: ", nombres[j];
62                 Escribir "Edad: ", edades[j];
63                 encontrado ← Verdadero;
64             FinSi
65         FinPara
66         Si NO encontrado Entonces
67             Escribir "No se encontró el estudiante.";
68         FinSi
69     FinSi
70
71     Escribir "Fin del programa.";
72
73 FinAlgoritmo
```

Lista de errores arrojada por PSEINT y sus respectivas correcciones

Taller_Errores (2).psc: Sintaxis Incorrecta: 12 errores.

Lin 3 (inst 1): ERROR 212: No debe utilizar subindices (NOMBRES).

Lin 3 (inst 1): ERROR 212: No debe utilizar subindices (EADADES).

Deben declararse como arreglos no como entero.

Corrección realizada.

Dimensión nombres[5]

Dimensión edades[5]

Lin 17 (inst 1): ERROR 256: La variable (EADADES) no es un arreglo.

Lin 18 (inst 1): ERROR 202: El identificador EADADES no corresponde a un arreglo o subproceso

Lin 18 (inst 1): ERROR 295: No coinciden los tipos (+). Los operandos deben ser de igual tipo.

Lin 19 (inst 1): ERROR 202: El identificador EADADES no corresponde a un arreglo o subproceso

Lin 20 (inst 1): ERROR 202: El identificador EADADES no corresponde a un arreglo o subproceso

Lin 22 (inst 1): ERROR 202: El identificador EADADES no corresponde a un arreglo o subproceso

Lin 35 (inst 1): ERROR 202: El identificador EADADES no corresponde a un arreglo o subproceso

Lin 36 (inst 1): ERROR 202: El identificador NOMBRES no corresponde a un arreglo o subproceso

Lin 47 (inst 1): ERROR 202: El identificador NOMBRES no corresponde a un arreglo o subproceso

Lin 48 (inst 1): ERROR 202: El identificador EADADES no corresponde a un arreglo o subproceso

Aunque el programa no arrojo este error, el promedio debe ser tipo real no entero, por eso lo coloco aca

Definir promedio Como Entero

Igualmente, estas variables no se encuentran declaradas, aquí hago la corrección.

Definir respuesta, nombre_busqueda Como Cadena;

Definir encontrado Como Lógico;

Estas variables pueden confundirse con palabras reservadas, por tanto, las cambie de

mayor <- 0

menor <- 100

Así quedaron corregidas.

edad_mayor <- 0;

edad_menor <- 9999;

Se adiciono este condicional antes del ciclo PARA para garantizar que el máximo de estudiantes sea de 5

```

Si n > 5 Entonces
    Escribir "Error: El máximo de estudiantes es 5";
    n ← 5;
FinSi

```

Dado que nombres y edades son variables de tipo arreglo, deben tener un subíndice. Este es el error

Código sin corrección

```

Para i = 1 Hasta n Hacer
    Escribir "Ingrese el nombre del estudiante ", i, ":"
    Leer nombres
    en inst. 1: Faltan subíndices para el arreglo (NOMBRES).
    Escribir "Ingrese la edad del estudiante ", i, ":"
    Leer edades[i]
    suma = suma + edades[i]
    Si edades[i] > mayor Entonces
        mayor = edades[i]
    FinSi
    Si edades[i] < menor
        menor = edades
    en inst. 1: Faltan subíndices para el arreglo (EDES).
FinPara

```

Código con la corrección

```

Para i ← 1 Hasta n Hacer
    Escribir "Ingrese el nombre del estudiante ", i, ":";
    Leer nombres[i];
    Escribir "Ingrese la edad del estudiante ", i, ":";
    Leer edades[i];
    suma ← suma + edades[i];

    Si edades[i] > edad_mayor Entonces
        edad_mayor ← edades[i];
    FinSi

    Si edades[i] < edad_menor Entonces
        edad_menor ← edades[i];
    FinSi
FinPara

```

En este condicional le falta la palabra reservado ENTONCES al final de las condiciones

```

Si respuesta == "si" Entonces
    Escribir "Ingrese el nombre a buscar:"
    Leer nombre_busqueda
    encontrado = falso
    Para j = 1 Hasta n
        Si nombres[j] = nombre_busqueda Entonces
            Escribir "Edad: ", edades[j]
            encontrado = verdadero
        FinSi
    FinPara
    Si encontrado = falso Entonces
        Escribir "No se encontró el estudiante."
    FinSi
FinSi

```

También falto imprimir el nombre del estudiante y decir que se ha encontrado el estudiante.

Código antes de la corrección

