

FUNDAMENTOS DE PROGRAMACIÓN
LABORATORIO 4
PROPUESTAS DE SOLUCIÓN
SEMESTRE ACADÉMICO 2022-2

Horarios: Todos los horarios

Elaborado por Mag. Sergio Ponce

INDICACIONES:

- Debe utilizar variables descriptivas, comentarios y mensajes descriptivos.
- El orden y la eficiencia de su implementación serán considerados en la calificación.

RESULTADOS ESPERADOS:

- Al finalizar la sesión, el alumno construirá programas usando programación modular usando paso de parámetro por valor y simulando el paso de parámetros por referencia

CONSIDERACIONES:

- La solución presentada para cada problema corresponde a una propuesta de solución por parte del autor.
- En programación pueden existir muchas soluciones para un mismo problema pero debe cumplir con todo lo solicitado, incluyendo las restricciones brindadas.

Desarrolle los siguientes problemas en lenguaje C:

1. Sumatorias - Horarios: 0386, 0387 y 0390

Se le pide implementar un programa en lenguaje C que muestre un menú de opciones al usuario para que pueda elegir una de ellas y calcular una determinada sumatoria:

Si el usuario ingresa la opción A debe realizar el siguiente cálculo:

$$\sum_{i=0}^n \frac{a^{i+1}}{b(2i+1)!} + \frac{b^\pi}{(3a-1)(2i)!}$$

Figura 1: Sumatoria 1

Si el usuario ingresa la opción B debe realizar el siguiente cálculo:

$$\sum_{i=0}^n \frac{(a-5)i!}{(2a)^i(5i+2)} + \frac{a^i b^{i+1}}{(4-b)}$$

Figura 2: Sumatoria 2

Donde:

- La cantidad de términos está representada por la n cuyo rango es $1 \leq n \leq 6$.
- a es un número entre [2 y 5].
- b es un número entre [1 y 3].

Tener en cuenta que para el desarrollo del programa debe mostrar mensajes específicos ante las siguientes situaciones:

- Al ingresar la opción debe verificar que sea A o B (en mayúscula). En caso no se cumpla, se deberá emitir el siguiente mensaje “La opción ingresada no es válida.” y el programa debe terminar.
- Debe validar que el usuario ingrese datos correctos para las variables n , a y b caso contrario, debe mostrar un mensaje de error por cada uno de los datos que se ingrese de forma incorrecta y el programa debe terminar (ver casos de prueba).
- Considere que primero debe ingresar todos los datos pedidos y luego realizar las validaciones solicitadas.

Debe implementar únicamente los siguientes módulos:

- Un módulo que solicite leer los siguientes datos: opción para escoger sumatoria, valor de n , valor de a y valor de b .
- Un módulo que valide los datos de entrada y retorne si la opción ingresada es válida, si el valor de n es válido, si el valor de a es válido y si el valor de b es válido.
- Un módulo que calcule y retorne el valor de la sumatoria 1.
- Un módulo que calcule y retorne el valor de la sumatoria 2.

Nota:

- Considerar $\pi = 3.1415926536$

Casos de prueba

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
Z  
Ingrese la cantidad de iteraciones (n): 5  
Ingrese el valor de a: 2  
Ingrese el valor de b: 3  
La opción ingresada no es valida
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
a  
Ingrese la cantidad de iteraciones (n): 6  
Ingrese el valor de a: 3  
Ingrese el valor de b: 2  
La opción ingresada no es valida
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
A  
Ingrese la cantidad de iteraciones (n): 10  
Ingrese el valor de a: 5  
Ingrese el valor de b: 1  
El valor de N no esta en el rango solicitado
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
B  
Ingrese la cantidad de iteraciones (n): 8  
Ingrese el valor de a: 12  
Ingrese el valor de b: 5  
El valor de N no esta en el rango solicitado.  
El valor de a no esta en el rango solicitado.  
El valor de b no esta en el rango solicitado
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
B  
Ingrese la cantidad de iteraciones (n): 5  
Ingrese el valor de a: 1  
Ingrese el valor de b: 5  
El valor de a no esta en el rango solicitado  
El valor de b no esta en el rango solicitado
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
B  
Ingrese la cantidad de iteraciones (n): 5  
Ingrese el valor de a: 2  
Ingrese el valor de b: 5  
El valor de b no esta en el rango solicitado
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
B  
Ingrese la cantidad de iteraciones (n): 4  
Ingrese el valor de a: 4  
Ingrese el valor de b: 2  
El resultado de la sumatoria es: 4680.479
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1
```

```

B: Sumatoria 2
A
Ingrese la cantidad de iteraciones (n): 1
Ingrese el valor de a: 2
Ingrese el valor de b: 1
El resultado de la sumatoria es: 2.967

```

```

Ingrese la opción correspondiente a la sumatoria que desea calcular:
A: Sumatoria 1
B: Sumatoria 2
A
Ingrese la cantidad de iteraciones (n): 6
Ingrese el valor de a: 2
Ingrese el valor de b: 3
El resultado de la sumatoria es: 10.647

```

Programa 1: Propuesta de solución - Sumatorias

