

FUNDAMENTOS DE PROGRAMACIÓN
LABORATORIO 9
PROPUESTAS DE SOLUCIÓN
SEMESTRE ACADÉMICO 2021-2

Horarios: Todos los horarios

Elaborado por Jennifer Zárate y Jorge Berrocal

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 diseño estructurado.

CONSIDERACIONES:

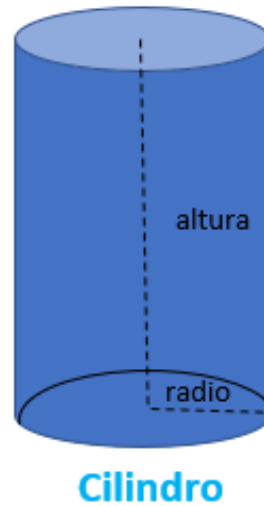
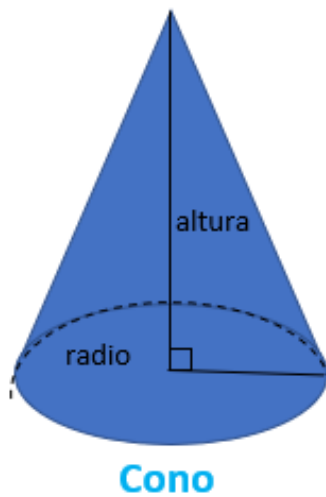
- La solución presentada para cada problema corresponde a una propuesta de solución por parte del autor.
- En programación pueden existir muchas soluciones para un mismo problema pero debe cumplir con todo lo solicitado, incluyendo las restricciones brindadas.

Desarrolle los siguientes problemas en lenguaje C:

1. Conos y cilindros

Existen diversas formas de analizar las figuras de acuerdo a los espacios que se tengan, por ejemplo, en dos dimensiones o tres dimensiones. Al realizar análisis tridimensionales, se toma en cuenta el ancho, la profundidad y la altura.

El cono es una figura que se encuentra en tres dimensiones y resulta de girar un triángulo rectángulo sobre uno de sus catetos, por lo que es un cuerpo geométrico que está constituido por un vértice y una base circular. Mientras que el cilindro también es una figura geométrica tridimensional que se genera al girar una recta alrededor de un eje y en torno a una superficie curva plana en la base, está formado por dos bases circulares. A continuación, las representaciones gráficas de ambas figuras:



Se le pide elaborar un programa en lenguaje C que, utilizando el paradigma de programación modular, permita realizar la validación y cálculo de valores de distintas figuras ingresados. Para ello, se solicitará ingresar la figura a evaluar, se verificará que sea la letra 'C' y terminará cuando se ingrese la letra 'Z'. Luego, se deberá evaluar el tipo de figura de acuerdo a la cantidad de bases que se ingrese, una base es para un cono y dos bases es para un cilindro. Para cualquier otra opción deberá imprimir el mensaje "Figura no identificada".

Dentro de cada tipo de figura, se solicitará ingresar la cantidad de figuras de ese tipo, calcular e imprimir el volumen de cada figura, y luego obtener el mayor volumen de conos y el menor volumen de cilindros calculados de dicho grupo.

Para hallar el volumen del cono o del cilindro, debe solicitar el radio y la altura y validar que sean positivos, además debe asumir el valor de PI como 3.141592. A continuación, las fórmulas a utilizar:

Fórmulas:

$$Volumen_{cono} = \pi * radio^2 * altura / 3$$

$$Volumen_{cilindro} = \pi * radio^2 * altura$$

También se le pide calcular e imprimir la cantidad total de figuras analizadas, el porcentaje de conos, el porcentaje de cilindros, así como cuál es el mayor volumen de los conos y el menor volumen de los cilindros.

Para su solución deberá implementar módulos con las siguientes restricciones y considere que no deberá añadir algún módulo adicional:

- Un módulo que calcule y devuelva dos o más valores.
- Dos módulos que calculen y devuelvan un valor

Se muestran los siguientes casos de prueba:

Caso de prueba 1

Ingrese el tipo de figura a analizar: C
Ingrese la cantidad de bases del grupo de figuras a evaluar: 2
Figura identificada: CILINDRO
Ingrese la cantidad de cilindros a evaluar: 3
Cilindro 1 - Ingrese el radio de la base: 5
Cilindro 1 - Ingrese la altura del cilindro: 10
El volumen del cilindro 1 es: 785.40
Cilindro 2 - Ingrese el radio de la base: 6
Cilindro 2 - Ingrese la altura del cilindro: 6.5
El volumen del cilindro 2 es: 735.13
Cilindro 3 - Ingrese el radio de la base: 10
Cilindro 3 - Ingrese la altura del cilindro: 3
El volumen del cilindro 3 es: 942.48
Ingrese el tipo de figura a analizar: C
Ingrese la cantidad de bases del grupo de figuras a evaluar: 1
Figura identificada: CONO
Ingrese la cantidad de conos a evaluar: 2
Cono 1 - Ingrese el radio de la base: 4
Cono 1 - Ingrese la altura del cono: 20
El volumen del cono 1 es: 335.10
Cono 2 - Ingrese el radio de la base: 8
Cono 2 - Ingrese la altura del cono: 6
El volumen del cono 2 es: 402.12
Ingrese el tipo de figura a analizar: C
Ingrese la cantidad de bases del grupo de figuras a evaluar: 1
Figura identificada: CONO
Ingrese la cantidad de conos a evaluar: 3
Cono 1 - Ingrese el radio de la base: 7.4
Cono 1 - Ingrese la altura del cono: 9.2
El volumen del cono 1 es: 527.57
Cono 2 - Ingrese el radio de la base: 5.7
Cono 2 - Ingrese la altura del cono: 11.2
El volumen del cono 2 es: 381.06
Cono 3 - Ingrese el radio de la base: 8
Cono 3 - Ingrese la altura del cono: 5.3
El volumen del cono 3 es: 355.21
Ingrese el tipo de figura a analizar: d
Ingrese el tipo de figura a analizar: Z
REPORTE

El total de figuras es: 8
El porcentaje de conos analizados es: 62.00
El porcentaje de cilindros analizados es: 37.00
El mayor volumen de los conos es: 527.57
El menor volumen de los cilindros es: 735.13

Caso de prueba 2

Ingrese el tipo de figura a analizar: A
Ingrese el tipo de figura a analizar: X
Ingrese el tipo de figura a analizar: C
Ingrese la cantidad de bases del grupo de figuras a evaluar: 1
Figura identificada: _____CONO_____