```

Si respuesta == "si"
    Escribir "Ingrese el nombre a buscar:"
    Leer nombre_busqueda
    encontrado = falso
    Para j = 1 Hasta n
        Si nombres[j] = nombre_busqueda
            Escribir "Edad: ", edades[j]
            encontrado = verdadero
        FinSi
    FinPara
    Si encontrado = falso
        Escribir "No se encontró el estudiante."
    FinSi
FinSi

```

Código después de la corrección

```

Si respuesta = "si" Entonces
    Escribir "Ingrese el nombre a buscar:";
    Leer nombre_busqueda;
    encontrado ← Falso;
    Para j ← 1 Hasta n Hacer
        Si nombres[j] = nombre_busqueda Entonces
            Escribir "Estudiante encontrado:";
            Escribir "Nombre: ", nombres[j];
            Escribir "Edad: ", edades[j];
            encontrado ← Verdadero;
        FinSi
    FinPara
    Si NO encontrado Entonces
        Escribir "No se encontró el estudiante.";
    FinSi
FinSi

```

1. Modificar la declaración de las listas, de tal manera que sea el usuario el que determine cuantos usuarios registrara.
2. Modificar el ciclo mientras principal por el ciclo hacer mientras, revise la lógica del programa.
3. Para la opción 2 modificar el ciclo PARA que permite la verificación de la existencia de un usuario y posteriormente la actualización de sus datos, de tal manera que finalice el ciclo una vez esto suceda.
4. Al igual que en la opción 2, modificar el ciclo PARA que permite la búsqueda de un usuario, de tal manera que una vez el usuario sea encontrado el ciclo finalice
5. En la opción 4 el menú que se presenta en pantalla a través de la opción mostrar, modificarlo de tal manera que pueda ser almacenado en una lista e impreso a través de un ciclo

Este es el código con lo solicitado

```

1  Algoritmo prueba
2  //DECLARACIÓN DE VARIABLES
3  Dimensionar menu[5]
4  definir op Como Entero
5  definir puede_registrar Como Caracter
6  definir encontrado Como Caracter
7
8  //Inicialización de la lista menu
9  Imprimir "
10 menu[1] + "
11 menu[2] + "
12 menu[3] + "
13 menu[4] + "
14 menu[5] + "
15 Imprimir "
16
17 Repetir
18 +   para i=1 hasta 5 Hacer
19     Imprimir menu[i]
20   FinPara
21   Imprimir "
22   Imprimir " <==> Elija la opción:==> "
23   leer op
24
25   puede_registrar = "si"
26   Segun op Hacer
27     1:

```

2. Modificar el ciclo mientras principal por el ciclo hacer mientras, revise la lógica del programa.

1. Modificar la declaración de las listas, de tal manera que sea el usuario el que determine cuantos usuarios registrara.

```
27 1:
28   Escribir "CUANTOS USUARIOS DESEA REGISTRAR: "
29   Leer N
30   Dimensionar cedula[N], nombres[N], apellidos[N], edad[N]
31
32
33   Imprimir "Digite la cedula del usuario a registrar: "
34   leer cedula_usuario
35   puede_registrar = "si"
36   //ciclo para --> VERIFICAR SI EL USUARIO EXISTE
37   para i ← 1 hasta N Hacer
38       si cedula_usuario = cedula[i] Entonces
39           puede_registrar = "no"
40       FinSi
41   FinPara
42
43   si puede_registrar = "si" Entonces
44       Imprimir "Digite los nombres del usuario a registrar: "
45       leer nombres_usuario
46       Imprimir "Digite los apellidos del usuario a registrar: "
47       leer apellidos_usuario
48       Imprimir "Digite la edad del usuario a registrar: "
49       leer edad_usuario
50       //Ciclo mientras --> verificar la posicion vacia para registrar
51       el usuario en cascada
52       i=1
53       a=0
```

Activar

