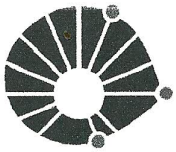
ANDRÉ VINÍCIUS SPINA

NÚMEROS PRIMOS E CRIPTOGRAFIA

CAMPINAS
2014



UNIVERSIDADE ESTADUAL DE CAMPINAS

Instituto de Matemática, Estatística
e Computação Científica

ANDRÉ VINÍCIUS SPINA

NÚMEROS PRIMOS E CRIPTOGRAFIA

Dissertação apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em matemática.

Orientador: RICARDO MIRANDA MARTINS

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA PELO ALUNO ANDRÉ VINÍCIUS SPINA, E ORIENTADA PELO PROF. DR. RICARDO MIRANDA MARTINS.

Assinatura do Orientador

CAMPINAS
2014

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Maria Fabiana Bezerra Muller - CRB 8/6162

Sp46n

Spina, André Vinícius, 1986-
Números primos e criptografia / André Vinícius Spina. – Campinas, SP : [s.n.], 2014.

Orientador: Ricardo Miranda Martins.
Dissertação (mestrado profissional) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Números primos. 2. Criptografia - Matemática. I. Martins, Ricardo Miranda,1983-. II. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Prime numbers and encryption
Palavras-chave em inglês:
Primes numbers
Cryptography - Mathematics
Área de concentração: Matemática em Rede Nacional
Titulação: Mestre em Matemática em Rede Nacional
Banca examinadora:
Ricardo Miranda Martins [Orientador]
Ary Orozimbo Chiacchio
Angelo Calil Bianchi
Data de defesa: 31-03-2014
Programa de Pós-Graduação: Matemática em Rede Nacional

Dissertação de Mestrado Profissional defendida em 31 de março de 2014 e aprovada Pela Banca Examinadora composta pelos Profs. Drs.

Prof.(a). Dr(a). RICARDO MIRANDA MARTINS

Prof.(a). Dr(a). ARY OROZIMBO CHIACCHIO

Prof.(a). Dr(a). ANGELO CALIL BIANCHI

Abstract

The paper presents a Number Theory introduction, trough a RSA and Diffie-Hellman cryptographic methods approach, where one can observe situations where they are effective. The mathematical theory introduced in this paper encompass prime numbers, Modular arithmetic,Primality test, groups and other Number Theory related branches.

Keywords: Primes numbers, Cryptography-Mathematics.

Resumo

A pesquisa apresentará uma introdução a Teoria dos Números através de uma abordagem sobre os métodos criptográficos RSA e Diffie-Hellman, onde pode-se constatar situações onde eles são eficientes. A teoria matemática presente nesse trabalho envolve conhecimentos em números primos, aritmética modular, testes de primalidade, grupos e outras questões envolvendo teoria dos números.

Palavras-chave: Números primos, Criptografia-Matemática.

Sumário

Dedico esta dissertação a toda minha família, amigos e principalmente a meus alunos, que ao estudá-la consigam sentir um pouco do prazer que a matemática é capaz de proporcionar.

Agradecimentos

A Deus por ter me dado forças que me permitiram chegar até aqui, por ter me apontado a direção a qual me levou a conquistar esse meu sonho. Obrigado Pai, o Senhor é meu caminho, minha verdade e minha vida. A minha família por toda a educação, carinho, confiança e paciência que me foram concedidas e que serviram de alicerce para o meu comprometimento e paixão com os estudos. Muito obrigado por terem me permitido sonhar. Em especial a minha Mãe e meu Pai, meus primeiros professores, meu porto seguro. A todos os amigos que Deus colocou no meu caminho, os quais me levaram a ser quem sou. A todos os meus professores que me ajudaram a chegar até aqui, desde minha mãe que me ensinou minhas primeiras palavras até meu orientador, Ricardo Miranda Martin, o qual acompanhou todos os passos da elaboração dessa dissertação. Agradeço também a CAPES pelo apoio financeiro e ao IMECC UNICAMP por abrir as portas para que eu pudesse realizar o sonho de cursar mestrado.

Introdução

Todos os números inteiros $n \geq 2$ são formados como produtos de números primos, ou seja, os números primos são a matéria prima para a construção dos números. Mas, como encontrar números primos? Os números primos aparecem ao acaso no nosso conjunto numérico e descobrir uma maneira eficaz de encontrá-los tem sido a tarefa de alguns matemáticos por mais de dois mil anos, tarefa que, apesar de alguns avanços, continua sem solução. Assim sendo, dar sentido a ordem dos números primos e descobrir uma maneira eficaz de encontrá-los é um problema e, também, um desafio.

