

Problema A

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

Considere o seguinte código:

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

int encontrarMaior(int vetor[], int tamanho) {
    int maior = vetor[0];

    for (int i = 1; i < tamanho; i++) {
        if (vetor[i] > maior) {
            maior = vetor[i];
        }
    }

    return maior;
}

int main() {
    int n;

    scanf("%d", &n);
    int *vetor = (int*)malloc(n * sizeof(int));

    for (int i = 0; i < n; i++) {
        scanf("%d", &vetor[i]);
    }
}
```

```
    int maiorElemento = encontrarMaior(vetor, n);  
    printf("O maior elemento do vetor é: %d\n", maiorElemento);  
    free(vetor);  
    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 que mais se ajusta ao código apresentado.

Exemplo de entrada	Exemplo de saída
	$O(n!)$