Ingrese la cantidad de conos a evaluar: 3

Cono 1 - Ingrese el radio de la base: 10

Cono 1 - Ingrese la altura del cono: 12

El volumen del cono 1 es: 1256.64

Cono 2 - Ingrese el radio de la base: 11.4

Cono 2 - Ingrese la altura del cono: 5.2

El volumen del cono 2 es: 707.69

Cono 3 - Ingrese el radio de la base: 19.7

Cono 3 - Ingrese la altura del cono: 6.5

El volumen del cono 3 es: 2641.64

Ingrese el tipo de figura a analizar: C

Ingrese la cantidad de bases del grupo de figuras a evaluar: 2

Figura identificada: _____CILINDRO_____

Ingrese la cantidad de cilindros a evaluar: 2

Cilindro 1 - Ingrese el radio de la base: 34

Cilindro 1 - Ingrese la altura del cilindro: 11

El volumen del cilindro 1 es: 39948.48

Cilindro 2 - Ingrese el radio de la base: 32.7

Cilindro 2 - Ingrese la altura del cilindro: 16.9

El volumen del cilindro 2 es: 56771.71

Ingrese el tipo de figura a analizar: Z

REPORTE

El total de figuras es: 5

El porcentaje de conos analizados es: 60.00

El porcentaje de cilindros analizados es: 40.00

El mayor volumen de los conos es: 2641.64

El menor volumen de los cilindros es: 9999.00

Programa 1: Propuesta de solución - Conos y Cilindros

```
1 #include<stdio.h>
2 #include<math.h>
3
4 #define PI 3.141592
5
6 void ObtenerValoresFiguras(int *tipoFigura, int *cantConos,int *cantCilindros,double *mayorVolumenCono,double *
    menorVolumenCilindro);
7 double CalcularVolumenCono(double radio,double altura);
8 double CalcularVolumenCilindro(double radio,double altura);
9
10 int main(){
11     int fin,tipoFigura, cantConos,cantCilindros,totalConos,totalCilindros;
12     double mayorVolumen, menorVolumen,mayorVolumenCono,menorVolumenCilindro,porcConos,porcCilindros;
13     char figura;
14
15     mayorVolumen=0;
16     menorVolumen=1000000;
17     fin=0;
18     totalConos=0;
19     totalCilindros=0;
20     while(!fin){
21         printf("Ingrese el tipo de figura a analizar: ");
```

```

22     scanf("\n %c",&figura);
23     if (figura=='C'){
24         ObtenerValoresFiguras(&tipoFigura, &cantConos,&cantCilindros,&mayorVolumenCono,&
25             menorVolumenCilindro);
26         if (tipoFigura==1){
27             totalConos+=cantConos;
28         }
29         else{
30             totalCilindros+=cantCilindros;
31         }
32         if (mayorVolumenCono>mayorVolumen){
33             mayorVolumen=mayorVolumenCono;
34         }
35         if (menorVolumenCilindro<menorVolumen){
36             menorVolumen=menorVolumenCilindro;
37         }
38         printf("\n");
39     }
40     else if (figura=='Z'){
41         fin=1;
42     }
43 }
44 if (fin){
45     porcConos=100*totalConos/(totalConos+totalCilindros);
46     porcCilindros=100*totalCilindros/(totalConos+totalCilindros);
47     printf("\nREPORTE\n");
48     printf("-----\n");
49     printf("El total de figuras es: %d\n",totalConos+totalCilindros);
50     printf("El porcentaje de conos analizados es: %.2lf %\n",porcConos);
51     printf("El porcentaje de cilindros analizados es: %.2lf %\n",porcCilindros);
52     printf("El mayor volumen de los conos es: %.2lf\n",mayorVolumen);
53     printf("El menor volumen de los cilindros es: %.2lf\n",menorVolumen);
54 }
55 return 0;
56 }
57
58 void ObtenerValoresFiguras(int *tipoFigura, int *cantConos,int *cantCilindros,double *mayorVolumenCono,double *
59     menorVolumenCilindro){
60     int nBases,nConos,nCilindros,i,j;
61     double radio,altura,volumen,mayor,menor;
62
63     printf("Ingrese la cantidad de bases del grupo de figuras a evaluar: ");
64     scanf("%d",&nBases);
65     /*1 si es cono y 2 si es cilindro*/
66     if (nBases==1){
67         *tipoFigura=1;
68         printf("Figura identificada: -----CONO-----\n");
69         printf("Ingrese la cantidad de conos a evaluar: ");
70         scanf("%d",&nConos);
71         /*por cada pirámide, se debe solicitar datos para hacer los cálculos*/
72         mayor=0;
73         for (i=1;i<=nConos;i++){
74             printf("Cono %d – Ingrese el radio de la base: ",i);
75             scanf("%lf",&radio);
76             if (radio>0){
77                 printf("Cono %d – Ingrese la altura del cono: ",i);
78                 scanf("%lf",&altura);
79                 volumen=CalcularVolumenCono(radio,altura);
80                 printf("El volumen del cono %d es: %.2lf\n",i,volumen);
81             }
82             else{
83                 printf("Ingresó una valor incorrecto para el radio.\n");
84             }
85             if (volumen>mayor){
86                 mayor=volumen;
87             }
88         }
89     }
90 }

```

```

87     }
88     *cantConos=nConos;
89     *cantCilindros=0;
90     *mayorVolumenCono=mayor;
91     *menorVolumenCilindro=9999;
92
93 }
94 else if (nBases==2){
95     *tipoFigura=2;
96     printf("Figura identificada: -----CILINDRO-----\n");
97     printf("Ingrese la cantidad de cilindros a evaluar: ");
98     scanf("%d",&nCilindros);
99     /*por cada pirámide, se debe solicitar datos para hacer los cálculos*/
100    menor=1000000;
101    for (j=1;j<=nCilindros;j++){
102        printf("Cilindro %d – Ingrese el radio de la base: ",j);
103        scanf("%lf",&radio);
104        if (radio>0){
105            printf("Cilindro %d – Ingrese la altura del cilindro: ",j);
106            scanf("%lf",&altura);
107            volumen=CalcularVolumenCilindro(radio,altura);
108            printf("El volumen del cilindro %d es: %.2lf\n",j,volumen);
109        }
110        else{
111            printf("Ingresó una valor incorrecto para el radio.\n");
112        }
113        if (volumen<menor){
114            menor=volumen;
115        }
116    }
117    *cantConos=0;
118    *cantCilindros=nCilindros;
119    *mayorVolumenCono=0;
120    *menorVolumenCilindro=menor;
121 }
122 else{
123     printf("Figura no identificado");
124 }
125 }
126
127 double CalcularVolumenCono(double radio,double altura){
128     return (PI*pow(radio,2)*altura/3);
129 }
130
131 double CalcularVolumenCilindro(double radio,double altura){
132     return (PI*pow(radio,2)*altura);
133 }

```

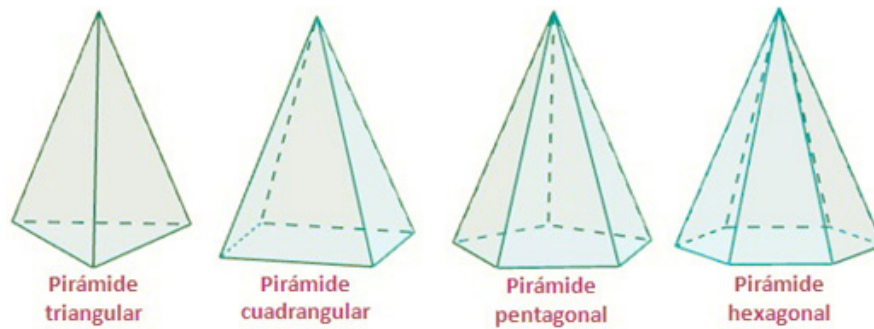
2. Figuras tridimensionales

La geometría del espacio es una rama de la geometría que se encarga de evaluar y estudiar las figuras que ocupan un lugar en el espacio, es decir, tienen un volumen. Estas figuras pueden ser esferas, poliedros, cilindros, pirámides, entre otros.

Dentro de las figuras geométricas se tiene a las pirámides que tienen solo una base de forma poligonal y sus caras son de forma triangular por lo que tienen siempre un vértice en la parte superior que une las caras. Por otro lado, se tienen los poliedros regulares cuyas caras son de la misma forma y del mismo tamaño.

