

Rojas Badillo Emilio , 23 2.2Ejercicios en C: 1,2,4 p48  
Ej p48e1N\_6A\_23

Fundamentos en programacion

```
17 September 05:07 PM 40% 65%  
p48e1N_6A_23.c x  
1 /*Nombre del Programa: p48e1N_6A_n1.c  
2 Elaboró: Rojas Badillo Emilio  
3 No Lista: 23  
4 Fecha: 17-9-25  
5 No. Versión: 1  
6 Descripción: Ecuacion movimiento uniformemente acelerado  
7 programa en Lenguaje C */  
8 #include <math.h> // Incluye cabecera en el archivo  
9 #include <stdio.h> // Incluye cabecera en el archivo  
10  
11 int main(int argc, char *argv[]) {  
12     int v0, t, d0; // declaracion de variables locales enteros  
13     float a; // declaracion de variables locales flotantes  
14     printf("Ingrese velocidad inicial [m/s]: "); // Imprime pedir  
15     scanf("%d", &v0); // Almacena en variable el d  
16     printf("Ingrese tiempo [s]: "); // Imprime pedir datos  
17     scanf("%d", &t); // Almacena en variable el d  
18     printf("Ingrese desplazamiento inicial [m]: "); // Imprime p  
19     scanf("%d", &d0); // Almacena en variable el dato ingresado  
20     printf("Ingrese aceleracion [m/s^2]: "); // Imprime pedir da  
21     scanf("%f", &a); // Almacena en variable el dato ingresado  
22     float df =  
23     d0 + (v0 * t) + (a * t * t / 2); // Hace calculo con los  
24     printf("El desplazamiento final es: %.2f\n", df); // Imprime  
25     return 0;  
26 }  
NORMAL p48e1N_6A_23.c % main 26/1  
[emilio@user1012007][~]$ cd Documents/ITQ/semestre-1/  
1.1/ 1.2/ 1.3/ 1.3.2/ 2.1/ 2.2/ .git/  
[emilio@user1012007][~]$ cd Documents/ITQ/semestre-1/  
1.1/ 1.2/ 1.3/ 1.3.2/ 2.1/ 2.2/ .git/  
[emilio@user1012007][~]$ cd Documents/ITQ/semestre-1/2.2/ej1  
[emilio@user1012007][ej1]$ ls  
flake.lock flake.nix p48e1N_6A_23.c  
[emilio@user1012007][ej1]$ nix run  
warning: Git tree '/home/emilio/Documents/ITQ/semestre-1' is dirty  
Ingrese velocidad inicial [m/s]: 20  
Ingrese tiempo [s]: 10  
Ingrese desplazamiento inicial [m]: 10  
Ingrese aceleracion [m/s^2]: 1  
El desplazamiento final es: 260.00  
[emilio@user1012007][ej1]$
```

ej p48e2N\_6A\_23

```
17 September 05:09 PM 40% 66%  
p48e1N_6A_23.c x p48e2N_6A_23.c x  
1 /*Nombre del Programa: p48e2N_6A_n1.c  
2 Elaboró: Rojas Badillo Emilio  
3 No Lista: 23  
4 Fecha: 17-9-25  
5 No. Versión: 1  
6 Descripción: Ecuaciones lineales  
7 programa en Lenguaje C */  
8 #include <stdio.h> // Incluye cabecera en el archivo  
9  
10 int main(int argc, char *argv[]) {  
11     int a, b, c, d, e, f; // declaracion de variables locales enter  
12     printf("Ingrese A: \n"); // Imprime pedir dato  
13     scanf("%d", &a); // Almacena dato en memoria  
14     printf("Ingrese B: \n"); // Imprime pedir dato  
15     scanf("%d", &b); // Almacena dato en memoria  
16     printf("Ingrese C: \n"); // Imprime pedir dato  
17     scanf("%d", &c); // Almacena dato en memoria  
18     printf("Ingrese D: \n"); // Imprime pedir dato  
19     scanf("%d", &d); // Almacena dato en memoria  
20     printf("Ingrese E: \n"); // Imprime pedir dato  
21     scanf("%d", &e); // Almacena dato en memoria  
22     printf("Ingrese F: \n"); // Imprime pedir dato  
23     scanf("%d", &f); // Almacena dato en memoria  
24  
25     float x = (c * e - b * f) / (a * e - b * d); // Calculo de datos  
26     printf("X es: %.2f\n", x); // Imprime calculo fi  
27     float y = (a * f - c * d) / (a * e - b * d); // Calculo de datos  
28     printf("Y es: %.2f\n", y); // Imprime calculo fi  
29     return 0;  
30 }  
NORMAL p48e2N_6A_23.c % main 11 6 3 16/41  
[emilio@user1012007][ej2]$ nix run  
warning: Git tree '/home/emilio/Documents/ITQ/semestre-1' is dirty  
Ingrese A:  
2  
Ingrese B:  
3  
Ingrese C:  
4  
Ingrese D:  
5  
Ingrese E:  
6  
Ingrese F:  
7  
X es: -1.00  
Y es: 2.00  
[emilio@user1012007][ej2]$
```