```

1 #include <stdio.h>
2 #include <math.h>
3 #define PI 3.1415926536
4
5 /*Prototipos*/
6 void leerDatos(char *, int *, int *, int *);
7 double calcularSumatoria1(int, int, int);
8 double calcularSumatoria2(int, int, int);
9 void validarDatosEntrada(char, int ,int ,int *, int *,int *,int *);
10
11 int main() {
12     int validarN, validarA, validarB, validarOpcion, n, a, b;
13     char opcion;
14     double sumatoria;
15
16     leerDatos(&opcion, &n, &a, &b);
17     validarDatosEntrada(opcion, n,a,b, &validarOpcion, &validarN,&validarA,&validarB);
18
19     /*Validamos las variables por separado para mandar mensajes de error diferentes*/
20     if (!validarOpcion) {
21         printf("La opción ingresada no es válida");
22     } else if (!validarN || !validarA || !validarB) {
23         if (!validarN) {
24             printf("El valor de N no está en el rango solicitado\n");
25         }
26         if (!validarA) {
27             printf("El valor de a no está en el rango solicitado\n");
28         }
29         if (!validarB) {
30             printf("El valor de b no está en el rango solicitado\n");
31         }
32     } else {
33         if (opcion=='A') {
34             sumatoria=calcularSumatoria1(n,a,b);
35         } else {
36             sumatoria=calcularSumatoria2(n,a,b);
37         }
38         printf("El resultado de la sumatoria es: %0.3lf", sumatoria);
39     }
40     return 0;
41 }
42 void leerDatos(char *opcion,int *n, int *a, int *b) {
43     printf("Ingrese la opción correspondiente a la sumatoria que desea calcular:\n");
44     printf("A: Sumatoria 1\n");

```

```

45     printf("B: Sumatoria 2\n");
46     scanf(" %c",opcion);
47     printf("Ingrese la cantidad de iteraciones (n): ");
48     scanf("%d",n);
49     printf("Ingrese el valor de a: ");
50     scanf("%d",a);
51     printf("Ingrese el valor de b: ");
52     scanf("%d",b);
53 }
54
55 double calcularSumatoria1(int n, int a, int b) {
56     int i=0,factorialImpares=1,factorialPares=1;
57     double sumatoria=0, termino1, termino2;
58     double numerador1, numerador2, denominador1, denominador2;
59     while (i<=n) {
60         /*los factoriales pares e impares se trabajan en variables separadas*/
61         if (i>0) {
62             factorialImpares=(2*i+1)*(2*i)*factorialImpares;
63             factorialPares=(2*i)*(2*i-1)*factorialPares;
64         }
65         numerador1=pow(a,i+1);
66         denominador1=b*factorialImpares;
67         termino1=numerador1/denominador1;
68
69         numerador2=pow(b,PI);
70         denominador2=(3*a-1)*factorialPares;
71         termino2=numerador2/denominador2;
72
73         sumatoria=sumatoria+termino1+termino2;
74         i++;
75     }
76     return sumatoria;
77 }
78
79 void validarDatosEntrada(char opcion, int n,int a,int b,int *validarOpcion, int *validarN,int *validarA,int *validarB) {
80     *validarOpcion=opcion=='A' || opcion=='B';
81     *validarN=n>=1 && n<=6;
82     *validarA=a>=2 && a<=5;
83     *validarB=b>=1 && b<=3;
84 }
85
86 double calcularSumatoria2(int n, int a, int b) {
87     int i=0,factorial=1;
88     double sumatoria=0, termino1,termino2;
89     double numerador1, denominador1, numerador2,denominador2;
90     while (i<=n) {
91         /*Considerar el factorial de 0 */
92         if (i>0) {
93             factorial=factorial*i;
94         }
95         numerador1=(a-5)*factorial;
96         denominador1=pow(2*a,i)*(5*i+2);
97         termino1=numerador1/denominador1;
98
99         numerador2=pow(a,i)*pow(b,i+1);
100        denominador2=4-b;
101        termino2=numerador2/denominador2;
102
103        sumatoria=sumatoria+termino1+termino2;
104        i++;
105    }
106 }

```

2. Sumatorias - Horarios: B301, 0384, 0385 y 0391

Se le pide implementar un programa en lenguaje C que muestre un menú de opciones al usuario para que pueda elegir una de ellas y calcular una determinada sumatoria:

Si el usuario ingresa la opción A, también podría ingresar la letra en minúscula, debe realizar el siguiente cálculo:

$$\sum_{i=0}^n \frac{(-1)^{i+2} 2(2^{2i-1} - 1)a^{2i-1}}{b(i!)^2} + \frac{b}{(a-1)}$$

Figura 3: Sumatoria 1

Si el usuario ingresa la opción B, también podría ingresar la letra en minúscula, debe realizar el siguiente cálculo:

$$\sum_{i=0}^n \frac{(2i)! b a^{2i+1}}{(2i+1)!} a^{2i+1} + a(i!)$$

Figura 4: Sumatoria 2

Donde:

- La cantidad de términos está representada por la n cuyo rango es $1 \leq n \leq 6$.
- a es un número entre [2 y 5].
- b es un número entre [1 y 6].

Tener en cuenta que para el desarrollo del programa debe mostrar mensajes específicos ante las siguientes situaciones:

- Al ingresar la opción debe verificar que sea A o B o a o b. En caso no se cumpla, se deberá emitir el siguiente mensaje “La opción ingresada no es válida.” y el programa debe terminar.
- Debe validar que el usuario ingrese datos correctos para las variables n , a y b caso contrario, debe mostrar un mensaje de error por cada uno de los datos que se ingrese de forma incorrecta y el programa debe terminar (ver casos de prueba).
- Considere que primero debe ingresar todos los datos pedidos y luego realizar las validaciones solicitadas.

Debe implementar únicamente los siguientes módulos:

- Un módulo que solicite leer los siguientes datos: opción para escoger sumatoria, valor de n , valor de a y valor de b .
- Un módulo que valide los datos de entrada y retorne si la opción ingresada es válida, si el valor de n es válido, si el valor de a es válido y si el valor de b es válido.
- Un módulo que calcule y retorne el valor de la sumatoria 1.
- Un módulo que calcule y retorne el valor de la sumatoria 2.

Casos de prueba

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
Z  
Ingrese la cantidad de iteraciones (n): 5  
Ingrese el valor de a: 2  
Ingrese el valor de b: 3  
La opción ingresada no es valida
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
a  
Ingrese la cantidad de iteraciones (n): 5  
Ingrese el valor de a: 3  
Ingrese el valor de b: 3  
El resultado de la sumatoria es: -245.291
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
a  
Ingrese la cantidad de iteraciones (n): 10  
Ingrese el valor de a: 3  
Ingrese el valor de b: 3  
El valor de N no esta en el rango solicitado
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
A  
Ingrese la cantidad de iteraciones (n): 6  
Ingrese el valor de a: 10  
Ingrese el valor de b: -1  
El valor de a no esta en el rango solicitado  
El valor de b no esta en el rango solicitado
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1  
B: Sumatoria 2  
B  
Ingrese la cantidad de iteraciones (n): 10  
Ingrese el valor de a: 5  
Ingrese el valor de b: 0  
El valor de N no esta en el rango solicitado  
El valor de b no esta en el rango solicitado
```

```
Ingrese la opcion correspondiente a la sumatoria que desea calcular:  
A: Sumatoria 1
```

```
B: Sumatoria 2
A
Ingrese la cantidad de iteraciones (n): 1
Ingrese el valor de a: 2
Ingrese el valor de b: 3
El resultado de la sumatoria es: 4.500
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:
A: Sumatoria 1
B: Sumatoria 2
A
Ingrese la cantidad de iteraciones (n): 5
Ingrese el valor de a: 4
Ingrese el valor de b: 3
El resultado de la sumatoria es: -4303.286
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:
A: Sumatoria 1
B: Sumatoria 2
b
Ingrese la cantidad de iteraciones (n): 1
Ingrese el valor de a: 2
Ingrese el valor de b: 3
El resultado de la sumatoria es: 26.000
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:
A: Sumatoria 1
B: Sumatoria 2
b
Ingrese la cantidad de iteraciones (n): 2
Ingrese el valor de a: 3
Ingrese el valor de b: 3
El resultado de la sumatoria es: 3574.200
```

```
Ingrese la opción correspondiente a la sumatoria que desea calcular:
A: Sumatoria 1
B: Sumatoria 2
A
Ingrese la cantidad de iteraciones (n): 1
Ingrese el valor de a: 5
Ingrese el valor de b: 6
El resultado de la sumatoria es: 1.300
```

```
Ingrese la opcion correspondiente a la sumatoria que desea calcular:
A: Sumatoria 1
B: Sumatoria 2
b
Ingrese la cantidad de iteraciones (n): 2
Ingrese el valor de a: 2
Ingrese el valor de b: 3
El resultado de la sumatoria es: 490.800
```