Se le pide elaborar un programa en lenguaje C que, utilizando el paradigma de programación modular, permita realizar la evaluación de las figuras tridimensionales. Deberá ingresar la cantidad de bases y de acuerdo a ello realizar la evaluación de un grupo de figuras de un tipo y otro. Si ingresa 1 se evaluarán pirámides, si ingresa más de 1 se evaluarán poliedros y si ingresa 0, terminará.

Sobre las pirámides, no se conoce la cantidad a evaluar. Por lo que, deberá considerar aquellas que tengan como base las indicadas en la siguiente figura. Si ingresa un valor diferente terminará de evaluar este tipo de figura:



Para cada pirámide, solicitará la cantidad de lados de la base y deberá mostrar el tipo. También solicitará la longitud de cada lado, la apotema de la base y la apotema de la pirámide para calcular y mostrar el área de la base, el área lateral y el área total. Podrá usar las siguientes fórmulas:

Fórmulas:

$$Area_{base} = N * L * apb/2$$

$$Area_{lateral} = N * L * ap/2$$

$$Area_{total} = Area_{base} + Area_{lateral}$$

Donde:

- N : número de lados de la base
- L : longitud de cada lado
- apb: apotema de la base
- ap: apotema de la pirámide

También deberá calcular y mostrar el mayor valor del área total y la mayor diferencia entre área de la base y área lateral del grupo de figuras del tipo determinado.

Con respecto a los poliedros, únicamente se trabajará con los que aparecen en la siguiente imagen:



Se solicitará al usuario la cantidad de poliedros a evaluar. Por cada uno de ellos, pedirá el número de caras para calcular y mostrar el tipo de prisma, el número de aristas, el número de vértices y el ángulo de cada vértice. Considerar los siguientes valores por cada tipo de poliedro:

Poliedro	Nº caras	Nº aristas
Tetraedro regular	4	6
Hexaedro regular	6	12
Octaedro regular	8	12
Dodecaedro regular	12	30
Icosaedro regular	20	30

Para obtener los valores solicitados, deberá usar la siguiente fórmula:

$$C + V = A + 2$$

Donde:

- C: número de caras
- V: número de vértices
- A: número de aristas

Para su solución deberá implementar módulos con las siguientes restricciones y considere que no deberá añadir algún módulo adicional:

- Dos módulos que calculen y devuelvan dos o más valores.
- Dos módulos que calculen y devuelvan un valor

Se muestran los siguientes casos de prueba:

Caso de prueba 1

Ingrese la cantidad de bases a evaluar: 2

Figura evaluada: Prisma

Ingrese la cantidad de prismas a evaluar: 4

Poliedro 1 - Ingrese el número de caras: 4

Poliedro 1 - Tetraedro regular con 4 caras, 6 aristas y 4 vértices de 90 ángulos

Poliedro 2 - Ingrese el número de caras: 12

Poliedro 2 - Dodecaedro regular - con 12 caras, 30 aristas y 20 vértices de 18 ángulos

Poliedro 3 - Ingrese el número de caras:

Poliedro 3 - Icosaedro regular - con 20 caras, 30 aristas y 12 vértices de 30 ángulos

Poliedro 4 - Ingrese el número de caras: 8

Poliedro 4 - Octaedro regular - con 8 caras, 12 aristas y 6 vértices de 60 ángulos

Ingrese la cantidad de bases a evaluar: 1

Figura evaluada: Pirámide

Pirámide 1 - Ingrese la cantidad de lados de la base de la pirámide: 3

Pirámide 1 - Pirámide Triangular

Pirámide 1 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 4 5 6.7

Pirámide 1 - Tiene de área base = 30.00, área lateral = 40.20 y área total = 70.20

Pirámide 1 - La diferencia entre el área de la base y el área lateral es: 10.20

Pirámide 2 - Ingrese la cantidad de lados de la base de la pirámide: 5

Pirámide 2 - Pirámide Pentagonal

Pirámide 2 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 5.4 2.1 7.3

Pirámide 2 - Tiene de área base = 28.35, área lateral = 98.55 y área total = 126.90

Pirámide 2 - La diferencia entre el área de la base y el área lateral es: 70.20

Pirámide 3 - Ingrese la cantidad de lados de la base de la pirámide: 6

Pirámide 3 - Pirámide Hexagonal

Pirámide 3 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 10 4.5 6.9

Pirámide 3 - Tiene de área base = 135.00, área lateral = 207.00 y área total = 342.00

Pirámide 3 - La diferencia entre el área de la base y el área lateral es: 72.00

Pirámide 4 - Ingrese la cantidad de lados de la base de la pirámide: 0

Ingresó un valor incorrecto

Pirámides - El mayor valor del área total es 342.00

Pirámides - El mayor valor de la diferencia de bases es 72.00

Ingrese la cantidad de bases a evaluar: 2

Figura evaluada: Prisma

Ingrese la cantidad de prismas a evaluar: 2

Poliedro 1 - Ingrese el número de caras: 12

Poliedro 1 - Dodecaedro regular - con 12 caras, 30 aristas y 20 vértices de 18 ángulos

Poliedro 2 - Ingrese el número de caras: 4

Poliedro 2 - Tetraedro regular con 4 caras, 6 aristas y 4 vértices de 90 ángulos

Ingrese la cantidad de bases a evaluar: 0

La cantidad de figuras analizadas es 9

Caso de prueba 2

Ingrese la cantidad de bases a evaluar: 1

Figura evaluada: Pirámide

Pirámide 1 - Ingrese la cantidad de lados de la base de la pirámide: 4

Pirámide 1 - Pirámide Cuadrangular

Pirámide 1 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 7 4 2.1

Pirámide 1 - Tiene de área base = 56.00, área lateral = 29.40 y área total = 85.40

Pirámide 1 - La diferencia entre el área de la base y el área lateral es: 26.60

Pirámide 2 - Ingrese la cantidad de lados de la base de la pirámide: 5

Pirámide 2 - Pirámide Pentagonal

Pirámide 2 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 9 4.5 7

Pirámide 2 - Tiene de área base = 101.25, área lateral = 157.50 y área total = 258.75

Pirámide 2 - La diferencia entre el área de la base y el área lateral es: 56.25

Pirámide 3 - Ingrese la cantidad de lados de la base de la pirámide: 6

Pirámide 3 - Pirámide Hexagonal

Pirámide 3 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 6 7 5.5

Pirámide 3 - Tiene de área base = 126.00, área lateral = 99.00 y área total = 225.00

Pirámide 3 - La diferencia entre el área de la base y el área lateral es: 27.00

Pirámide 4 - Ingrese la cantidad de lados de la base de la pirámide: 5

Pirámide 4 - Pirámide Pentagonal

Pirámide 4 - Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: 6 5.4 3.4

Pirámide 4 - Tiene de área base = 81.00, área lateral = 51.00 y área total = 132.00

Pirámide 4 - La diferencia entre el área de la base y el área lateral es: 30.00

Pirámide 5 - Ingrese la cantidad de lados de la base de la pirámide: 0

Ingresó un valor incorrecto

Pirámides - El mayor valor del área total es 258.75

Pirámides - El mayor valor de la diferencia de bases es 56.25

Ingrese la cantidad de bases a evaluar: 3

Figura evaluada: Poliedro regular

Ingrese la cantidad de poliedros regulares a evaluar: 2

Poliedro 1 - Ingrese el número de caras: 8

Poliedro 1 - Octaedro regular - con 8 caras, 12 aristas y 6 vértices de 60 ángulos

Poliedro 2 - Ingrese el número de caras: 20

Poliedro 2 - Icosaedro regular - con 20 caras, 30 aristas y 12 vértices de 30 ángulos

