**Teoria da Computação**

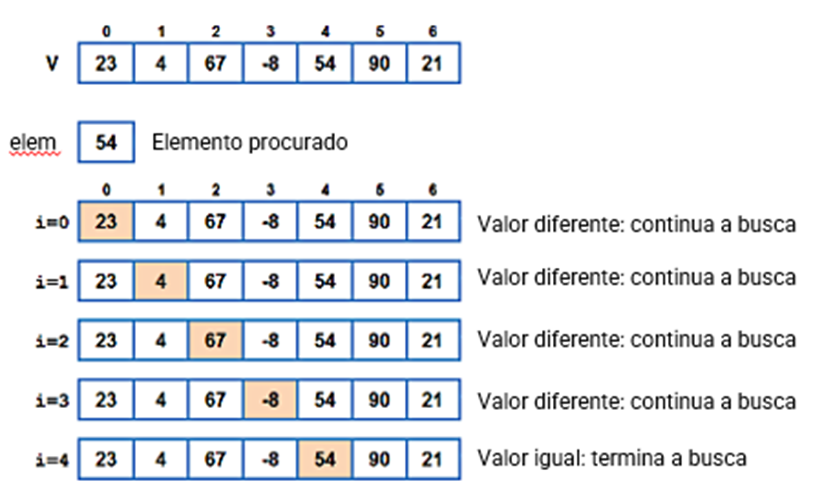
* Limites da Computação:

**O crescimento da função é exponencial e extremamente trabalhoso mesmo para um computador, daí a importância da análise de complexidade de algoritmos: estudar meios para se conhecer melhor problemas e os limites de suas soluções.** Na análise de algoritmos, não interessa a velocidade do computador mas sim o número de passos computacionais envolvidos. Soluções computacionais (algoritmos) que operam em velocidade logarítmica, linear ou polinomial são incomparavelmente menos custosos computacionalmente do que algoritmos que operam de forma exponencial.

* Buscas e Ordenações

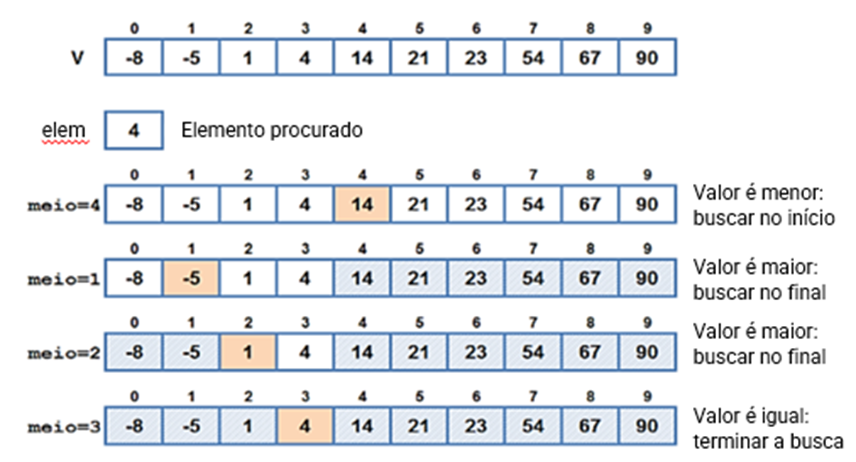
Se você pensar bem, quase tudo que fazemos em termos de computação é a busca de elementos em uma lista ou ordenação de elementos em uma lista. Métodos de Busca e Ordenação são a essência de muitos problemas e soluções em Computação!

* Busca sequencial ou linear: A busca sequencial ou linear é muito importante em estudos comparativos de análise de algoritmos. Dada uma lista de elementos, procura-se saber se um determinado elemento está na lista ou não.



Percebe-se que a busca sequencial é um algoritmo pouco eficiente para encontrar um item em uma lista grande de elementos. No pior caso, todos os elementos devem ser visitados apenas para se descobrir que o elemento não estava na lista.

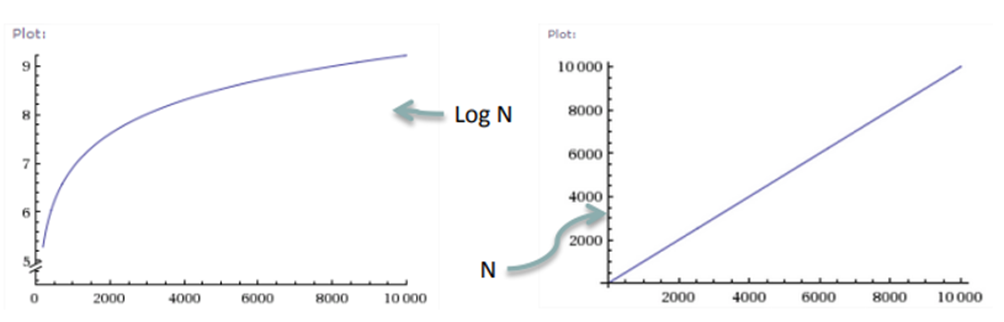
* **Busca binária:**



Neste caso, a lista está previamente ordenada.  O método faz sucessivas divisões na lista comparando com o valor maior da primeira lista e descarta metade da lista até encontrar o elemento que está procurando.

*Comparando busca sequencial com busca binária*

* Se a lista não está ordenada, a busca binária não se aplica.
* A busca sequencial é muito ineficiente para conjuntos grandes de valores.



Costuma-se dizer que a busca binária tem complexidade n e que a busca binária tem complexidade log n, fazendo muito menos comparações a partir de um certo “n” de entrada. O gráfico da Busca binária cresce mais devagar, fazendo deste método mais eficiente do que a busca sequencial.

* Máquinas de Turing: É um dispositivo imaginário que formou a estrutura para fundamentar a ciência da computação moderna. Em 1936 foi formalizado o termo algoritmo: um conjunto finito de instruções simples e precisas, que são descritas com um número finito de símbolos. “Qualquer processo aceito por nós homens como um algoritmo é precisamente o que uma máquina de Turing pode fazer”
* Autômatos finitos: Portas automáticas e elevadores. Um estado pode representar em qual estado o elevador está e as entradas podem ser os sinais recebidos dos botões. Tal computador precisaria de poucos bits para guardar essa informação. Dispositivos desse tipo requerem que se utilize a metodologia e a terminologia de autômatos finitos. Modelos para computadores quando existe pouca disponibilidade de memória. Esses computadores estão no coração de vários dispositivos eletromecânicos (forno de micro-ondas, máquinas de lavar, portas automáticas, elevadores). Lida com definições e propriedades de modelos matemáticos da computação. Autômatos finitos são usados em processamento de textos, compiladores e projetos de hardware. O modelo denominado gramática livre de contexto é utilizado em linguagens de programação e inteligência artificial.

É uma lista que contém cinco objetos:

* Conjunto de estado
* Alfabeto de entrada
* Função de transição
* Estado inicial
* Estados de aceitação