Programa 2: Propuesta de solución - Sumatorias

```
1 #include <stdio.h>
2 #include <math.h>
3
4 void leerDatos(char *, int *, int *, int *);
5 double calcularSumatoria1(int, int, int);
6 double calcularSumatoria2(int, int, int);
7 void validarDatosEntrada(char, int ,int ,int *, int *,int *,int *);
8
9 int main() {
10     int validarN, validarA, validarB, validarOpcion, n, a, b;
11     char opcion;
12     double sumatoria;
13
14     leerDatos(&opcion, &n, &a, &b);
15     validarDatosEntrada(opcion, n,a,b, &validarOpcion, &validarN,&validarA,&validarB);
16
17     /*Validamos las variables por separado para mandar mensajes de error diferentes*/
18     if (!validarOpcion) {
19         printf("La opción escogida no es válida");
20     } else if (!validarN || !validarA || !validarB) {
21         if (!validarN) {
22             printf("El valor de N no está en el rango solicitado\n");
23         }
24         if (!validarA) {
25             printf("El valor de a no está en el rango solicitado\n");
26         }
27         if (!validarB) {
28             printf("El valor de b no está en el rango solicitado\n");
29         }
30     } else {
31         if (opcion=='A' || opcion=='a') {
32             sumatoria=calcularSumatoria1(n,a,b);
33         } else {
34             sumatoria=calcularSumatoria2(n,a,b);
35         }
36         printf("El resultado de la sumatoria es: %0.3lf", sumatoria);
37     }
38     return 0;
39 }
40 void leerDatos(char *opcion,int *n, int *a, int *b) {
41     printf("Ingrese la opción correspondiente a la sumatoria que desea calcular:\n");
42     printf("A: Sumatoria 1\n");
43     printf("B: Sumatoria 2\n");
44     scanf(" %c",opcion);
45     printf("Ingrese la cantidad de iteraciones (n): ");
46     scanf("%d",n);
47     printf("Ingrese el valor de a: ");
48     scanf("%d",a);
49     printf("Ingrese el valor de b: ");
50     scanf("%d",b);
51 }
52
53 double calcularSumatoria1(int n, int a, int b) {
54     int i=0,factorial=1;
55     double sumatoria=0, termino1, termino2;
56     double numerador1, numerador2, denominador1, denominador2;
57     while (i<=n) {
58         /*Considerar el factorial de 0 */
59         if (i>0) {
60             factorial=factorial*i;
61         }
62         numerador1=pow(-1,i+2)*2*(pow(2,2*i-1)-1)*pow(a,2*i-1);
63         denominador1=b*pow(factorial,2);
64         termino1=numerador1/denominador1;
65     }
```

```

66     numerador2=b;
67     denominador2=a-1;
68     termino2=numerador2/denominador2;
69
70     sumatoria=sumatoria+termino1+termino2;
71     i++;
72 }
73 return sumatoria;
74 }

75 void validarDatosEntrada(char opcion, int n,int a,int b,int *validarOpcion, int *validarN,int *validarA,int *validarB) {
76     *validarOpcion=opcion=='A' || opcion=='a' || opcion=='B' || opcion=='b';
77     *validarN=n>=1 && n<=6;
78     *validarA=a>=2 && a<=5;
79     *validarB=b>=3 && b<=6;
80 }
81

83 double calcularSumatoria2(int n, int a, int b) {
84     int i=0,factorialImpares=1,factorialPares=1,factorial=1;
85     double sumatoria=0, termino1,termino2, numerador1, denominador1;
86     while (i<=n) {
87         if (i>0) {
88             /*los factoriales pares e impares se trabajan en variables separadas*/
89             factorial=factorial*i;
90             factorialImpares=(2*i+1)*(2*i)*factorialImpares;
91             factorialPares=(2*i)*(2*i-1)*factorialPares;
92         }
93         numerador1=factorialPares*b*pow(a,2*i+1);
94         denominador1=2*i+1;
95         termino1=numerador1/denominador1;
96
97         termino2=a*factorial;
98         sumatoria=sumatoria+termino1+termino2;
99         i++;
100    }
101 }

