



TECNOLOGÍA Y SOCIEDAD

PROF. YULITH VANESSA ALTAMIRANO FLORES

TALLER #2
CONCEPTOS GENERALES

KEVIN ALEJANDRO GONZALEZ TORRES
GRUPO 932

1.- Definir dos variables de tipo flotante, x y y. Asignarles valores iniciales.

```
C: > Users > kevin > Downloads > 123.cpp > ...
1  #include<stdio.h>
2
3  int main()
4  {
5      float x=5, y=7;
6
7  }
8  return 0;
9
```

2.- Calcular y mostrar el resultado de la expresión $(x + 2 * y) / (3 * x - y)$

```
C: > Users > kevin > Downloads > 123.cpp > ...
1  #include<stdio.h>
2
3  int main()
4  {
5      float x=5, y=7, result;
6
7      result = (x + 2 * y) / (3 * x - y);
8      printf("%i", result);
9  }
10 return 0;
11
```

3.- Usar `#define` para definir una constante simbólica llamada `TASA_DESCUENTO` con el valor 0.1. Calcular y mostrar el precio después de aplicar un descuento del 10% a un producto.

```
C: > Users > kevin > Downloads > 123.cpp > ...
1  #include<stdio.h>
2  #define TASA_DESCUENTO 0.1
3
4  int main()
5  {
6      float producto;
7      printf("Ingresa el precio del producto:");
8      scanf("%f", &producto);
9      printf("El precio del producto con descuento es: %f", (producto-(producto*TASA_DESCUENTO)));
10     return 0;
11 }
12
```

4.- Pedir al usuario que ingrese un número entero positivo `n`. Imprimir los primeros `n` números primos.

```

1  #include <stdio.h>
2  #include <stdbool.h>
3
4  bool esPrimo(int num) {
5      if (num <= 1) {
6          return false;
7      }
8      for (int i = 2; i * i <= num; i++) {
9          if (num % i == 0) {
10             return false;
11         }
12     }
13     return true;
14 }
15
16 int main() {
17     int n;
18     printf("Ingrese un número entero positivo: ");
19     scanf("%d", &n);
20
21     if (n <= 0) {
22         printf("Ingrese un número entero positivo válido.\n");
23         return 1;
24     }
25
26     printf("Los primeros %d números primos son:\n", n);
27
28     int cantidadEncontrados = 0;
29     int numero = 2; // Comenzamos desde el primer número primo
30
31     while (cantidadEncontrados < n) {
32         if (esPrimo(numero)) {
33             printf("%d ", numero);
34             cantidadEncontrados++;
35         }
36         numero++;
37     }
38
39     return 0;
40 }

```

5.- Implementar una calculadora que pueda realizar operaciones de suma, resta, multiplicación y división. El usuario debe ingresar dos números y el operador.

```

71 float basic_calculator()
72 {
73     int option;
74     float num1, num2, result;
75
76     printf("CALCULADORA DE OPERACIONES BASICAS\n\n");
77     printf("1.- Suma\n");
78     printf("2.- Resta\n");
79     printf("3.- Multiplicacion\n");
80     printf("4.- Division\n\n");
81
82     printf("Digita tus 2 numeros\n");
83     printf("Numero 1: ");
84     scanf("%f", &num1);
85     printf("Numero 2: ");
86     scanf("%f", &num2);
87     printf("\n\n");
88
89     printf("Selecciona tu operacion: ");
90     scanf("%i", &option);
91
92     switch (option)
93     {
94     case 1:
95         result = num1 + num2;
96         break;
97
98     case 2:
99         result = num1 - num2;
100        break;
101
102     case 3:
103         result = num1 * num2;
104         break;
105
106     case 4:
107         if (num2 != 0)
108         {
109             result = num1 / num2;
110         }
111         else
112         {
113             printf("ERROR - NO SE PUEDE DIVIDIR ENTRE CERO\n");
114         }
115         break;
116
117     default:
118         printf("ERROR - ESTA NO ES UNA OPCIÓN VALIDA");
119         break;
120     }
121     return result;