ej p48e4N\_6A\_23

```
p48e1N_6A_23.c x p48e2N_6A_23.c x p48e4N_6A_23.c x
1 /*Nombre del Programa: p48e4N_6A_n1.c
2 Elaboró: Rojas Badillo Emilio
3 No Lista: 23
4 Fecha: 17-9-25
5 No. Versión: 1
6 Descripción: Area y volumen de figuras geometricas
7 programa en Lenguaje C */
8 #include <math.h> // Incluye cabecera en el archivo
9 #include <stdio.h> // Incluye cabecera en el archivo
10
11 int main(int argc, char *argv[]) {
12     int l1 = 0, l2 = 0, l3 = 0, r = 0,
13     h = 0; // Declaracion de variables locales enteras
14     float a, v; // Declaracion de variables locales flotantes
15     float PI = 3.1416; // Declaracion de constate pi
16     printf("\033[2J\033[H");
17     printf("Ingrese l1: \n");
18     scanf("%d", &l1); // Almacena dato ingresado
19     en memoria
20     a = 6 * pow(l1, 2); // Calculo de area
21     v = pow(l1, 3); // Calculo de volumen
22     printf("Area de cubo = %.2f \n", a); // Imprimir area
23     printf("Volumen de cubo = %.2f \n", v); // Imprimir volumen
24
25     printf("Ingrese l1: \n"); // Imprimir pedir datos
26     scanf("%d", &l1); // Almacena dato ingresado en memoria
27     printf("Ingrese l2: \n"); // Imprimir pedir datos
28     scanf("%d", &l2); // Almacena dato ingresado en memoria
29     printf("Ingrese l3: \n"); // Imprimir pedir datos
30     scanf("%d", &l3); // Almacena dato ingresado en memoria
31
32     a = 2 * l1 * l2 + 2 * l1 * l3 + 2 * l2 * l3; // Calculo de area
33     v = l1 * l2 * l3; // Calculo de volumen
34     printf("Area de paralelepipedo recto rectangular = %.2f \n",
35     a); // Imprimir area
36     printf("Volumen de paralelepipedo recto rectangular = %.2f \n",
37     v); // Imprimir volumen
38
39     printf("Ingrese el radio: \n"); // Imprimir pedir datos
40     scanf("%d", &r); // Almacena dato ingresado en memoria
41     printf("Ingrese la altura: \n"); // Imprimir pedir datos
42     scanf("%d", &h); // Almacena dato ingresado en memoria
43
44     a = (2 * PI * pow(r, 2)) + (2 * PI * r * h); // Calculo
45     de area
46     v = PI * h * pow(r, 2); // Calculo
47     de volumen
48     printf("Area de cilindro recto circular = %.2f \n", a); // Imprimir
49     r area
50     printf("Volumen de cilindro recto circular= %.2f \n", v); // Imprimir
51     volumen
52
53     printf("Ingrese el radio: \n"); // Imprimir pedir datos
54     scanf("%d", &r); // Almacena dato ingresado en memoria
55     printf("Ingrese la altura: \n"); // Imprimir pedir datos
56     scanf("%d", &h); // Almacena dato ingresado en memoria
57
58     printf("Ingrese l1: \n"); // Imprimir pedir datos
59     scanf("%d", &l1); // Almacena dato ingresado en memoria
60
61     a = (PI * pow(r, 2)) + (PI * r * l1); // Calculo de area
62     v = (PI * h * pow(r, 2)) / 3; // Calculo de volumen
63     printf("Area de cono recto circular = %.2f \n", a); // Imprimir
64     area
65     printf("Volumen de cono recto circular= %.2f \n", v); // Imprimir
66     volumen
67
68     return 0;
69 }
```