```

3. Límites de velocidad - Horarios: 0380, 0381, 0388 y 0389

En una ciudad se tienen las calles A, B, C, D, E, F, G y H, para las cuales se han establecido los siguientes límites de velocidad según la siguiente tabla (considerar que son las únicas calles que existen):

Nombre de las calles	Límite de velocidad (Km/h)
A, B, C	30
D, E, F	35
G, H	50

Tabla: Límites de velocidad

Se le pide que implemente un programa en lenguaje C que solicite al usuario la cantidad de calles que recorrió y devuelva un resumen con: La distancia total recorrida, La cantidad de calles donde excedió la velocidad permitida, La cantidad de calles donde no excedió la velocidad permitida, El monto total de la multa que debe pagar; además, deberá indicar la cantidad de calles que no se han tomado en cuenta para el resumen debido a que ingresaron algún dato incorrecto.

Para la implementación de su solución debe considerar lo siguiente:

- Debe validar que la cantidad de calles ingresada por el usuario debe ser mayor que 0 y menor o igual que 5. En caso la cantidad de calles ingresada no sea correcta, se deberá emitir el siguiente mensaje “La cantidad de calles no es correcta.” y el programa debe terminar.

- Para cada calle ingresada deberá solicitar: El nombre de la calle, la distancia recorrida en kilómetros, el tiempo que le tomó desplazarse (en segundos). En caso alguno de los datos ingresados no sea correcto deberá emitir el siguiente mensaje “Uno de los datos ingresados no es correcto, por lo que esta calle no se tomará en cuenta para el resumen de resultados.”. Si existen más calles por procesar, el programa debe continuar su ejecución.
- Excederse hasta 10 Km/h por encima del límite de velocidad de una calle, conlleva a una sanción del 20 % de la Unidad Impositiva Tributaria (UIT) y excederse más de 10 Km/h por encima del límite de velocidad de una calle, conlleva a una sanción del 40 % de la Unidad Impositiva Tributaria (UIT). Considerar una UIT= \$/ 4600.

Debe implementar únicamente los siguientes módulos:

- Un módulo que solicite leer los siguientes datos: Nombre de la calle, Distancia recorrida y el tiempo empleado.
- Un módulo que valide si los tres datos ingresados para una calle (Distancia recorrida, Tiempo empleado y Nombre de la calle) son correctos.
- Un módulo que calcule la multa a pagar en una calle. Considerar que si no se excedió el límite de velocidad de una calle, el módulo debe devolver 0.
- Un módulo que imprima el resumen de los resultados (ver casos de prueba para validar los escenarios en donde no se debe imprimir el resumen de resultados).

Nota:

Para la comparación de las velocidades (debido a la cantidad de decimales que podría tener), considerar que una velocidad es mayor que otra si su diferencia es mayor que 0.001.

Casos de prueba

```
Ingrese la cantidad de calles por las que paso: -3
La cantidad de calles no es correcta
```

```
Ingrese la cantidad de calles por las que paso: 10
La cantidad de calles no es correcta
```

```
Ingrese la cantidad de calles por las que paso: 1
Para la calle 1
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H): a
Ingrese la distancia recorrida en kilómetros: 100
Ingrese el tiempo en segundos: 3600
Uno de los datos ingresados en la calle 1 no es correcto, no se tomará en cuenta para el resumen de resultados.
```

```
La única calle que ingreso no tiene datos correctos, no se puede mostrar el resumen.
```

```
Ingrese la cantidad de calles por las que paso: 2
```

Para la calle 1
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H) : A
Ingrese la distancia recorrida en kilómetros: 60
Ingrese el tiempo en segundos: 7200
En la calle A no excedió el límite de velocidad

Para la calle 2
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H) : b
Ingrese la distancia recorrida en kilómetros: 100
Ingrese el tiempo en segundos: 3600
Uno de los datos ingresados en la calle 2 no es correcto, no se tomará en cuenta para el resumen de resultados.

