

Problema F

Considere as seguintes classes de problemas relativas à complexidade assintótica:

$f(n) = O(1)$	ou complexidade constante
$f(n) = O(\log n)$	ou complexidade logarítmica
$f(n) = O(n)$	ou complexidade linear
$f(n) = O(n \log n)$	ou complexidade log linear
$f(n) = O(n^2)$	ou complexidade quadrática
$f(n) = O(n^3)$	ou complexidade cúbica
$f(n) = O(2^n)$	ou complexidade exponencial
$f(n) = O(n!)$	ou complexidade fatorial

Usando as classes de problemas, classifique o seguinte código em função da entrada.

```
#include <stdio.h>
#include <stdlib.h>

int buscaElemento(int *vetor, int n, int elemento){
    int inicio = 0, fim = n - 1;

    while(inicio <= fim){
        int meio = (inicio + fim) / 2;

        if(vetor[meio] == elemento){
            // Elemento foi encontrado. Retorna sua posição
            return meio;
        } else if (vetor[meio] > elemento){
            fim = meio - 1;
        } else {
            inicio = meio + 1;
        }
    }

    // Não encontrou o elemento no vetor
    return -1;
}

int main(){
    int n;
```

```

scanf("%d", &n);

int *vetor = (int*) malloc (sizeof(int) * n);

for(int i = 0; i < n; ++i){
    vetor[i] = i;
}

int elemento;
scanf("%d", &elemento);

int resposta = buscaElemento(vetor, n, elemento);

if(resposta == -1){
    printf("Elemento nao foi encontrado.\n");
} else {
    printf("Elemento encontrado na posicao %d!\n", resposta);
}

return 0;
}

```

Você deve completar o seguinte código para imprimir a classe de complexidade assintótica que mais se ajusta ao código acima.

```

int main(){
    printf("____\n");
    return 0;
}

```

Você tem apenas uma única tentativa!

Entrada

Este problema não tem entrada.

Saída

A classe de complexidade do código apresentado

Exemplo de entrada	Exemplo de saída
	O(n!)