Nos últimos anos essa questão ganhou uma nova forma de incentivo, devido a um método criptográfico chamado RSA, no qual todo o sistema financeiro atual esta apoiado e o qual se utiliza de números primos para tornar essas operações seguras. Segurança essa, que se baseia na dificuldade de encontrar os fatores primos de um dado número. Se esse problema tivesse solução, se fosse possível fatorar um número rapidamente, a segurança do sistema financeiro atual deixaria de existir, pois tal método criptográfico seria inutilizado, bem como todas as suas atuais aplicações.

Sabemos que grande parte das transações que envolvem dinheiro, tais como compras com cartão de crédito e até mesmo saques em caixas eletrônicos são feitas via internet, podendo assim, serem interceptadas sem que ninguém perceba. Para que essas transações tornem-se seguras, mesmo sendo interceptadas por pessoas mal intencionadas, elas devem ser protegidas, de modo que somente o destinatário (banco ou empresa do cartão de crédito) consiga ler e utilizar as informações ali contidas e para isso usamos a criptografia. Criptografia é o estudo de métodos para codificar mensagens de maneira que só o destinatário possa ler e entender o conteúdo que ela contém, de modo que para qualquer outra pessoa que consiga ter acesso a tal mensagem essa torne-se ilegível, sendo então incapaz de ter acesso às suas informações. Os códigos utilizados na criptografia para codificar uma mensagem e também para decodificá-la são chamadas de chaves.

Embora o conceito utilizado seja bem simples, encontrar uma maneira de codificar tais mensagens com a segurança necessária, não foi tarefa fácil. Isso é devido à maneira como estas transações bancárias ocorrem, por exemplo, quando é realizada uma compra com cartão de crédito via *internet*, o comprador precisa informar a loja sobre os dados de seu cartão de crédito, porém, se esses dados forem revelados a outra pessoa, essa pessoa conseguirá fazer compras com tal cartão e, assim, para que isso não ocorra, as informações do cartão de crédito precisam ser codificadas pelo computador antes de serem enviadas.

O problema é que a codificação feita pelo computador do comprador não pode acontecer de maneira aleatória, pois a loja que receberá os dados do cartão de crédito precisa saber decodificar. O que ocorre, é que a loja informa como deve ser feito a codificação (a chave de codificação), tornando possível para o computador do comprador codificar os dados do cartão de crédito de maneira que a loja consiga decodifica-los e, assim, utilizar tais dados. Porém, como o processo de

codificação foi enviado via *internet* pela loja até o comprador, ele fica vulnerável à interceptações de terceiros, os quais uma vez sabendo o código (chave) utilizado para codificar os dados do cartão de crédito, podem decodificá-los desfazendo a codificação realizada. O que, assim sendo, qualquer que seja o código que seu computador utilize para criptografar, ele sempre estará sujeito a cair nas mãos de terceiros, tornando assim essa operação nada segura.

Até pouco tempo atrás, isso poderia ser verdade, porém nos dias atuais existem métodos criptográficos que, mesmo tendo acesso a chave utilizada para codificar a mensagem, decodificá-la não é tão simples assim. Decodificar a mensagem simplesmente desfazendo o processo de codificação (utilizando um processo inverso ao utilizado para codificar) torna-se tão trabalhoso e demorado de maneira que é impossível colocá-lo em prática.

Os métodos criptograficos vêm sendo aprimorados desde a época do Imperador Julio César, que utilizava mensagens codificadas para se comunicar com seus soldados durante época de guerra. Por volta do ano de 1960 foi inventada a criptografia que utiliza chaves públicas, uma criptografia onde mesmo se a chave transmitida for interceptada não será suficiente para a decodificação da mensagem. A principal diferença entre a utilização da chave pública de uma chave convencional, está na decodificação da mensagem, pois para se decodificar uma mensagem que se utiliza de uma chave convencional bastava realizar um processo inverso ao realizado para codificar tal mensagem. Já na utilização da chave pública, realizar o processo inverso se torna tão trabalhoso e demorado que vem a ser impossível decodificar a mensagem através deste processo. O porque desse processo se tornar tão trabalhoso tem a mesma essência do problema em descobrir se um número é primo.