RESUMEN

Distancia total recorrida es: 60.00Km
Cantidad de calles con algún dato incorrecto: 1
Cantidad de calles donde excedió la velocidad permitida: 0
Cantidad de calles donde NO excedió la velocidad permitida: 1
Monto total a pagar es: S/.0.00

Ingrese la cantidad de calles por las que paso: 3

Para la calle 1
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H) : A
Ingrese la distancia recorrida en kilómetros: 60
Ingrese el tiempo en segundos: 7200
En la calle A no excedió el límite de velocidad

Para la calle 2
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H) : B
Ingrese la distancia recorrida en kilómetros: 100
Ingrese el tiempo en segundos: 7500
En la calle B se excedió el límite de velocidad

Para la calle 3
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H) : F
Ingrese la distancia recorrida en kilómetros: 120
Ingrese el tiempo en segundos: 14000
En la calle F no excedió el límite de velocidad

RESUMEN

Distancia total recorrida es: 280.00Km
Cantidad de calles con algún dato incorrecto: 0
Cantidad de calles donde excedió la velocidad permitida: 1
Cantidad de calles donde NO excedió la velocidad permitida: 2
Monto total a pagar es: S/.1840.00

Ingrese la cantidad de calles por las que paso: 2

Para la calle 1
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H) : A
Ingrese la distancia recorrida en kilómetros: 60
Ingrese el tiempo en segundos: 7200

```

En la calle A no excedió el límite de velocidad

Para la calle 2
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H): C
Ingrese la distancia recorrida en kilómetros: 40
Ingrese el tiempo en segundos: 7000
En la calle C no excedió el límite de velocidad

RESUMEN
Distancia total recorrida es: 100.00Km
Cantidad de calles con algún dato incorrecto: 0
Cantidad de calles donde excedió la velocidad permitida: 0
Cantidad de calles donde NO excedió la velocidad permitida: 2
Monto total a pagar es: S/.0.00

```

```

Ingrese la cantidad de calles por las que paso: 3

Para la calle 1
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H): D
Ingrese la distancia recorrida en kilómetros: 150
Ingrese el tiempo en segundos: 3000
En la calle D se excedió el límite de velocidad

Para la calle 2
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H): A
Ingrese la distancia recorrida en kilómetros: 120
Ingrese el tiempo en segundos: 7000
En la calle A se excedió el límite de velocidad

Para la calle 3
Ingrese el nombre de la calle (A,B,C,D,E,F,G,H): E
Ingrese la distancia recorrida en kilómetros: 200
Ingrese el tiempo en segundos: 14500
En la calle E se excedió el límite de velocidad

RESUMEN
Distancia total recorrida es: 470.00Km
Cantidad de calles con algún dato incorrecto: 0
Cantidad de calles donde excedió la velocidad permitida: 3
Cantidad de calles donde NO excedió la velocidad permitida: 0
Monto total a pagar es: S/.5520.00

```

Programa 3: Propuesta de solución - Límites de velocidad

```

1 #include <stdio.h>
2 #define UIT 4600
3 #define LIMITE_A_B_C 30
4 #define LIMITE_D_E_F 35
5 #define LIMITE_G_H 50
6 #define DIF_VEL 0.001
7 #define MONTO1 0.20
8 #define MONTO2 0.40
9
10 void leerDatos(double *,double *,char *,int);
11 int validarDatos(double, double, char);
12 double calcularMulta(char, double);