Ingrese la cantidad de bases a evaluar: 0

La cantidad de figuras analizadas es 6

Programa 2: Propuesta de solución - Figuras Tridimensionales

```
1  #include<stdio.h>
2  #include<math.h>
3
4  int ValidarLadosBase(int nLadosBase);
5  void ObtenerValoresPiramide(int j,int nLadosBase,double *areaBase,double *areaLateral);
6  void ObtenerAristasVerticesPoliedro(int i,int *nCaras,int *nAristas,int *nVertices);
7
8  int main(){
9      int nBases, cantFiguras,cantPiramides,cantPrismas,fin,i,nCaras,nAristas,nVertices,angulo,hayPiramide,j,nLadosBase;
10     double areaBase,areaLateral, areaPiramide,diferenciaBases,mayorAreaPiramide,mayorDiferencia;
11     mayorAreaPiramide=0;
12     mayorDiferencia=0;
13     cantFiguras=0;
14     cantPiramides=0;
15     cantPrismas=0;
16     fin=0;
17     do{
18         printf("Ingrese la cantidad de bases a evaluar: ");
19         scanf("%d",&nBases);
```

```

20     if (nBases==1){
21         printf("Figura evaluada: Pirámide\n");
22         hayPiramide=1;
23         j=1;
24         while (hayPiramide){
25             printf("Pirámide # %d – Ingrese la cantidad de lados de la base de la pirámide: ",j);
26             scanf("%d",&nLadosBase);
27             if (ValidarLadosBase(nLadosBase)){
28                 ObtenerValoresPiramide(j,nLadosBase,&areaBase,&areaLateral);
29                 areaPiramide=areaBase+areaLateral;
30                 diferenciaBases=fabs(areaBase-areaLateral);
31                 printf("Pirámide # %d – Tiene de área base = %.2lf, área lateral = %.2lf y área total = %.2lf\n",j,areaBase,areaLateral,areaPiramide);
32                 printf("Pirámide # %d – La diferencia entre el área de la base y el área lateral es: %.2lf\n",j,diferenciaBases);
33                 j++;
34                 if (areaPiramide>mayorAreaPiramide){
35                     mayorAreaPiramide=areaPiramide;
36                 }
37                 if (diferenciaBases>mayorDiferencia){
38                     mayorDiferencia=diferenciaBases;
39                 }
40             }else{
41                 hayPiramide=0;
42                 printf("Ingresó un valor incorrecto\n");
43             }
44         }
45         printf("Pirámides – El mayor valor del área total es %.2lf\n",mayorAreaPiramide);
46         printf("Pirámides – El mayor valor de la diferencia de bases es %.2lf\n",mayorDiferencia);
47         printf("\n");
48         cantFiguras+=j-1;
49     }
50     else if (nBases>1){
51         printf("Figura evaluada: Poliedro regular\n");
52         printf("Ingrese la cantidad de poliedros regulares a evaluar: ");
53         scanf("%d",&cantPrismas);
54
55         for (i=1;i<=cantPrismas;i++){
56             ObtenerAristasVerticesPoliedro(i,&nCaras,&nAristas,&nVertices);
57             angulo=360/nVertices;
58             printf("con %d caras, %d aristas y %d vértices de %d ángulos\n",nCaras,nAristas,nVertices,angulo);
59             ;
60         }
61         printf("\n");
62         cantFiguras+=cantPrismas;
63     }
64     else if (nBases==0){
65         fin=1;
66     }
67     else {
68         printf("La cantidad de bases es incorrecta. Ingrese una base correcta.\n");
69     }
70     while (!fin);
71
72     printf("La cantidad de figuras analizadas es %d",cantFiguras);
73
74     return 0;
75 }
76
77 int ValidarLadosBase(int nLadosBase){
78     return ((nLadosBase>=3) && (nLadosBase<=6));
79 }
80
81 void ObtenerValoresPiramide(int j,int nLadosBase,double *areaBase,double *areaLateral){
82     double lado,apb,ap;
83     if (nLadosBase==3){
84         printf("Pirámide # %d – Pirámide Triangular \n",j);

```

```

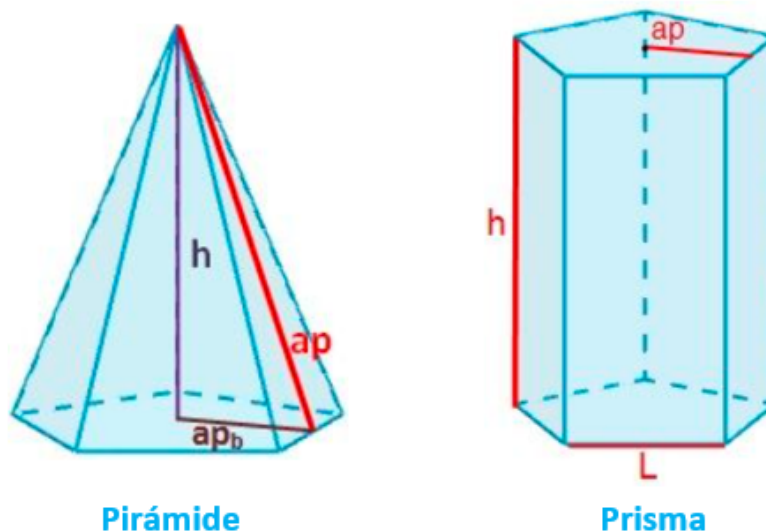
84     }
85     else if (nLadosBase==4){
86         printf("Pirámide # %d – Pirámide Cuadrangular \n",j);
87     }
88     else if (nLadosBase==5){
89         printf("Pirámide # %d – Pirámide Pentagonal \n",j);
90     }
91     else if (nLadosBase==6){
92         printf("Pirámide # %d – Pirámide Hexagonal \n",j);
93     }
94
95     printf("Pirámide # %d – Ingrese la longitud de los lados, la apotema de la base y la apotema de la pirámide: ",j);
96     scanf("%lf %lf %lf",&lado,&apb,&ap);
97     *areaBase=nLadosBase*lado*apb/2;
98     *areaLateral=nLadosBase*lado*ap/2;
99 }
100
101 void ObtenerAristasVerticesPoliedro(int i,int *nCaras,int *nAristas,int *nVertices){
102
103     printf("Poliedro # %d – Ingrese el número de caras: ",i);
104     scanf("%d",&nCaras);
105
106     if (*nCaras==4){
107         printf("Poliedro # %d – Tetraedro regular ",i);
108         *nAristas=6;
109     }
110     else if(*nCaras==6){
111         printf("Poliedro # %d – Hexaedro regular – ",i);
112         *nAristas=12;
113     }
114     else if(*nCaras==8){
115         printf("Poliedro # %d – Octaedro regular – ",i);
116         *nAristas=12;
117     }
118     else if(*nCaras==12){
119         printf("Poliedro # %d – Dodecaedro regular – ",i);
120         *nAristas=30;
121     }
122     else if(*nCaras==20){
123         printf("Poliedro # %d – Icosaedro regular – ",i);
124         *nAristas=30;
125     }
126     *nVertices=*nAristas-*nCaras+2;
127 }

```

3. Pirámides y prismas

La geometría del espacio es la rama de la geometría que se encarga del estudio de figuras geométricas de tres dimensiones: ancho, profundidad y altura. Dentro de estas figuras, se tiene como ejemplo a los conos, prismas, cubos, poliedros regulares, entre otros.