A chave pública utilizada na criptografia RSA é constituída de um número composto suficientemente grande, enquanto que a chave que ficará privada (e será utilizada para decodificar) consiste dos fatores primos de tal número composto. Como dito anteriormente, é um processo que não conhecemos nenhum método eficaz o suficiente, tornando segura a Criptografia RSA. Desfazer o processo, ou seja, fatorar tal número ou encontrar os primos que o compõe, é um processo que ainda não conhecemos nenhum método eficaz suficiente, e esse fato é o que torna tal chave segura.

Assim sendo, esse trabalho se pautará no estudo de métodos para se detectar números primos e de uma aplicação envolvendo números primos, a Criptografia de Chave Pública, a qual se utiliza de propriedades especiais dos números primos. No capítulo 1 será abordado um estudo a partir da definição dos números primos, onde, com a ajuda de alguns resultados matemáticos importantes, avançaremos o conhecimento sobre os números primos, a ponto de traçar testes de primalidades modernos que serão analisados no Capítulo 2 no qual levantaremos as vantagens e desvantagens de cada um. Ao final do Capítulo 2 poderá se constatar o grande problema que todos os testes de primalidade sofrem, e que será a essência dos métodos criptográficos, que serão apresentados no Capítulo 3. No Capítulo 4 será apresentada uma sequência didática idealizada que pode servir de molde para uma utilização em sala de aula.

Capítulo 1

PRELIMINARES

Será tratado nesse capítulo o estudo de alguns teoremas, ideias e conceitos sobre números inteiros os quais são imprescindíveis para a análise e compreensão dos números primos, bem como a dificuldade em se detectar a primalidade de um número, sendo esse o grande trunfo do método criptográfico RSA.

O resultado mais relevante até o momento para se detectar a primalidade de um número é o chamado teste de Miller, o qual utiliza a aritmética modular para analisar e encontrar números primos. Esse teste de primalidade é o que se tem de mais moderno para se detectar se um número é primo ou não, mas mesmo assim ainda só é viável sua utilização até certo ponto, pois dependendo do tamanho do número a ser examinado, tal teste pode demorar tanto tempo para encontrar uma resposta, que utilizá-lo se torna inviável.

Afinal, o que é um número primo?

Definição 1. Um número inteiro positivo é chamado de primo quando possui somente dois divisores: 1 e ele próprio.

Definição 2. Um número inteiro positivo é chamado de composto quando possui 3 ou mais divisores.

Exemplo 1. 8 não é primo, pois 4 divide 8
5 é primo, pois 2 não divide 5, 3 não divide 5 e 4 não divide 5.

Quando um número b divide um número n, escrevemos: $b \mid n$, exemplo: $4 \mid 8$ (4 divide 8). O que equivale a dizer que a divisão de n por b deixa resto zero ou ainda que: $n = b \cdot q$, exemplo: $8 = 4 \cdot 2$.

Se um número b **não** divide um número n, escrevemos: $b \nmid n$. O que equivale a dizer que a divisão de n por b deixa resto diferente de zero, ou ainda: $n = b \cdot q + r$, com $0 < r < b$, exemplo: $11 = 4 \cdot 2 + 3$.

Neste ponto é importante utilizarmos, como recurso, o Algoritmo da Divisão, o qual ajudará a identificarmos mais rapidamente os possíveis resultados. Tal Algoritmo nos diz que todo $n \in \mathbb{Z}$ pode ser escrito como:

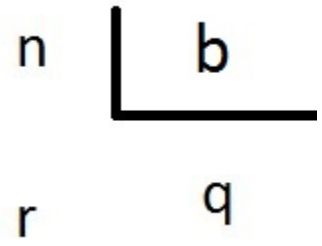


Figura 1.1: Algoritmo da divisão

$n = b \cdot q + r$, onde $0 \leq r < b$, sendo $b \in \mathbb{N}$.

Assim sendo, temos os possíveis casos:

- Se $r = 0$, temos que b divide n e podemos escrever: $b \mid n$.
- Se $r \neq 0$, temos que b não divide n e escrevemos $b \nmid n$.

Como saber se um número é primo?

Quanto maior for o número a ser verificado, maior a quantidade de candidatos a divisores este número terá.

Exemplo 2. 5021 é primo?