```

```

13 void mostrarResumenResultados(double, double, int, int, int);
14
15 int main() {
16     int cantCalles, cont=1, calleMultas=0, callesIncorrectas=0, datosValidos, calleOk=0;
17     double velocidad, tiempo, distancia, montoTotalMultas=0, distanciaTotal=0, montoMultas;
18     char calle;
19     printf("Ingrese la cantidad de calles por las que pasó: ");
20     scanf("%d", &cantCalles);
21
22     if (cantCalles>0 && cantCalles<=5) {
23         /*implementamos la lógica solicitada*/
24         while (cont<=cantCalles) {
25             leerDatos(&distancia, &tiempo, &calle, cont);
26             datosValidos=validarDatos(distancia, tiempo, calle);
27             if (datosValidos) { /*Solo si todos los datos son válidos trabajamos con esa calle*/
28                 velocidad=distancia/(tiempo/3600);
29                 /*determinamos si hay multa en la calle transitada, si no hay multa
30                 montoMultas=0*/
31                 montoMultas=calcularMultas(calle, velocidad);
32                 if (montoMultas!=0) {
33                     /*hay multa*/
34                     printf("En la calle %c se excedió el límite de velocidad\n", calle);
35                     calleMultas++;
36                 } else {
37                     printf("En la calle %c no excedió el límite de velocidad\n", calle);
38                     calleOk++;
39                 }
40                 /*acumulamos los datos solicitados*/
41                 montoTotalMultas=montoTotalMultas+montonMultas;
42                 distanciaTotal=distanciaTotal+distancia;
43             } else {
44                 printf("Uno de los datos ingresados en la calle %d no es correcto, no se tomará en cuenta para el
45                 resumen de resultados.\n", cont);
46                 callesIncorrectas++;
47             }
48             cont++;
49         }
50         if (cantCalles==1 && callesIncorrectas==1) {
51             printf("\nLa única calle que ingresó no tiene datos correctos, no se puede mostrar el resumen.");
52         } else {
53             mostrarResumenResultados(montoTotalMultas, distanciaTotal, callesIncorrectas, calleMultas, calleOk);
54         }
55     } else {
56         printf("La cantidad de calles no es correcta");
57     }
58     return 0;
59 }
60 void leerDatos(double *distancia, double *tiempo, char *calle, int cont) {
61     printf("\nPara la calle %d\n", cont);
62     printf("Ingrese el nombre de la calle (A,B,C,D,E,F,G,H): ");
63     scanf(" %c", calle);
64     printf("Ingrese la distancia recorrida en kilómetros: ");
65     scanf("%lf", distancia);
66     printf("Ingrese el tiempo en segundos: ");
67     scanf("%lf", tiempo);
68 }
69 int validarDatos(double distancia, double tiempo, char calle) {
70     int datosValidos, distanciaValida, tiempoValido, calleValida;
71     distanciaValida=distancia>0;
72     tiempoValido=tiempo>0;
73     calleValida=(calle=='A') || (calle=='B') || (calle=='C') || (calle=='D') || (calle=='E') || (calle=='F') || (calle=='G') || (calle=='H');
74     datosValidos=distanciaValida && tiempoValido && calleValida;
75     return datosValidos;
76 }
77 double calcularMultas(char calle, double velocidad) {
78     /*Si no excedió el límite de velocidad, no corresponde multa*/

```

```

78     double montoMultas=0, diferenciaVelocidad;
79     if (calle=='A' || calle=='B' || calle=='C' ) {
80         diferenciaVelocidad=velocidad-LIMITE_A_B_C;
81     } else if (calle=='D' || calle=='E' || calle=='F') {
82         diferenciaVelocidad=velocidad-LIMITE_D_E_F;
83     } else if (calle=='G' || calle=='H') {
84         diferenciaVelocidad=velocidad-LIMITE_G_H;
85     }
86     if (diferenciaVelocidad>DIF_VEL) {
87         /*hay diferencia*/
88         if (diferenciaVelocidad-10>DIF_VEL) {
89             /*la diferencia es mayor que 10*/
90             montoMultas=MONTO2*UIT;
91         } else {
92             montoMultas=MONTO1*UIT;
93         }
94     }
95     return montoMultas;
96 }
97 void mostrarResumenResultados(double montoTotalMultas, double distanciaTotal,int callesIncorrectas,int calleMultas, int calleOk) {
98     printf("\nRESUMEN\n");
99     printf("Distancia total recorrida es: %.2lfKm\n", distanciaTotal);
100    printf("Cantidad de calles con algun dato incorrecto: %d\n",callesIncorrectas);
101    printf("Cantidad de calles donde excedio la velocidad permitida: %d\n",calleMultas);
102    printf("Cantidad de calles donde NO excedio la velocidad permitida: %d\n",calleOk);
103    printf("Monto total a pagar es: S/. %.2lf\n", montoTotalMultas);
104 }
```

4. Encuestas - Horarios: 0382, 0383

En cierto país se van a realizar elecciones, por lo que una empresa encuestadora ha convocado a varias personas para que realicen encuestas de manera presencial.

El país se encuentra dividido en 3 regiones (Costa, Sierra y Selva) y por cada región se ha establecido la siguiente tarifa por encuesta realizada:

- Pago por encuesta realizada en la región Costa: S/ 0.8
- Pago por encuesta realizada en la región Sierra: S/ 1.5
- Pago por encuesta realizada en la región Selva: S/ 1

Además, cada región ha sido dividida en 3 zonas (zona 1, zona 2 y zona 3) y cada zona tiene un límite máximo de encuestas que se puede realizar:

- En la región Costa:
 - Límite de encuestas en Zona 1: 45
 - Límite de encuestas en Zona 2: 55
 - Límite de encuestas en Zona 3: 60
- En la región Sierra:
 - Límite de encuestas en Zona 1: 75
 - Límite de encuestas en Zona 2: 80
 - Límite de encuestas en Zona 3: 90
- En la región Selva:

- Límite de encuestas en Zona 1: 40
- Límite de encuestas en Zona 1: 50
- Límite de encuestas en Zona 1: 65

Se le pide que implemente un programa en lenguaje C que solicite al usuario la región en donde encuestó, hasta que zona llegó a encuestar y la cantidad de encuestas realizadas por zona y que devuelva: La cantidad total de encuestas válidas que realizó, la cantidad total de encuestas extra que realizó, el monto que le pagaron por las encuestas válidas, el monto que le pagaron por las encuestas extra y la zona en donde le pagaron más por encuestas.

Para la implementación de su solución debe considerar lo siguiente:

- Debe validar que la región ingresada sea C para Costa, S para Sierra o E para Selva y que la cantidad de zonas encuestadas sea un número entre 1 y 3. En caso no se cumpla, se deberá emitir el siguiente mensaje “Uno de los datos ingresados no es correcto.” y el programa debe terminar.
- Debe validar que la cantidad de encuestas sea un número mayor que cero, caso contrario deberá emitir el siguiente mensaje “La cantidad de encuestas no puede ser un numero menor o igual que cero. No se considerara en el resumen.” y el programa debe continuar con su flujo.
- Considerar que si se supera el límite de encuestas realizada por zona, el pago por las encuestas restantes será de S/ 0.5
- Considerar que si se encuestó en la zona 2, antes se encuestó en la zona 1 y si se encuestó en la zona 3, antes se encuestó en la zona 1 y 2.