```
54 Mientras i ≤ N Hacer
55     si cedula[i] = "" Entonces
56         cedula[i] = cedula_usuario
57         nombres[i] = nombres_usuario
58         apellidos[i] = apellidos_usuario
59         edad[i] = edad_usuario
60         a=i
61         Imprimir ""
62         imprimir "USUARIO REGISTRADO ",a
63         imprimir "cedula: ", cedula[i], " NOMBRES: ", nombres[i], " APELLIDOS: "
64         apellidos[i], " EDAD: ", edad[i]
65         Imprimir ""
66         //i=N+1
67     SI i<N Entonces
68         Imprimir "Digite la cedula del usuario a registrar: "
69         leer cedula_usuario
70         Imprimir "Digite los nombres del usuario a registrar: "
71         leer nombres_usuario
72         Imprimir "Digite los apellidos del usuario a registrar: "
73         leer apellidos_usuario
74         Imprimir "Digite la edad del usuario a registrar: "
75         leer edad_usuario
76     FinSi
77
78     FinSi
79     i=i+1
80 FIN MIENTRAS
81 o
82 imprimir ""
83 imprimir "EL USUARIO NO PUEDE REGISTRARSE"
84 imprimir ""
```

```

85 FinSi
86 // Mostrar todos los usuarios registrados
87 Imprimir "*****"
88 Imprimir "*** LISTADO DE TODOS LOS USUARIOS REGISTRADOS ***"
89 Imprimir "*****"
90
91 Para i ← 1 Hasta N Hacer
92     Si cedula[i] ≠ "" Entonces
93         Imprimir "*****"
94         Imprimir "Usuario #", i
95         Imprimir "Cédula: ", cedula[i]
96         Imprimir "Nombres: ", nombres[i]
97         Imprimir "Apellidos: ", apellidos[i]
98         Imprimir "Edad: ", edad[i]
99         Imprimir "*****"
100     FinSi
101 FinPara
102 2: //modificar este ciclo que permita terminar este ciclo
103 encontrado = "nok"
104 imprimir "DIGITE LA CEDULA DEL USUARIO AL CUAL SE ACTUALIZARA LOS DATOS: "
105 leer cedula_usuario_mod
106 i=1
107 Mientras i ≤ a Hacer
108     si cedula[i] = cedula_usuario_mod Entonces
109         Imprimir "Digite los nuevos nombres: "
110         leer nuevo_nombre
111         imprimir "Digite los nuevos apellidos: "
112         leer nuevos_apellidos
113         imprimir "Digite la nueva edad: "
114         leer nueva_edad
115

```

3. Para la opción 2 modificar el ciclo PARA que permite la verificación de la existencia de un usuario y posteriormente la actualización de sus datos, de tal manera que finalice el ciclo una vez esto suceda.

3. Para la opción 2 modificar el ciclo PARA que permite la verificación de la existencia de un usuario y posteriormente la actualización de sus datos, de tal manera que finalice el ciclo una vez esto suceda.

```

102 2: //modificar este ciclo que permita terminar este ciclo
103 encontrado = "nok"
104 imprimir "DIGITE LA CEDULA DEL USUARIO AL CUAL SE ACTUALIZARA LOS DATOS: "
105 leer cedula_usuario_mod
106 i=1
107 Mientras is a Hacer
108     si cedula[i] = cedula_usuario_mod Entonces
109         Imprimir "Digite los nuevos nombres: "
110         leer nuevo_nombre
111         imprimir "Digite los nuevos apellidos: "
112         leer nuevos_apellidos
113         imprimir "Digite la nueva edad: "
114         leer nueva_edad
115
116         nombres[i] = nuevo_nombre
117         apellidos[i] = nuevos_apellidos
118         edad[i] = nueva_edad
119         encontrado = "ok"
120         i=i+1
121     SiNo
122         i=i+1
123
124     FinSi
125
126 Fin Mientras
127
128 si encontrado = "nok" Entonces
129     imprimir ""
130     Imprimir "EL USUARIO NO SE ENCUENTRA REGISTRADO"
131     imprimir ""
132 FinSi
133
134 imprimir ""
135 imprimir "USUARIOS REGISTRADOS"
136 para i <-1 hasta a Hacer
137     imprimir "cedula: ", cedula[i], " NOMBRES: ", nombres[i], " APELLIDOS: ", apellidos[i], " EDAD: ", edad[i]
138
139 FinPara
  
```

