



PRÁTICA 2: Análise Experimental

Implemente os Algoritmos *prefixMedia1()* e *prefixMedia2()* apresentados a seguir utilizando funções (uma para cada algoritmo). Pede-se:

- Obtenha os tempos de execução para vários valores distintos (pelo menos 5) de n para cada uma das duas funções.
- Gere um gráfico de $n \times \text{tempo de execução}$ para cada um dos algoritmos.

Algoritmo *prefixMedia1*(X, n):

Entradas: inteiro positivo n e vetor X com n números

Saída: vetor A com n números tal que $A[i]$ é a média dos elementos $X[0], \dots, X[i]$

```
para  $i \leftarrow 0$  até  $n-1$  faça
     $a \leftarrow 0$ 
    para  $j \leftarrow 0$  até  $i$  faça
         $a \leftarrow a + X[j]$ 
    fim para
     $A[i] \leftarrow a / (i+1)$ 
fim para
retorne  $A$ 
```



Algoritmo prefixMedia2(X, n):

Entradas: inteiro positivo n e vetor X com n números

Saída: vetor A com n números tal que $A[i]$ é a média dos elementos $X[0], \dots, X[i]$

$s \leftarrow 0$

para $i \leftarrow 0$ **até** $n-1$ **faça**

$s \leftarrow s + X[i]$

$A[i] \leftarrow s / (i+1)$

fim para

retorne A

DICA 1: Para obtenção dos tempos experimentais, utilize os comandos `clock()` e `difftime(tempo1,tempo2)` da biblioteca `time.h`.

Para obter o tempo em segundos deve-se usar `calcular clock()/CLOCKS_PER_SEC`, sendo `clock()` o tempo do processador em *clocks* e `CLOCKS_PER_SEC` uma constante que fornece o número de *clocks* em um segundo. O tempo em *clocks* é de um tipo de dados especial chamado `clock_t`.

O intervalo entre dois tempos (em *clocks*) pode ser calculado por `difftime(tempo1,tempo2)`. O código a seguir mostra o exemplo para calcular o tempo de um pedaço do código.



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

main(){
    clock_t tempo1, tempo2;
    double tempo_total;
    ...
    tempo1=clock();
    // inicio da parte do programa em que quer calcular o tempo
    ...
    // fim da parte do programa em que quer calcular o tempo
    tempo2=clock();
    tempo_total=difftime(tempo2,tempo1)/CLOCKS_PER_SEC;
    printf("Tempo=%g \n", tempo_total);
    ...
}
```

DICA 2: Utilize alocação dinâmica. Peça o tamanho n e então aloque dinamicamente os vetores **X** e **A** com n elementos cada na função `main()`.

DICA 3: Utilize faixas de n diferentes para os dois algoritmos.

DICA 4: Utilize um vetor gerado automaticamente (exemplo: vetor de uns). Teste se a saída está correta, mas depois não precisa mais mostrar as saídas, apenas os tempos.

DICA 5: Utilize o programa desenvolvido para coletar os tempos para cada n . Gere o gráfico utilizando algum aplicativo (por exemplo, Excel) ou à mão.