Debe implementar únicamente los siguientes módulos:

- Un módulo que solicite leer la cantidad de encuestas realizadas en una zona.
- Un módulo que calcule la cantidad de encuestas extra realizadas. Si no hubo encuestas extra, debe devolver cero.
- Un módulo que calcule el monto que se pagará por encuesta.
- Un módulo que imprima el resumen de los resultados (ver casos de prueba).

Nota:

Para la comparación de los montos (debido a la cantidad de decimales que podrían tener), considerar que un monto es mayor que otro si su diferencia es mayor que 0.001.

Casos de prueba

```
Ingrese la región (C)osta, (S)ierra, S(E)lva: Z
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 1
Uno de los datos ingresados no es correcto
```

```
Ingrese la región (C)osta, (S)ierra, S(E)lva: e
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 3
Uno de los datos ingresados no es correcto
```

Ingrese la región (C)osta, (S)ierra, S(E)lva: S
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 2
Ingrese la cantidad de encuestas que realizó en la zona 1: -1
La cantidad de encuestas no puede ser un número menor o igual que cero.
No se considerara en el resumen.
Ingrese la cantidad de encuestas que realizó en la zona 2: 50

RESUMEN

Cantidad total de encuestas validas que realizo: 50
Cantidad total de encuestas extra que realizo: 0
Monto que le pagaron por las encuestas validas: 75.000
Monto que le pagaron por las encuestas extra: 0.000
La zona 2 fue en donde le pagaron más por encuestas, el monto fue: 75.000

Ingrese la región (C)osta, (S)ierra, S(E)lva: C
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 3
Ingrese la cantidad de encuestas que realizó en la zona 1: 40
Ingrese la cantidad de encuestas que realizó en la zona 2: 50
Ingrese la cantidad de encuestas que realizó en la zona 3: 61

RESUMEN

Cantidad total de encuestas válidas que realizó: 150
Cantidad total de encuestas extra que realizó: 1
Monto que le pagaron por las encuestas válidas: 120.000
Monto que le pagaron por las encuestas extra: 0.500
La zona 3 fue en donde le pagaron más por encuestas, el monto fue: 48.500

Ingrese la región (C)osta, (S)ierra, S(E)lva: S
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 3
Ingrese la cantidad de encuestas que realizó en la zona 1: 20
Ingrese la cantidad de encuestas que realizó en la zona 2: 100
Ingrese la cantidad de encuestas que realizó en la zona 3: 30

RESUMEN

Cantidad total de encuestas validas que realizó: 130
Cantidad total de encuestas extra que realizó: 20
Monto que le pagaron por las encuestas válidas: 195.000
Monto que le pagaron por las encuestas extra: 10.000
La zona 2 fue en donde le pagaron más por encuestas, el monto fue: 130.000

Ingrese la región (C)osta, (S)ierra, S(E)lva: E
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 1
Ingrese la cantidad de encuestas que realizó en la zona 1: 115

RESUMEN

Cantidad total de encuestas válidas que realizó: 40
Cantidad total de encuestas extra que realizó: 75
Monto que le pagaron por las encuestas válidas: 40.000
Monto que le pagaron por las encuestas extra: 37.500
La zona 1 fue en donde le pagaron más por encuestas, el monto fue: 77.500

```

Ingrese la región (C)osta, (S)ierra, S(E)lva: S
Ingrese hasta qué zona pudo realizar encuestas (1,2 ó 3): 3
Ingrese la cantidad de encuestas que realizó en la zona 1: 43
Ingrese la cantidad de encuestas que realizó en la zona 2: -33
La cantidad de encuestas no puede ser un numero menor o igual que cero.
No se considerará en el resumen
Ingrese la cantidad de encuestas que realizó en la zona 3: 90

```

RESUMEN

```

Cantidad total de encuestas validas que realizó: 133
Cantidad total de encuestas extra que realizó: 0
Monto que le pagaron por las encuestas válidas: 199.500
Monto que le pagaron por las encuestas extra: 0.000
La zona 3 fue en donde le pagaron más por encuestas, el monto fue: 135.000

