Documentação da Função de Verificação de Números Primos

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

A função retorna um Booleano true caso o número passado seja primo e um false caso não seja.

* Análise 1: Se o número passado for 0, 1 ou um número negativo, ele não será considerado na função.
* Análise 2: Se o número passado for 2 ou 3, ele será considerado como primo.
* Análise 3: A primeira análise utilizando um cálculo, essa parte verificará se é um número par ou se é divisível por 3, eliminando assim uma boa parte dos números primos dentre o intervalo dos números passados.
* Análise 4: Após verificar que o número não é par ou divisível por 3, fazemos então uma estrutura de repetição que analisará até a raiz quadrada do número passado, já que é o maior número possível a dividir sem sobra.   
  No primeiro loop analisamos se o número passado é divisível por 5 ou 5 + 2, que seria 7, caso não seja divisível, continuamos o loop, sendo o próximo valor 5 + 6, e 5 + 6 + 2, ou seja, 11.

Texto

Descrição gerada automaticamente com confiança média

Essa tática serve para otimizar os condicionamentos, já que analisamos anteriormente se o número é divisível por 2, 3 e 4 (pois todo número divisível por 4 é um número par), então começamos com a divisão por 5 e então com os números ímpares e avanço de 6 em 6.

A função tem o uso computacional de 2 análises no melhor caso (seja par ou divisível por 3) e no seu pior caso, √n.