En el caso de los poliedros pueden clasificarse en regulares e irregulares. Los poliedros regulares son aquellos que tienen todas sus caras, aristas y ángulos iguales; sobre los poliedros irregulares, tiene todas partes diferentes. Otra clasificación de los poliedros corresponde a las pirámides y prismas las cuales se diferencian, entre otras cosas, porque las pirámides tienen una sola base de forma poligonal y lados triangulares, mientras que los prismas tienen dos bases poligonales y lados rectangulares como se muestra en la siguiente figura:



Las bases de estas figuras pueden ser de distintas formas. Por ejemplo: triangulares, cuadradas, pentagonales, entre otros. Para efectos de este ejercicio, considerar la base hasta con 8 lados.

Se le pide elaborar un programa en Lenguaje C utilizando el paradigma de programación modular, que permita realizar la validación y cálculo de valores de distintos poliedros ingresados. Para ello, se solicitará ingresar la figura a evaluar, se verificará que sea la letra 'P' y terminará cuando se ingrese la letra 'F'. Para cada figura validada, se deberá evaluar el tipo de poliedro de acuerdo a la cantidad de bases que se ingrese, una base es para una pirámide y dos bases es para un prisma. Para cualquier otra opción deberá imprimir el mensaje "Poliedro no identificado". Dentro de cada opción, deberá ingresar la cantidad de figuras de ese tipo, calcular e imprimir el volumen de cada figura, y luego calcular el promedio de todos los volúmenes calculados de dicho grupo.

Para hallar el volumen de la pirámide o prisma, debe considerar que se trata de poliedros regulares y también se validará que la base o bases tengan de 3 a 8 lados. A continuación, las fórmulas a utilizar:

Fórmulas:

$$Volumen_{pirámide} = Área_{base} * altura / 3$$

$$Volumen_{pirámide} = Área_{base} * altura$$

$$Área_{base} = nLados * valorLado * apotema / 2$$

También se le pide calcular e imprimir la cantidad total de poliedros analizados, el porcentaje de pirámides, el porcentaje de prismas, así como cuál es el mayor y menor de los promedios calculados.

Para su solución deberá implementar módulos con las siguientes restricciones y considere que no deberá añadir algún módulo adicional:

- Un módulo que calcule y devuelva dos o más valores.
- Dos módulos que calculen y devuelvan un valor

Se muestran los siguientes casos de prueba:

Caso de prueba 1

Ingrese el tipo de poliedro a analizar: P

Ingrese la cantidad de bases que tendrán las figuras a evaluar: 2

Figura identificada: _____PRISMA _____

Ingrese la cantidad de prismas a evaluar: 3

Prisma 1 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 4 4 2

Prisma 1 - Ingrese la altura de la pirámide: 5

El volumen del prisma 1 es: 80.00

Prisma 2 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 5 6 3

Prisma 2 - Ingrese la altura de la pirámide: 4

El volumen del prisma 2 es: 180.00

Prisma 3 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 6 10 5

Prisma 3 - Ingrese la altura de la pirámide: 6

El volumen del prisma 3 es: 900.00

Promedio de volúmenes de los 3 prismas es 386.67

Ingrese el tipo de poliedro a analizar: F

REPORTE _____

El total de poliedros es: 3

El porcentaje de pirámides analizadas es: 0.00

El porcentaje de prismas analizados es: 100.00

El menor valor de promedio de volumen de los poliedros es: 386.67

El mayor valor de promedio de volumen de los poliedros es: 386.67

Caso de prueba 2

Ingrese el tipo de poliedro a analizar: P

Ingrese la cantidad de bases que tendrán las figuras a evaluar: 1

Figura identificada: _____PIRÁMIDE_____

Ingrese la cantidad de pirámides a evaluar: 3

Pirámide 1 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 7 6 9

Pirámide 1 - Ingrese la altura de la pirámide: 10

El volumen de la pirámide 1 es: 630.00

Pirámide 2 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 8 12 7

Pirámide 2 - Ingrese la altura de la pirámide: 5

El volumen de la pirámide 2 es: 560.00

Pirámide 3 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 4 4 2

Pirámide 3 - Ingrese la altura de la pirámide: 5

El volumen de la pirámide 3 es: 26.67

Promedio de volúmenes de las 3 pirámides es 405.56

Ingrese el tipo de poliedro a analizar: P

Ingrese la cantidad de bases que tendrán las figuras a evaluar: 2

Figura identificada: _____PRISMA _____

Ingrese la cantidad de prismas a evaluar: 4

Prisma 1 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 8 6 12

Prisma 1 - Ingrese la altura de la pirámide: 5

El volumen del prisma 1 es: 1440.00

Prisma 2 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 3 6 3

Prisma 2 - Ingrese la altura de la pirámide: 3

El volumen del prisma 2 es: 81.00

Prisma 3 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 7 4 2

Prisma 3 - Ingrese la altura de la pirámide: 8

El volumen del prisma 3 es: 224.00

Prisma 4 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 4 5 3

Prisma 4 - Ingrese la altura de la pirámide: 5

El volumen del prisma 4 es: 150.00

Promedio de volúmenes de los 4 prismas es 473.75

Ingrese el tipo de poliedro a analizar: P

Ingrese la cantidad de bases que tendrán las figuras a evaluar: 2

Figura identificada: _____PRISMA _____

Ingrese la cantidad de prismas a evaluar: 2

Prisma 1 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 6 8 4

Prisma 1 - Ingrese la altura de la pirámide: 6

El volumen del prisma 1 es: 576.00

Prisma 2 - Sobre la base, ingrese la cantidad de lados, el valor del lado y la apotema: 6 9 12

Prisma 2 - Ingrese la altura de la pirámide: 7

El volumen del prisma 2 es: 2268.00

Promedio de volúmenes de los 2 prismas es 1422.00

Ingrese el tipo de poliedro a analizar: R

Ingrese el tipo de poliedro a analizar: 2

Ingrese el tipo de poliedro a analizar: F

REPORTE

El total de poliedros es: 9

El porcentaje de pirámides analizadas es: 33.00

El porcentaje de prismas analizados es: 66.00

El menor valor de promedio de volumen de los poliedros es: 405.56

El mayor valor de promedio de volumen de los poliedros es: 1422.00

Programa 3: Propuesta de solución - Pirámides y Prismas

```

1  #include<stdio.h>
2
3  int EsFiguraValida(char figura);
4  void ObtenerValoresFiguras(int *tipoPoliedro,double *promedioVolumen, int *cantPiramides,int *cantPrismas);
5  double HallarAreaBase(int nLadosBase,double valorLado,double apotema);
6
7  int main(){
8      int fin, cantPiramides,cantPrismas,figuraValida,tipoPoliedro,totalPiramides,totalPrismas;
9      char figura;
10     double mayorVolumen,menorVolumen,promedioVolumen,porcPiramides,porcPrismas;
11
12     mayorVolumen=0;
13     menorVolumen=10000;
14     fin=0;
15     totalPiramides=0;
16     totalPrismas=0;
17     while(!fin){
18         printf("Ingrese el tipo de poliedro a analizar: ");
19         scanf("\n %c",&figura);
20         figuraValida=EsFiguraValida(figura);
21         if (figuraValida==1){
22             ObtenerValoresFiguras(&tipoPoliedro,&promedioVolumen, &cantPiramides,&cantPrismas);
23             if (tipoPoliedro==1){
24                 totalPiramides+=cantPiramides;
25             }
26             else{
27                 totalPrismas+=cantPrismas;
28             }
29             /*Hallar figura de mayor volumen y menor volumen*/
30             if (promedioVolumen>mayorVolumen){
31                 mayorVolumen=promedioVolumen;
32             }
33             if (promedioVolumen<menorVolumen){
34                 menorVolumen=promedioVolumen;
35             }
36             printf("\n");
37         }
38         else if (figuraValida==-1){
39             fin=1;
40         }
41     }
42     if (fin){
43         porcPiramides=100*totalPiramides/(totalPiramides+totalPrismas);
44         porcPrismas=100*totalPrismas/(totalPiramides+totalPrismas);
45         printf("\nREPORTE\n");
46         printf("-----\n");
47         printf("El total de poliedros es: %d\n",(totalPiramides+totalPrismas));
48         printf("El porcentaje de pirámides analizadas es: %.2lf %\n",porcPiramides);
49         printf("El porcentaje de prismas analizados es: %.2lf %\n",porcPrismas);
50         printf("El menor valor de promedio de volumen de los poliedros es: %.2lf\n",menorVolumen);
51         printf("El mayor valor de promedio de volumen de los poliedros es: %.2lf\n",mayorVolumen);
52     }
53     return 0;
54 }
55
56 int EsFiguraValida(char figura){
57     int valida;
58     if (figura=='P'){
59         return 1;
60     }
61     else if (figura=='F'){
62         return -1;
63     }
64     else{
65         return 0;