```

139     imprimir ""
140     Imprimir "EL USUARIO NO SE ENCUENTRA REGISTRADO"
141     imprimir ""
142 FinSi
143
144 imprimir ""
145 imprimir "USUARIOS REGISTRADOS"
146 para i <-1 hasta a Hacer
147     imprimir "cedula: ", cedula[i], " NOMBRES: ", nombres[i], " APELLIDOS: ", apellidos[i], " EDAD: ", edad[i]
148
149 FinPara
150 3:
  
```


4. Al igual que en la opción 2, modificar el ciclo PARA que permite la búsqueda de un usuario, de tal manera que una vez el usuario sea encontrado el ciclo finalice

```
150 3:
151     encontrado = "nok"
152     imprimir "DIGITE LA CEDULA DEL USUARIO A BUSCAR: "
153     leer cedula_usuario
154
155     para i = 1 hasta N Hacer
156         si cedula_usuario = cedula[i] Entonces
157             imprimir "<<=====>>"
158             imprimir "EL USUARIO SE ENCUENTRA REGISTRADO"
159             imprimir "<=====>"
160             imprimir "cedula: ", cedula[i], " NOMBRES: ", nombres[i], " APELLIDOS: ", apellidos[i], " EDAD: ", edad[i]
161             encontrado = "ok"
162         FinSi
163     FinPara
164
165     si encontrado = "nok" Entonces
166         imprimir ""
167         imprimir "EL USUARIO NO SE ENCUENTRA REGISTRADO"
168         imprimir ""
169     FinSi
170 4:
```

```
170 4:
171     encontrado = "nok"
172     imprimir "Cedula del pasajero: "
173     leer cedula_pasajero
174     b=0
175     i = 1
176     //nuevo estoy corrigiendolo
177     Mientras i≤N
178         SI cedula[i] ≠ cedula_pasajero Entonces
179             encontrado = "nok"
180             Imprimir "USUARIO", b, "NO ENCONTRADO"
181             b=b+i
182             i=i+1
183         FinSi
184     FIN MIENTRAS
185
186
187     mientras i ≤ N Hacer
188
189         si cedula[i] = cedula_pasajero Entonces
```

[illegible]

```

214         opcion 3:
215             imprimir "000 <<=====>> OPCIÓN NO VALIDA <<===== "
216             imprimir "I<^>^<^>^<^>^<^> HASTA LA PROXIMA <^>^<^>^<^>I"
217             imprimir "000 <<=====>> OPCIÓN NO VALIDA <<===== "
218         De Otro Modo:
219             imprimir "000 <<=====>> OPCIÓN NO VALIDA <<===== "
220             imprimir " =====>> OPCIÓN NO VALIDA <<===== "
221             imprimir "000 <<=====>> OPCIÓN NO VALIDA <<===== "
222     Fin Segun
223
224     i = N
225   FinSi
226   i = i + 1
227 inMientras
228
229 si encontrado = "nok" Entonces
230     imprimir "EL USUARIO NO SE ENCUENTRA REGISTRADO - NO PUEDE COMPRAR TIQUETES"
231     imprimir ""
232 siNo
233     imprimir "USUARIO ENCONTRADO"
234     imprimir ""
235
236 Segun opcion_viaje Hacer
237     opcion 1:
238         imprimir "Y TIENE UN VIAJE PENDIENTE A ESPAÑA"
239         imprimir ""
240     opcion 2:

```

```

240         opcion 2:
241             imprimir "Y TIENE UN VIAJE PENDIENTE A INGLATERRA"
242             imprimir ""
243         opcion 3:
244             imprimir "Y NO COMPRO NINGUN TIQUETE"
245             imprimir ""
246         De Otro Modo:
247             imprimir "Y TOMO UNA OPCIÓN NO VALIDA"
248             imprimir ""
249         Fin Segun
250     FinSi
251
252     De Otro Modo:
253         mostrar "OPCIÓN INCORRECTA, ESCOJA ENTRE 1 A 5"
254
255     Fin Segun
256
257     //IMPRIMIENDO NUEVAMENTE EL MENU PARA ACTUALIZAR LA VARIABLE OP
258     para i←1 hasta 5 Hacer
259         Imprimir menu[i]
260     FinPara
261     Imprimir "Elija la opción: "
262     leer op
263
264     FinMientras
265     FinAlgoritmo

```