Para verificar se ele é primo olhando para a definição de números primos, temos que 5021 é primo se for divisível somente por 1e por ele mesmo, logo teria que se verificar se 5021 não é divisível por 2, por 3, por 4, por 5, por 6,..., por 5020, o que dá um total de 5018 candidatos a verificar.

1.1 Teorema Fundamental da Aritmética

Todos os números não primos podem ser formados como produtos de primos e de maneira única, é o que diz o Teorema Fundamental da Aritmética:

Teorema 1. (Teorema Fundamental da Aritmética) Um número natural maior que 1, ou é primo, ou se escreve de maneira única como um produto de primos, a menos de ordenação.

Demonstração: Vamos primeiro demonstrar que um número natural n se escreve como um produto de primos:

Para isso iremos utilizar o príncipio de indução finita. Observe que $n = 2$, o teorema é valido, pois 2 é primo. Suponha agora que o teorema é valido para todo número natural maior que 2 e menor que n e, então, provaremos que também é válido para n. Observe que dessa forma n não é primo e, portanto, existe $2 \leq n_1 < n$ que divide n, e então escrevemos $n = n_1 \cdot k$ sendo $2 \leq k < n$. Dessa forma por hipótese de indução, temos que n_1 e k podem ser escrito como produto de primos: $n_1 = p_1 \cdots p_n$ e $k = q_1 \cdots q_n$, e então podemos escrever $n = n_1 \cdot k = p_1 \cdots p_n \cdot q_1 \cdots q_n$.

Falta agora demonstrar a unicidade dessa escrita.

Suponha por absurdo que n possa ser escrito de duas maneiras diferentes ao menos da ordem de seus fatores, ou seja:
$n = p_1 \cdot p_2 \cdot ... \cdot p_n$ e também:
$n = q_1 \cdot q_2 \cdot ... \cdot q_s$, com p_i e q_j números primos sendo $i \in \{1, 2, ..., n\}$ e $j \in \{1, 2, ..., s\}$.

Observe que $p_1|n$ pois $n = p_1 \cdot p_2 \cdot ... \cdot p_n$, e dessa forma temos também que:
$p_1|q_1 \cdot q_2 \cdot ... \cdot q_s$ que também podemos escrever:
$q_1 \cdot q_2 .... \cdot q_s = p_1 \cdot a$

Como p_1 é um número primo e q_j também são primos, conclui-se que $p_1 = q_j$ para algum j. Podemos reordenar os q_j de forma que $p_1 = q_1$.

Desta forma podemos simplificar a igualdade $q_1 \cdot q_2 \cdot ... \cdot q_s = p_1 \cdot a$ obtendo: $q_2 \cdot ... \cdot q_s = a$

Lembrando que $n = p_1 \cdot p_2 \cdot ... \cdot p_n$, temos então que $a = p_2 \cdot ... \cdot p_n$, e portanto:
$p_2|a$, e portanto, como $q_2 \cdot ... \cdot q_s = a$, temos que:
$p_2|q_2 \cdot ... \cdot q_s$ que é o mesmo de:
$q_2 \cdot ... \cdot q_s = p_2 \cdot b$ onde $b = p_3 \cdot ... \cdot p_n$

Como p_2 é um número primo e q_j também são primos, conclui-se que $p_2 = q_j$ para algum j. Podemos reordenar os q_j de forma que $p_2 = q_2$.

Podendo assim simplificar a igualdade $q_2 \cdot ... \cdot q_s = b$ obtendo: $q_3 \cdot ... \cdot q_s = p_3 \cdot ... \cdot p_n$

Prosseguindo desta forma, podemos concluir que $p_1 = q_1, p_2 = q_2, ..., p_n = q_s$, provando assim, que n se escreve como produtos de primos de maneira única.

O Teorema Fundamental da Aritmética é o resultado que nos permite escrever qualquer número como um produto de primos. No ensino Fundamental o utilizamos em muitas ocasiões, simplificar frações, calcular o m.m.c., m.d.c. entre outras aplicações.

Exemplo 3. Escrevendo 10 e 18 como um produto de primos, temos:
$10 = 2 \cdot 5$
$18 = 3^2 \cdot 2$

O Teorema Fundamental da Aritmética garante que um número composto terá sempre os mesmos números primos em sua composição. Desta maneira, utilizando o algoritmo da fatoração podemos encontrar os fatores primos de um número e escrevê-lo como um produto de primos.