```



```

66     }
67 }
68
69 void ObtenerValoresFiguras(int *tipoPoliedro,double *promedioVolumen, int *cantPiramides,int *cantPrismas){
70     int nBases,nPiramides,nPrismas,i,j,nLadosBase;
71     double sumaVolumen,valorLado,apotema,altura,areaBase,volumen;
72
73     printf("Ingrese la cantidad de bases que tendrán las figuras a evaluar: ");
74     scanf("%d",&nBases);
75     /*tipoPoliedro -> 1 si es pirámide y 2 si es prisma*/
76     if (nBases==1){
77         *tipoPoliedro=1;
78         printf("Figura identificada: -----PIRÁMIDE-----\n");
79         printf("Ingrese la cantidad de pirámides a evaluar: ");
80         scanf("%d",&nPiramides);
81         /*por cada pirámide, se debe solicitar datos para hacer los cálculos*/
82         sumaVolumen=0;
83         for (i=1;i<=nPiramides;i++){
84             printf("Pirámide %d – Sobre la base, ingrese la cantidad de lados, el valor del lado y el apotema: ",i);
85             scanf("%d %lf %lf",&nLadosBase,&valorLado,&apotema);
86             if (nLadosBase>=3 && nLadosBase<=8){
87                 printf("Pirámide %d – Ingrese la altura de la pirámide: ",i);
88                 scanf("%lf",&altura);
89                 areaBase= HallarAreaBase(nLadosBase,valorLado,apotema);
90                 volumen=areaBase*altura/3;
91                 printf("El volumen de la pirámide %d es: %.2lf\n",i,volumen);
92                 sumaVolumen+=volumen;
93             }
94             else{
95                 printf("Ingresó una incorrecta cantidad de lados.\n");
96             }
97         }
98         *promedioVolumen=sumaVolumen/nPiramides;
99         printf("Promedio de volúmenes de las %d pirámides es %.2lf\n",nPiramides,*promedioVolumen);
100         *cantPiramides=nPiramides;
101         *cantPrismas=0;
102     }
103     else if (nBases==2){
104         *tipoPoliedro=2;
105         printf("Figura identificada: -----PRISMA -----\n");
106         printf("Ingrese la cantidad de prismas a evaluar: ");
107         scanf("%d",&nPrismas);
108         /*por cada pirámide, se debe solicitar datos para hacer los cálculos*/
109         sumaVolumen=0;
110         for (j=1;j<=nPrismas;j++){
111             printf("Prisma %d – Sobre la base, ingrese la cantidad de lados, el valor del lado y el apotema: ",j);
112             scanf("%d %lf %lf",&nLadosBase,&valorLado,&apotema);
113             if (nLadosBase>=3 && nLadosBase<=8){
114                 printf("Prisma %d – Ingrese la altura de la pirámide: ",j);
115                 scanf("%lf",&altura);
116                 areaBase= HallarAreaBase(nLadosBase,valorLado,apotema);
117                 volumen=areaBase*altura;
118                 printf("El volumen del prisma %d es: %.2lf\n",j,volumen);
119                 sumaVolumen+=volumen;
120             }
121             else{
122                 printf("Ingresó una incorrecta cantidad de lados.\n");
123             }
124         }
125         *promedioVolumen=sumaVolumen/nPrismas;
126         printf("Promedio de volúmenes de los %d prismas es %.2lf\n",nPrismas,*promedioVolumen);
127         *cantPiramides=0;
128         *cantPrismas=nPrismas;
129     }
130     else{
131         printf("Poliedro no identificado");
132     }

```

```

133 }
134
135 double HallarAreaBase(int nLadosBase, double valorLado, double apotema){
136     return ((nLadosBase*valorLado)*apotema/2);
137 }

```

4. Trayectoria de Projectiles

Se conoce que la trayectoria de cualquier proyectil sigue la forma de una parábola invertida. Esta trayectoria está en función de la velocidad inicial del disparo (V_0), así como del ángulo con la horizontal (Alfa) con el que es disparado el proyectil.

Sabiendo lo anterior, se desea conocer para un grupo determinado de proyectiles que serán lanzados con velocidades (V_0) y ángulos (Alfa) diferentes, cuál será su posición X, Y y sus componentes de velocidad horizontal (V_x) y vertical (V_y) en un número determinado de mediciones que se realizará durante la trayectoria de cada proyectil. Se deberá tener presente que cada medición se realizara en lapsos de tiempo iguales, debiendo la última medición ser tomada en el momento que el proyectil toque tierra (ver figura). Por último, se necesita saber para cada proyectil cual fue la mayor medición de X en el eje horizontal (X_{max}) y de Y en el eje vertical (Y_{max}).

