

Ficha de Trabalho
<b>Curso:</b> Programação em C/C++ - fundamentos
<b>UFCD/Módulo/Temática:</b> UFCD 0809_2/N- Programação em C/C++
<b>Ação:</b> 0809_2/N
<b>Formador/a:</b> Hugo Dias
<b>Data:</b> 13/02/2025
<b>Nome do Formando/a:</b>

## Estruturas iterativas

1. As estruturas iterativas *for* são utilizadas para repetir o mesmo bloco de código, *n* vezes, e o código abaixo é um exemplo desse tipo de repetição:

```
#include <stdio.h>
int main() {
    int i;
    for(i=1 ;i<=5; i++)
        printf("Programacao em C\n");
    return 0;
}
```

```
#include <stdio.h>
int main() {
    printf("Programacao em C\n");
    printf("Programacao em C\n");
    printf("Programacao em C\n");
    printf("Programacao em C\n");
    printf("Programacao em C\n");
    return 0;
}
```

2. Desenvolver um código em C, que utiliza a estrutura iterativa *for* para escrever os números inteiros de 1 a 100.



3. Implementar um programa em C, que utiliza a estrutura iterativa *for* para escrever os números inteiros de 1 a 100, e indicar os que são pares e os que são ímpares.
4. Implementar um programa em C que solicita um valor ao utilizador, e escreve a soma dos números inteiros até esse valor (através da estrutura iterativa *while*).

$$n=5 \\ 1+2+3+4+5$$

$$n=8 \\ 1+2+3+4+5+6+7+8$$

5. As estruturas iterativas *while* permitem a repetição do mesmo bloco de código até que uma condição se verifique. No exemplo abaixo são escritos os anos bissextos desde 2000 até ao ano lido do teclado.

```
#include <stdio.h>
int main() {
    int ano,i;

    printf("Introduza o ano limite:");
    scanf("%d", &ano);

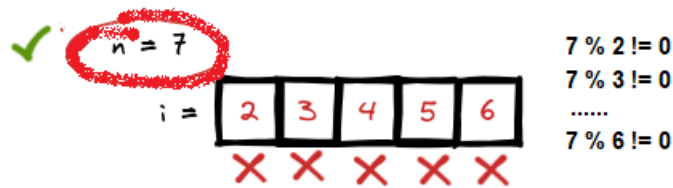
    i=2000;
    while( i<=ano ) {
        if( i%400==0 || i%4==0 && i%100!=0)
            printf("%d ",i);

        i=i+1;
    }
    return 0;
}
```

Alterar este código de forma que:

- Só apresente os primeiros 5 anos bissextos entre 2000 e o valor lido do teclado
- Calcular a diferença, em anos, entre o primeiro bissexto e o último escrito no terminal

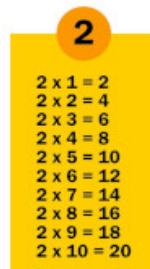
6. Implementar um programa em C que solicita um valor ao utilizador, e verifica se esse número é primo.



7. Através de uma estrutura iterativa *do...while* implementar o código que lê e soma todos os valores introduzidos pelo utilizador, até que seja introduzido o valor "0", assim que seja introduzido este valor, o programa deve terminar.



8. Implementar um programa em C que recebe um valor inteiro, e escreve no terminal a sua tabuada.



9. Implementar um programa que para um dado valor, o escreve a seguinte figura:

```

Enter a positive number n: 6
1 2 3 4 5 6
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1

```

10. O programa que se segue escreve no terminal todos os múltiplos de 5 entre dois valores definidos pelo utilizador.

```
#include <stdio.h>
int main() {
    int inicio, fim, i;

    printf("Introduza o valor inicial:");
    scanf("%d", &inicio);

    printf("Introduza o valor final:");
    scanf("%d", &fim);

    for( i=inicio ; i<=fim; i++){
        if( i%5 == 0)
            printf("%d", i);
    }
    return 0;
}
```

Alterar este código de forma que:

- Escreve no terminal todos os valores exceto os múltiplos de 5
- Assim que for detectado o primeiro múltiplo de 5 o programa termina

11. O operador ternário é uma forma simplificada de utilização do *if..else*, tal como mostra o exemplo abaixo:

```
#include <stdio.h>
int main() {
    int valor;

    printf("Introduza um valor:");
    scanf("%d", &valor);

    (valor%2==0) ? printf("%d par\n", valor) : printf("%d impar\n", valor);

    return 0;
}
```

Alterar este código para que seja possível:

- Escrever no terminal os valores pares e impares de 1 até *valor*