```

6.- Implementar un programa que simule un juego de adivinar el número. El programa debe generar un número aleatorio entre 1 y 100 y permitir al usuario adivinar hasta

que lo acierte. Dar pistas como "Demasiado alto" o "Demasiado bajo".

```
33 srand(time(NULL));
34 random_num = rand() % 101 + 1;
35
36 printf("Numero aleatorio generado\n;Trata de adivinar el numero!\n\n");
37
38 while (correct_num != random_num)
39 {
40     printf("Numero: ");
41     scanf("%i", &correct_num);
42
43     if (correct_num > random_num)
44     {
45         printf(";Mas bajo!\n");
46     }
47
48     else if (correct_num < random_num)
49     {
50         printf(";Mas alto!\n");
51     }
52
53     else
54     {
55         printf(";Numero correcto, el numero era: %i", random_num);
56     }
57 }
58 break;
```

7.- Pedir al usuario que ingrese un número. Calcular el factorial del número utilizando una función recursiva.

```
1  #include <stdio.h>
2
3  unsigned int long factorial(int num) {
4      if (num == 0 || num == 1) {
5          return 1;
6      }
7      return num * factorial(num - 1);
8  }
9
10 int main() {
11     int numero;
12     printf("Ingrese un número: ");
13     scanf("%d", &numero);
14
15     if (numero < 0) {
16         printf("El factorial no está definido para números negativos.\n");
17     } else {
18         unsigned long long resultado = factorial(numero);
19         printf("El factorial de %d es %llu.\n", numero, resultado);
20     }
21
22     return 0;
23 }
```

8.- Definir un arreglo de enteros llamado calificaciones con capacidad para 10 elementos.

Pedir al usuario que ingrese 10 calificaciones y almacenarlas en el arreglo. Luego,

mostrar las calificaciones ingresadas y sacar el promedio. Y Ordenar el arreglo calificaciones en orden ascendente.

```
1  #include <stdio.h>
2
3  void ordenarAscendente(int arr[], int n) {
4      for (int i = 0; i < n - 1; i++) {
5          for (int j = 0; j < n - i - 1; j++) {
6              if (arr[j] > arr[j + 1]) {
7                  int temp = arr[j];
8                  arr[j] = arr[j + 1];
9                  arr[j + 1] = temp;
10             }
11         }
12     }
13 }
14
15 int main() {
16     int calificaciones[10];
17     int totalCalificaciones = 10;
18
19     printf("Ingrese 10 calificaciones:\n");
20     for (int i = 0; i < totalCalificaciones; i++) {
21         printf("Calificación %d: ", i + 1);
22         scanf("%d", &calificaciones[i]);
23     }
24
25     printf("Calificaciones ingresadas:\n");
26     for (int i = 0; i < totalCalificaciones; i++) {
27         printf("%d ", calificaciones[i]);
28     }
29
30     // Calcular el promedio
31     int suma = 0;
32     for (int i = 0; i < totalCalificaciones; i++) {
33         suma += calificaciones[i];
34     }
35     float promedio = (float)suma / totalCalificaciones;
36     printf("\nPromedio: %.2f\n", promedio);
37
38     // Ordenar el arreglo de calificaciones en orden ascendente
39     ordenarAscendente(calificaciones, totalCalificaciones);
40
41     printf("Calificaciones ordenadas:\n");
42     for (int i = 0; i < totalCalificaciones; i++) {
43         printf("%d ", calificaciones[i]);
44     }
45
46     return 0;
47 }
48
```


10.- Definir una función llamada calcularPotencia() que tome dos argumentos: la base b y el exponente e, y devuelva el resultado de b elevado a la e.

```
1  #include <stdio.h>
2
3  double calcularPotencia(double base, int exponente) {
4      double resultado = 1.0;
5
6      for (int i = 0; i < exponente; i++) {
7          resultado *= base;
8      }
9
10     return resultado;
11 }
12
13 int main() {
14     double base;
15     int exponente;
16
17     printf("Ingrese la base: ");
18     scanf("%lf", &base);
19
20     printf("Ingrese el exponente: ");
21     scanf("%d", &exponente);
22
23     double resultado = calcularPotencia(base, exponente);
24     printf("%.2lf elevado a la %d es %.2lf\n", base, exponente, resultado);
25
26     return 0;
27 }
28
```