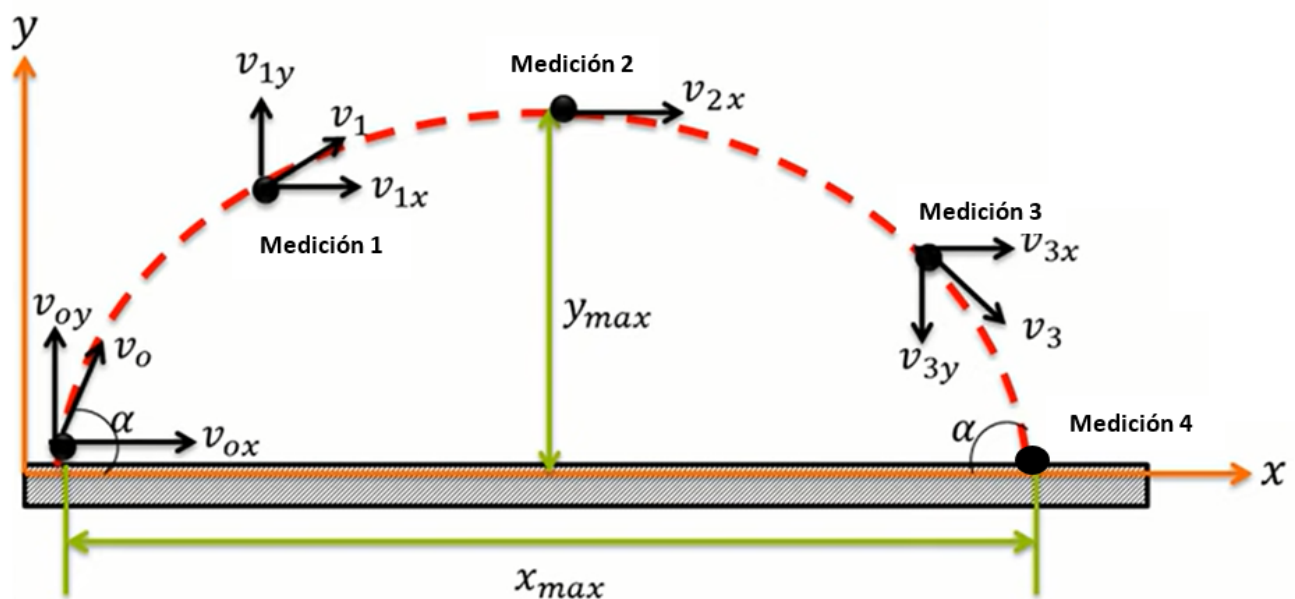


Figura 1: Ejemplo de la Trayectoria de un Disparo

Se pide desarrollar un programa en Lenguaje C que en el módulo principal solicite la cantidad de disparos que se ejecutarán y la cantidad de mediciones que se evaluarán en cada trayectoria. Se deberá validar que el número de disparos sea mayor que cero (0) y el número de mediciones sea mayor o igual a cuatro (4). En caso no cumpliera cualquiera de dichos criterios deberá indicar un mensaje de error correspondiente.

De ingresar los datos correctamente, deberá invocar a un módulo que reciba como parámetros el número de disparos a realizar, así como la cantidad de mediciones que se efectuarán en cada disparo. Con dichos datos, deberá solicitar para cada disparo la velocidad inicial (V_0) y el ángulo del disparo (alfa) en grados, validando que la velocidad inicial sea mayor que cero y el ángulo del disparo mayor a 0 grados y menor a 90 grados. Si no fuera así, deberá mostrar un mensaje de error y volver a solicitar el ingreso de los datos del disparo.

Con la velocidad inicial y el ángulo de cada disparo, deberá calcular el periodo de tiempo en el que se espaciarán las mediciones, para lo cual deberá dividir el tiempo total de la trayectoria del proyectil entre las mediciones indicadas. Debiendo obtener para cada medición, el tiempo en el que se realizó, los componentes de la velocidad en el eje horizontal (V_x) y vertical (V_y) así como la posición X y Y del proyectil. Por último, deberá mostrar la máxima posición en el eje horizontal (X_{max}) y vertical (Y_{max}) logrado en cada disparo, así como los tiempos en que fueron tomadas dichas mediciones.

Deberá considerar que la solución del programa cuente con al menos cuatro módulos incluido el módulo principal.

Para resolver el problema podrá utilizar las siguientes consideraciones:

$$\text{Tiempo Total de la Trayectoria del Proyectil} = 2 * V_o * \frac{\sin(\text{Ángulo en Radianes})}{GRAVEDAD}$$

$$V_x = V_o * \cos(\text{Ángulo en Radianes})$$

$$V_y = V_o * \sin(\text{Ángulo en Radianes}) - GRAVEDAD * \text{Tiempo}$$

$$X = V_x * \text{Tiempo}$$

$$Y = V_y * \text{Tiempo} + \frac{GRAVEDAD * \text{Tiempo}^2}{2}$$

$$\text{Ángulo en Radianes} = \frac{PI * \text{Ángulo en Grados}}{180}$$

$$PI = 3.14159 \quad y \quad GRAVEDAD = 9.80 \frac{m}{s^2}$$

Caso de prueba 1

Ingrese el número de disparos ejecutados: -4

Ingrese el número de mediciones por disparo: 4

El número de disparos debe ser mayor que cero (0).

Caso de prueba 2

Ingrese el número de disparos ejecutados: 6

Ingrese el número de mediciones por disparo: 3

El número de mediciones debe ser mayor igual que cuatro (4).

Caso de prueba 3

Ingrese el número de disparos ejecutados: 1

Ingrese el número de mediciones por disparo: 6

Ingrese los datos del disparo número 1:

- Angulo (alfa) en grados: 120

- Velocidad Inicial (V_0) en m/s: 5

Debe ingresar un ángulo mayor que 0 grados y menor que 90 grados.

Ingrese los datos del disparo número 1:

- Angulo (alfa) en grados: 50

- Velocidad Inicial (V_0) en m/s: -10

Debe ingresar una velocidad inicial mayor que 0

Ingrese los datos del disparo número 1:

- Angulo (alfa) en grados: 40

- Velocidad Inicial (V_0) en m/s: 25

Las mediciones del disparo número 1 son:

- 1: Tiempo: 0.55 s, V_x : 19.15 m/s, V_y : 10.71 m/s, X: 10.47 m, Y: 7.32 m
- 2: Tiempo: 1.09 s, V_x : 19.15 m/s, V_y : 5.36 m/s, X: 20.94 m, Y: 11.71 m
- 3: Tiempo: 1.64 s, V_x : 19.15 m/s, V_y : 0.00 m/s, X: 31.40 m, Y: 13.18 m
- 4: Tiempo: 2.19 s, V_x : 19.15 m/s, V_y : -5.36 m/s, X: 41.87 m, Y: 11.71 m
- 5: Tiempo: 2.73 s, V_x : 19.15 m/s, V_y : -10.71 m/s, X: 52.34 m, Y: 7.32 m
- 6: Tiempo: 3.28 s, V_x : 19.15 m/s, V_y : -16.07 m/s, X: 62.81 m, Y: 0.00 m

El recorrido horizontal (X_{\max}) del disparo número 1 fue de 62.81 m y se logró en 3.28 s mientras que la altura máxima (Y_{\max}) alcanzada fue de 13.18 m y se logró a los 1.64 s

Caso de prueba 4

Ingrese el número de disparos ejecutados: 2

Ingrese el número de mediciones por disparo: 5

Ingrese los datos del disparo número 1:

- Angulo (alfa) en grados: 55

- Velocidad Inicial (V_0) en m/s: 8

Las mediciones del disparo número 1 son:

- 1: Tiempo: 0.27 s, V_x : 4.59 m/s, V_y : 3.93 m/s, X: 1.23 m, Y: 1.40 m
- 2: Tiempo: 0.53 s, V_x : 4.59 m/s, V_y : 1.31 m/s, X: 2.45 m, Y: 2.10 m
- 3: Tiempo: 0.80 s, V_x : 4.59 m/s, V_y : -1.31 m/s, X: 3.68 m, Y: 2.10 m
- 4: Tiempo: 1.07 s, V_x : 4.59 m/s, V_y : -3.93 m/s, X: 4.91 m, Y: 1.40 m
- 5: Tiempo: 1.34 s, V_x : 4.59 m/s, V_y : -6.55 m/s, X: 6.14 m, Y: 0.00 m

El recorrido horizontal del disparo número 1 fue de 6.14 m y se logró en 1.34 s mientras que la altura máxima alcanzada fue de 2.10 m y se logró a los 0.80 s

Ingrese los datos del disparo número 2:

- Angulo (alfa) en grados: 70

- Velocidad Inicial (V_0) en m/s: 10

Las mediciones del disparo número 2 son:

- 1: Tiempo: 0.38 s, V_x : 3.42 m/s, V_y : 5.64 m/s, X: 1.31 m, Y: 2.88 m
- 2: Tiempo: 0.77 s, V_x : 3.42 m/s, V_y : 1.88 m/s, X: 2.62 m, Y: 4.33 m
- 3: Tiempo: 1.15 s, V_x : 3.42 m/s, V_y : -1.88 m/s, X: 3.94 m, Y: 4.33 m
- 4: Tiempo: 1.53 s, V_x : 3.42 m/s, V_y : -5.64 m/s, X: 5.25 m, Y: 2.88 m
- 5: Tiempo: 1.92 s, V_x : 3.42 m/s, V_y : -9.40 m/s, X: 6.56 m, Y: 0.00 m