Podemos melhorar um pouco esse método se notarmos o seguinte:

Voltando ao exemplo 2, ao verificar que o número 5021 não é divisível por 2, podemos afirmar que este número (pelo Teorema Fundamental da Aritmética) não contém o fator 2 em sua decomposição em fatores primos, e portanto, nenhum outro número que contenha o 2 em sua decomposição (os múltiplos de 2) pode dividir o número 5021. Nenhum número do conjunto $M(2) = \{2, 4, 6, 8, ...\}$ divide o número 5021.

Utilizando o mesmo pensamento para todos os outros primos menores que 5021, podemos concluir que basta verificar então se o número em questão, no caso 5021, é divisível pelos primos que serão menores que ele.

Se pensarmos um pouco adiante, conseguiremos deduzir que também não se faz necessário verificar todos os primos menores que o número em questão.

Exemplo 4. Para o número 73, temos que: 2 não divide 73, 3 não divide 73, 5 não divide 73 e 7 não divide 73 e como $\sqrt{73} < 9$, temos que $9 \cdot 9 > 73$, e dessa forma não existe nenhum fator primo

de 73 maior do que $\sqrt{7}3$, porque se houvesse o outro fator primo também teria que ser maior que $\sqrt{7}3$ (já que verificamos para todos os menores do que $\sqrt{7}3$,) o que gera um absurdo, uma vez que a multiplicação de dois números maiores do que 9 é sempre maior do que 73.

Dessa forma para saber se um número é primo, basta verificar se ele é divisível por algum número primo menor ou igual a raiz quadrada deste número.

No caso do 5021, basta verificar se ele é divisível por algum número primo menor ou igual a $\sqrt{5}021 \simeq 71$, se nenhuma destas verificações acontecer 5021 será primo.

Embora torne o método mais eficaz ele ainda funciona até certo ponto, já que quando se trata de números grandes, os números que precisam ser verificados acabam sendo muitos, sem contar que, faz-se necessário conhecer todos os números primos que são menores que a raiz quadrada deste.

Para seguir em frente, é fundamental utilizar uma aritmética focada nos restos de divisões, a chamada arimética modular.

Utilizando a divisão euclidiana, temos que um número $n \in \mathbb{Z}$ pode ser escrito como: $n = m \cdot q + r$ onde $m \in \mathbb{N}$ e $0 \leq r < m$.

Mas, e se o que se precisa analisar for somente o resto desta divisão sem se preocupar com o valor do quociente?

Nesse caso, poderemos escrever: $n \equiv r \mod m$, ou seja, n deixa resto r quando dividido por m, ou ainda de uma maneira mais geral:
$n \equiv a \mod m$, dizemos que n é congruo a a no módulo m, o que significa que n e a tem o mesmo resto na divisão por m.

Traçando um paralelo com a divisibilidade, podemos escrever também:
$m|n - a \Leftrightarrow n \equiv a \mod m$, pois, se n e a deixam o mesmo resto na divisão por m, temos que a diferença entre n e a é divisível por m.

Exemplo 5. $34 \equiv 10 \mod 3$, pois ambos deixam resto 1 quando divididos por 3. Utilizando divisibilidade temos então: $3|34 - 10$.

A utilização da aritmética modular facilita-rá cálculos com números grandes, em que, estamos preocupados somente com o resto que eles deixam quando divididos por algum outro número.

Como ilustração, vejamos um exemplo:

Exemplo 6. Qual o resto da divisão de 4^{75} por 17?
$4^2 \equiv -1 \mod 17$ pois $4^2 = 16$, portanto:
$4 \cdot 4^{74} \equiv 4 \cdot (4^2)^{37} \mod 17$
$4 \cdot (4^2)^{37} \equiv 4 \cdot (-1)^{37} \mod 17$
$4^{75} \equiv -4 \equiv 13 \mod 17$, ou seja, o resto da divisão de 4^{75} por 17 é 13.

1.2 Pequeno Teorema de Fermat

Um resultado muito importante e que nos auxilia no cálculo de congruências é o Pequeno Teorema de Fermat.

Teorema 2. (Pequeno Teorema de Fermat) Se p for um número primo, então p divide $a^p - a$ para todo número inteiro a :

$p|a^p - a$, ou ainda: $a^p \equiv a \mod p$.

Demonstração:

Uma maneira de se provar tal resultado, é utilizar o Princípio de Indução Finita, já que ele nos fornece um método eficaz de demonstrações em situações envolvendo números Naturais.

