Guia para While

```
// ========================================================
// 1. Ciclos básicos: Diferencia entre while y do-while
// ========================================================
void ejemploWhileYDoWhile() {
    int i = 0;
    while (i > 5) {
        printf("(while) i = %d\n", i);
        i++;
    }
    do {
        printf("(do-while) i = %d\n", i);
        i++;
    } while (i > 5);
}
```

```
// ========================================================
// 2. Imprimir del 1 al n con while
// ========================================================
void imprimirDel1AlN(int n) {
    int i = 1;
    while (i <= n) {
        printf("%d ", i);
        i++;
    }
    printf("\n");
}
```

```
// ========================================================
// 3. Suma de números hasta n
// ========================================================
```



```
int sumaHastaN(int n) {
    int suma = 0, i = 1;
    while (i <= n) {
        suma += i;
        i++;
    }
    return suma;
}
```

```
// ========================================================
// 4. Contar dígitos de un número
// ========================================================
int contarDigitos(int n) {
    int contador = 0;
    do {
        contador++;
        n /= 10;
    } while (n > 0);
    return contador;
}
```

```
// ========================================================
// 5. Invertir número con do-while
// ========================================================
int invertirNumero(int n) {
    int invertido = 0;
    do {
        invertido = invertido * 10 + (n % 10);
        n /= 10;
    } while (n > 0);
    return invertido;
}
```



```
// ========================================================
// 6. Calcular factorial con while
// ========================================================
int factorial(int n) {
    int resultado = 1;
    while (n > 1) {
        resultado *= n;
        n--;
    }
    return resultado;
}
```

```
// ========================================================
// 7. Serie Fibonacci hasta n
// ========================================================
void fibonacci(int n) {
    int a = 0, b = 1, temp;
    while (a <= n) {
        printf("%d ", a);
        temp = a + b;
        a = b;
        b = temp;
    }
    printf("\n");
}
```

```
// ========================================================
// 8. Verificar si un número es primo
// ========================================================
void esPrimo(int n) {
    int i = 2;
    if (n <= 1) {
        printf("%d no es primo\n", n);
        return;
```



```
    }
    while (i * i <= n) {
        if (n % i == 0) {
            printf("%d no es primo\n", n);
            return;
        }
        i++;
    }
    printf("%d es primo\n", n);
}
```

```
// ========================================================
// 9. Triángulo de Pascal con while
// ========================================================
int combinatoria(int n, int k) {
    int res = 1, i = 1;
    while (i <= k) {
        res = res * (n - i + 1) / i;
        i++;
    }
    return res;
}

void trianguloPascal(int n) {
    int i = 0;
    while (i <= n) {
        int j = 0;
        while (j <= i) {
            printf("%d ", combinatoria(i, j));
            j++;
        }
        printf("\n");
        i++;
```

```
    }
}
```

```
// ========================================================
// 10. Patrón numérico tipo triángulo
// ========================================================
void patronNumerico(int n) {
    int i = 1;
    while (i <= n) {
        int j = 1;
        while (j <= i) {
            printf("%d ", i);
            j++;
        }
        printf("\n");
        i++;
    }
}
```

```
// ========================================================
// 11. Cuadrado de asteriscos con while
// ========================================================
void imprimirCuadrado(int lado) {
    int fila = 0;
    while (fila < lado) {
        int col = 0;
        while (col < lado) {
            printf("* ");
            col++;
        }
        printf("\n");
        fila++;
    }
}
```



```
// ========================================================
// 12. Lista doble ligada - estructura y recorrido
// ========================================================
typedef struct Nodo {
    int valor;
    struct Nodo* anterior;
    struct Nodo* siguiente;
} Nodo;

void imprimirNormal(Nodo* cabeza) {
    while (cabeza != NULL) {
        printf("%d ", cabeza->valor);
        cabeza = cabeza->siguiente;
    }
    printf("\n");
}

void imprimirInverso(Nodo* cabeza) {
    if (cabeza == NULL) return;
    while (cabeza->siguiente != NULL) {
        cabeza = cabeza->siguiente;
    }
    while (cabeza != NULL) {
        printf("%d ", cabeza->valor);
        cabeza = cabeza->anterior;
    }
    printf("\n");
}

Nodo* obtenerMitad(Nodo* cabeza) {
    Nodo* lento = cabeza;
    Nodo* rapido = cabeza;
    while (rapido != NULL && rapido->siguiente != NULL) {
        lento = lento->siguiente;
        rapido = rapido->siguiente->siguiente;
```



```
    }
    return lento;
}
```

```
// ========================================================
// 13. Arreglos: contar positivos y encontrar primer par
// ========================================================
int contarPositivos(int arreglo[], int tam) {
    int i = 0, contador = 0;
    while (i < tam) {
        if (arreglo[i] > 0) {
            contador++;
        }
        i++;
    }
    return contador;
}

int primerPar(int arreglo[], int tam) {
    int i = 0;
    while (i < tam) {
        if (arreglo[i] % 2 == 0) {
            return arreglo[i];
        }
        i++;
    }
    return -1;
}
```

```
// ========================================================
// 14. Extras: contar ceros en un número
// ========================================================
int contarCeros(int n) {
    int ceros = 0;
```



```
    if (n == 0) return 1;
    while (n > 0) {
        if (n % 10 == 0)
            ceros++;
        n /= 10;
    }
    return ceros;
}
```

```
// ========================================================
// 15. Verificar si un número es palíndromo
// ========================================================
int esPalindromo(int n) {
    return n == invertirNumero(n);
    printf("%d es palindromo", n);
}
```

```
// ========================================================
// 16. Calcular suma y promedio de calificaciones
// ========================================================
void sumaYPromedio(int calificaciones[], int tam) {
    int i = 0, suma = 0;
    while (i < tam) {
        suma += calificaciones[i];
        i++;
    }
    float promedio = (float)suma / tam;
    printf("Suma: %d, Promedio: %.2f\n", suma, promedio);
}
```

```
// ========================================================
// 17. Mostrar tabla de multiplicar
// ========================================================
void tablaMultiplicar(int n) {
```

```
    int i = 1;
    while (i <= 10) {
        printf("%d x %d = %d\n", n, i, n * i);
        i++;
    }
}
```