El recorrido horizontal del disparo (X_{\max}) número 2 fue de 6.56 m y se logró en 1.92 s mientras que la altura máxima (Y_{\max}) alcanzada fue de 4.33 m y se logró a los 0.77 s

Programa 4: Propuesta de solución - Trayectoria de Projectiles

```
1 #include <stdio.h>
2 #include <math.h>
3
4 //Constantes con las que trabajará el programa.
5 #define PI 3.14159
6 #define GRAVEDAD 9.8
7
8 //Declaración de la funciones utilizadas por el programa.
9 double ConvertirAnguloRadianes(double);
10 double CalcularTiempoTotal(double, double);
11 void Procesar_Trayectoria_De_Disparos(double, double);
12
13 int main ()
14 {
15     int Numero_Disparos, Numero_Mediciones;
16
17     //Solicita el número de disparos y mediciones para cada disparo.
18     printf("Ingrese el número de disparos ejecutados: ");
19     scanf("%d", &Numero_Disparos);
20     printf("Ingrese el número de mediciones por disparo: ");
21     scanf("%d", &Numero_Mediciones);
22
23     //Valida que Numero_Disparos sea mayor que 0 y Numero_Mediciones mayor igual a 4.
24     if ( Numero_Disparos > 0 && Numero_Mediciones >= 4 )
25         Procesar_Trayectoria_De_Disparos(Numero_Disparos, Numero_Mediciones);
26     else if ( Numero_Disparos <= 0 )
```

```

27         printf("\nEl numero de disparos debe ser mayor que cero (0).");
28     else if ( Numero_Mediciones < 5 )
29         printf("\nEl numero de mediciones debe ser mayor igual que cuatro (4).");
30
31     return 0;
32 }
33
34 //Recoge los datos de cada disparo y procede a realizar las diferentes mediciones.
35 void Procesar_Trayectoria_De_Disparos(double Numero_Disparos, double Numero_Mediciones)
36 {
37     int Cont_Disparos, Cont_Mediciones;
38     double Angulo_Grados, Angulo_Rad;
39     double Velocidad_Inicial, Vx, Vy;
40     double Tiempo_Total, Tiempo_Delta, Tiempo, Tiempo_Y_Max;
41     double X, Y, X_Max, Y_Max;
42
43     //Inicializa contador de disparos.
44     Cont_Disparos = 1;
45
46     //Mientras no se hayan evaluados todos los disparos indicados.
47     while (Cont_Disparos <= Numero_Disparos)
48     {
49         //Solicita los valores de cada disparo.
50         printf("\nIngresa los datos del disparo numero %d:\n", Cont_Disparos);
51         printf("– Angulo (alfa) en grados: ");
52         scanf("%lf", &Angulo_Grados);
53         printf("– Velocidad Inicial (Vo) en m/s: ");
54         scanf("%lf", &Velocidad_Inicial);
55
56         //Si el ángulo y la velocidad han sido ingresadas dentro de los rangos indicados.
57         if ( Angulo_Grados > 0 && Angulo_Grados < 90 && Velocidad_Inicial > 0 )
58         {
59             //Convierte a radianes el ángulo ingresado en grados.
60             Angulo_Rad = ConvertirAnguloRadianes(Angulo_Grados);
61
62             //Obtiene el tiempo total de la trayectoria del disparo.
63             Tiempo_Total = CalcularTiempoTotal(Velocidad_Inicial, Angulo_Rad);
64
65             //Obtiene el diferencial de tiempo de cada medición.
66             Tiempo_Delta = Tiempo_Total / Numero_Mediciones;
67
68             //Inicializa las variables Tiempo, X_Max, Y_Max y Cont_Mediciones en cero.
69             Tiempo = X_Max = Y_Max = Cont_Mediciones = 0;
70
71             //Imprime el encabezado de las mediciones de cada disparo.
72             printf("Las mediciones del disparo numero %d son:\n", Cont_Disparos);
73
74             while ( Cont_Mediciones < Numero_Mediciones )
75             {
76                 //Incrementa el tiempo el delta calculado y el contador de mediciones en uno (1).
77                 Tiempo += Tiempo_Delta;
78                 Cont_Mediciones++;
79
80                 //Obtiene las velocidades en X e Y de cada medición.
81                 Vx = Velocidad_Inicial * cos(Angulo_Rad);
82                 Vy = Velocidad_Inicial * sin(Angulo_Rad) – GRAVEDAD * Tiempo;
83
84                 //Obtiene el desplazamiento en X e Y de cada medición.
85                 X = Vx * Tiempo;
86                 Y = Vy * Tiempo + GRAVEDAD * pow(Tiempo, 2) / 2;
87
88                 //Verifica si aun no llega a la altura mayor.
89                 if ( Y > Y_Max )
90                 {
91                     Y_Max = Y;
92                     Tiempo_Y_Max = Tiempo;
93                 }
94             }
95         }
96     }
97 }

```

```

94         //Imprime los resultados de cada medición.
95         printf(" - # %d: ", Cont_Mediciones);
96         printf("Tiempo: %6.2lf s, Vx: %7.2lf m/s, Vy: %7.2lf m/s, X: %7.2lf m, Y: %7.2lf m\n", Tiempo,
97             Vx, Vy, X, Y);
98     }
99
100     //Se define el valor maximo alcanzado horizontalmente que es igual al valor de la última medición.
101     X_Max = X;
102
103     //Imprime el resultado final del disparo
104     printf("\nEl recorrido horizontal (Xmax) del disparo numero %d fue de %7.2lf m y se logro en %6.2lf s\n",
105         Cont_Disparos, X_Max, Tiempo.Total);
106     printf("mientras que la altura maxima (Ymax) alcanzada fue de %7.2lf m y se logro a los %6.2lf s\n", Y_Max
107         , Tiempo.Y_Max );
108
109     //Incrementa en uno (1) el contador de disparos.
110     Cont_Disparos ++;
111 }
112 else if ( Angulo_Grados <= 0 || Angulo_Grados >= 90 )
113     printf("Debe ingresar un angulo mayor que 0 grados y menor que 90 grados.");
114 else
115     printf("Debe ingresar una velocidad inicial mayor que 0");
116 }
117 }
118
119 //Recibe un ángulo en grados y devuelve su conversión en radianes.
120 double ConvertirAnguloRadianes(double Angulo_Grados)
121 {
122     //Calcula el ángulo en Radianes.
123     double Angulo_Rad = Angulo_Grados * PI / 180;
124
125     //Retorna el valor caculado del ángulo en radianes.
126     return Angulo_Rad;
127 }
128
129 //Dada la velocidad inicial y ángulo de un disparo calcula el tiempo total de la Trayectoria.
130 double CalcularTiempoTotal(double Velocidad_Inicial, double Angulo_Rad)
131 {
132     //Calcula el tiempo total de la trayectoria del disparo.
133     double Tiempo_Total = 2 * Velocidad_Inicial * sin(Angulo_Rad) / GRAVEDAD;
134
135     //Retorna el valor del tiempo total de la trayectoria.
136     return Tiempo_Total;
137 }

```

Puede usar cualquier estructura selectiva por lo que el uso de la estructura selectiva múltiple queda a su criterio

Puede usar cualquier estructura anidada