```

Programa 4: Propuesta de solución - Encuestas

```

1 #include <stdio.h>
2 #define DIF_MONTOS 0.001
3 #define LIM_C_1 45
4 #define LIM_C_2 55
5 #define LIM_C_3 60
6 #define LIM_S_1 75
7 #define LIM_S_2 80
8 #define LIM_S_3 90
9 #define LIM_E_1 40
10 #define LIM_E_2 50
11 #define LIM_E_3 65
12 #define MONTO_C 0.8
13 #define MONTO_S 1.5
14 #define MONTO_E 1
15 #define MONTO_EXTRA 0.5
16
17 int calcularExcesoEncuestas(char, int, int);
18 void leerDatos(int, int *);
19 double calcularMontoXEncuesta(char);
20 void mostrarResumenResultados(double ,int ,int ,double ,int , double );
21
22 int main() {
23     int cantZonas,cantEncuestasExtra,cantEncuestasValidas,regionValida,zonaMayor=-1;
24     int cantZonasValidas,cont=1,zona,cantEncuestas,cantTotalEncuestasValidas,cantTotalEncuestasExtra;
25     double montoTotalZona,montoZonaValida,montoMayor=-1,montoEncuestaValida,sumaMontoValido=0,sumaMontoExtra=0,
26         montoZonaExtra;
27     char region;
28
29     printf("Ingrese la región (C)osta, (S)ierra, S(E)lva: ");
30     scanf(" %c",&region);
31     printf("Ingrese hasta que zona pudo realizar encuestas (1,2 o 3): ");
32     scanf(" %d",&cantZonas);
33     cantZonasValidas=cantZonas>0 && cantZonas<=3;
34     regionValida=region=='C' || region=='S' || region=='E';
35     if(cantZonasValidas && regionValida) {
36         while (cont<=cantZonas) {
37             /*pedimos la cantidad de encuestas realizadas por zona*/
38
39             leerDatos(cont,&cantEncuestas);
40             if (cantEncuestas>0) {
41                 /*Una zona x región tiene un máximo de encuestas*/
42                 cantEncuestasExtra=calcularExcesoEncuestas(region, cont, cantEncuestas);
43                 cantEncuestasValidas=cantEncuestas-cantEncuestasExtra;
44                 /*El monto a pagar por encuesta depende de la región*/
45                 montoEncuestaValida=calcularMontoXEncuesta(region);

```

```

45     montoZonaValida=montoEncuestaValida*cantEncuestasValidas;
46     sumaMontoValido=sumaMontoValido+montoZonaValida;
47     /*uso contadores para el total de encuestas válidas y extra*/
48     cantTotalEncuestasValidas=cantTotalEncuestasValidas+cantEncuestasValidas;
49     cantTotalEncuestasExtra=cantTotalEncuestasExtra+cantEncuestasExtra;
50     montoZonaExtra=0;
51     if (cantEncuestasExtra>0) {
52         /*Hay encuestas extra*/
53         montoZonaExtra=cantEncuestasExtra*MONTO_EXTRA;
54         sumaMontoExtra=sumaMontoExtra+montoZonaExtra;
55     }
56     montoTotalZona=montoZonaValida+montoZonaExtra;
57     if (montoTotalZona-montoMayor>DIF_MONTOS) {
58         montoMayor=montoTotalZona;
59         zonaMayor=cont;
60     }
61 } else {
62     printf("La cantidad de encuestas no puede ser un número menor o igual que cero. No se considerara
63         en el resumen\n");
64 }
65 cont++;
66 mostrarResumenResultados(sumaMontoValido,cantTotalEncuestasValidas,cantTotalEncuestasExtra,sumaMontoExtra,
67     zonaMayor,montoMayor);
68 } else {
69     printf("Uno de los datos ingresados no es correcto");
70 }
71 return 0;
72 }
73 void leerDatos(int cont, int *cantEncuestas) {
74     printf("Ingrese la cantidad de encuestas que realizo en la zona %d: ",cont);
75     scanf("%d",cantEncuestas);
76 }
77 int calcularExcesoEncuestas(char region, int zona, int cantEncuestas) {
78     /*Si no hay exceso de encuestas, retorna 0*/
79     int cantEncuestasExtra=0,limiteEncuesta;
80     if (region=='C') {
81         if (zona==1) {
82             limiteEncuesta=LIM_C_1;
83         } else if (zona==2) {
84             limiteEncuesta=LIM_C_2;
85         } else {
86             limiteEncuesta=LIM_C_3;
87         }
88     } else if (region=='S') {
89         if (zona==1) {
90             limiteEncuesta=LIM_S_1;
91         } else if (zona==2) {
92             limiteEncuesta=LIM_S_2;
93         } else {
94             limiteEncuesta=LIM_S_3;
95         }
96     } else {
97         if (zona==1) {
98             limiteEncuesta=LIM_E_1;
99         } else if (zona==2) {
100            limiteEncuesta=LIM_E_2;
101        } else {
102            limiteEncuesta=LIM_E_3;
103        }
104    if (cantEncuestas-limiteEncuesta>0) {
105        cantEncuestasExtra=cantEncuestas-limiteEncuesta;
106    }
107 }
108 double calcularMontoXEncuesta(char region) {
109     double monto;

```

```

110     if (region=='C') {
111         monto=MONTO_C;
112     } else if (region=='S') {
113         monto=MONTO_S;
114     } else {
115         monto=MONTO_E;
116     }
117     return monto;
118 }
119 void mostrarResumenResultados(double sumaMontoValido,int cantTotalEncuestasValidas,int cantTotalEncuestasExtra,double
120     sumaMontoExtra,int zonaMayor, double montoMayor) {
121     printf("\nRESUMEN\n");
122     printf("Cantidad total de encuestas válidas que realizó: %d\n", cantTotalEncuestasValidas);
123     printf("Cantidad total de encuestas extra que realizó: %d\n", cantTotalEncuestasExtra);
124     printf("Monto que le pagaron por las encuestas válidas: %.3lf\n", sumaMontoValido);
125     printf("Monto que le pagaron por las encuestas extra: %.3lf\n", sumaMontoExtra);
126     printf("La zona %d fue en donde le pagaron mas por encuestas, el monto fue: %.3lf\n", zonaMayor, montoMayor);
127 }
```

No debe usar estructuras algorítmicas iterativas anidadas, iterativas con salida controlada o por centinela.