Ingrese l1:  
5  
Area de cubo = 150.00  
Volumen de cubo = 125.00  
Ingrese l1:  
3  
Ingrese l2:  
4  
Ingrese l3:  
5  
Area de paralelepipedo recto rectangular = 94.00  
Volumen de paralelepipedo recto rectangular = 60.00  
Ingrese el radio:  
5  
Ingrese la altura:  
6  
Area de cilindro recto circular = 345.58  
Volumen de cilindro recto circular= 471.24  
Ingrese el radio:  
5  
Ingrese la altura:  
10  
Ingrese l1:  
34  
Area de cono recto circular = 612.61  
Volumen de cono recto circular= 261.80  
[emilio@user1012007][ej3]{main}\$

```
p48e1N_6A_23.c x p48e2N_6A_23.c x p48e4N_6A_23.c x
37
38 printf("Ingrese el radio: \n"); // Imprimir pedir datos
39 scanf("%d", &r); // Almacena dato ingresado en memoria
40
41 printf("Ingrese la altura: \n"); // Imprimir pedir datos
42 scanf("%d", &h); // Almacena dato ingresado en memoria
43
44 a = (2 * PI * pow(r, 2)) + (2 * PI * r * h); // Calculo
45 de area
46 v = PI * h * pow(r, 2); // Calculo
47 de volumen
48 printf("Area de cilindro recto circular = %.2f \n", a); // Imprimir
49 r area
50 printf("Volumen de cilindro recto circular= %.2f \n", v); // Imprimir
51 volumen
52
53 printf("Ingrese el radio: \n"); // Imprimir pedir datos
54 scanf("%d", &r); // Almacena dato ingresado en memoria
55 printf("Ingrese la altura: \n"); // Imprimir pedir datos
56 scanf("%d", &h); // Almacena dato ingresado en memoria
57
58 printf("Ingrese l1: \n"); // Imprimir pedir datos
59 scanf("%d", &l1); // Almacena dato ingresado en memoria
60
61 a = (PI * pow(r, 2)) + (PI * r * l1); // Calculo de area
62 v = (PI * h * pow(r, 2)) / 3; // Calculo de volumen
63 printf("Area de cono recto circular = %.2f \n", a); // Imprimir
64 area
65 printf("Volumen de cono recto circular= %.2f \n", v); // Imprimir
66 volumen
67
68 return 0;
69 }
```

Ingrese l1:  
5  
Area de cubo = 150.00  
Volumen de cubo = 125.00  
Ingrese l1:  
3  
Ingrese l2:  
4  
Ingrese l3:  
5  
Area de paralelepipedo recto rectangular = 94.00  
Volumen de paralelepipedo recto rectangular = 60.00  
Ingrese el radio:  
5  
Ingrese la altura:  
6  
Area de cilindro recto circular = 345.58  
Volumen de cilindro recto circular= 471.24  
Ingrese el radio:  
5  
Ingrese la altura:  
10  
Ingrese l1:  
34  
Area de cono recto circular = 612.61  
Volumen de cono recto circular= 261.80  
[emilio@user1012007][ej3]{main}\$ ls  
flake.lock flake.nix p48e4N\_6A\_23.c  
[emilio@user1012007][ej3]{main}\$