DIATINF – Diretoria Acadêmica de Gestão e Tecnologia da Informação
TADS – Tecnologia em Análise e Desenvolvimento de Sistemas
Algoritmos – 2024.1

Exercícios – Força bruta

1. Problema **Par de soma s**

   Descreva um algoritmo de força bruta para determinar se existe um par de números em um *array* cuja soma seja S. Escreva um breve texto explicando seu algoritmo, com exemplo. Determine o desempenho usando a notação *big-Oh*.

   Exemplo: Para $S = 15$, no *array* { 1 4 9 13 18 20 32 45 } não existe um par de números no *array* cuja soma seja 15. Já no *array* { 10 20 9 1 63 6 5 12 } existem dois pares: $9 + 6 = 15$ e $10 + 5 = 15$.

1.1 *Resolução*

O Problema foi solucionados de duas maneiras distintas, o primeiro algoritmo foi feito da seguinte forma:
Na primeira parte do código a declaração da função é feita com os parâmetros, respectivamente; ponteiro que aponta para o primeiro índice da lista(***lista[]***); quantidade de elementos da lista(***size***); valor que vai ser comparado com a soma dos elementos da lista(***value***).
Em seguida é verificado se a lista está vazia, caso esteja, o programa retorna falso.
O primeiro laço percorre toda a lista, esse laço vai ser executado ***n*** vezes.
o segundo laço que está indentado dentro do primeiro vai percorrer toda a lista comparando a soma do primeiro elemento com os demais, esse laço também será executado ***n*** vezes
O condicional que vem em seguida compara o valor passado como parâmetro(***value***), com os elementos somados, um por um.

1.2 *código*

```
#include <iostream>
bool soma_pares(int lista[], int size, int value){
  if (size == 0){
    return false;
  }
  else {
    int i, j;
    for (i = 0; i<size; i++){
        for (j = i+1; j<size; j++){
          if (lista[i]+lista[j] == value)
              return true;
    }
  }
  }
  return false;
}
```

1.3 Complexidade

O primeiro laço repete ***n*** vezes, sendo ***n*** o tamanho da lista, o segundo laço, da mesma maneira repete ***n*** vezes, o ***Big Oh*** desse algoritmo é: ***n.n = O(n²)***