Com o Pequeno Teorema de Fermat faz uso dos Números Inteiros, provaremos primeiro que ele é válido para os Números Naturais e em seguida estenderems a prova para os Números Inteiros.

Queremos provar então, em um primeiro momento, que a proposição $a^p \equiv a \mod p$ é válida para os Números Naturais.

1)Utilizando $a = 1$ como base, temos:
$a^p = 1^p = 1, \forall$ primo p, logo: $1^p \equiv 1 \mod p$

2) Após ter concluído que para $a = 1$ é verdade. Suponhamos que $a^p \equiv a \mod p, \forall a \in \mathbb{N}$ esta é nossa Hipótese de Indução.

3) Em seguida utilizando nossa hipótese provaremos que:

$$(a+1)^p \equiv a + 1 \mod p$$

Expandindo $(a+1)^p$ segundo o Binômio de Newton: $\sum_{i=0}^{p} \binom{p}{i} \cdot a^{p-i} \cdot 1^i$.

Se $i = 0$, temos $\binom{p}{0} \cdot a^p \cdot 1^0 = a^p$.

Se $i = p$, temos $\binom{p}{p} \cdot a^{p-p} \cdot 1^p = 1^p$.

Logo tirando estes dois casos do somatório, temos:

$$(a+1)^p = a^p + 1^p + \sum_{i=1}^{p-1} \binom{p}{i} \cdot a^{p-i} \cdot 1^i$$

Como $\binom{p}{i} = \frac{p \cdot (p-1) \cdot \ldots \cdot (p-i+1)}{i!}$, é sempre inteiro e como p não divide $i!$, temos que$\binom{p}{i}$ possui p como um de seus fatores primos e, portanto, p divide $\binom{p}{i}$.

Em termos de congruência: $\binom{p}{i} \equiv 0 \mod p$.

Assim:

$$\sum_{i=1}^{p-1} \cdot \binom{p}{i} \cdot a^{p-i} \cdot 1^i \equiv 0 \mod p.$$

E como por Hipótese de Indução temos que $a^p \equiv a \mod p, \forall a \in \mathbb{N}$, podemos concluir que:

$$a^p + 1^p + \sum_{i=1}^{p-1} \binom{p}{i} \cdot a^{p-i} \cdot 1^i \equiv a + 1 \mod p$$

Encontrando assim que:
$(a+1)^p \equiv a^p + 1^p + \sum_{i=1}^{p-1} \binom{p}{i} \cdot a^{p-i} \cdot 1^i \equiv a^p + 1^p \equiv (a+1) \mod p$.

Desta forma provamos que : $a^p \equiv a \mod p$ é valido para todo $a \in \mathbb{N}$

Precisamos agora, estender esse resultado para os Números Inteiros ($\mathbb{Z}$), para isso falta que provemos que $a^p \equiv a \mod p$ também é valido para os números negativos.

Em um primeiro momento suponha que $p = 2$ e $a = -\alpha$, com α positivo:

$(-\alpha)^2 \equiv -\alpha \mod 2$

$\alpha^2 \equiv -\alpha \mod 2$

$\alpha^2 + \alpha \equiv 0 \mod 2$

Observe agora que, se α for par , ou seja : $\alpha = 2k$, temos: $(2k)^2 + 2k \equiv 0 \mod 2$, que é sempre verdade.

Se α for ímpar, ou seja: $\alpha = 2k + 1$, temos:

$(2k + 1)^2 + 2k + 1 \equiv 2.2k^2 + 2 \cdot 2k + 1 + 2k + 1 \equiv 0 \mod 2$, que também é sempre verdade.

Assim, concluímos que, se $p = 2$, temos que $a^p \equiv a \mod p$ é sempre verdade $\forall a \in \mathbb{Z}$. Por último falta provarmos que, se p for um número ímpar (uma vez que o único primo par existente é o 2) e a um número negativo, continua sendo válido: $a^p \equiv a \mod p$.

Suponhamos então $a = -\alpha$ e p um primo ímpar, sendo $\alpha \in \mathbb{N}$, temos:

$(-\alpha)^p = -\alpha^p$, pois p é ímpar.

Utilizemos então $\alpha^p \equiv \alpha \mod p$, resultado que já provamos para os Números Naturais utilizando o Princípio da Indução Finita, e em seguida podemos multiplicar por (-1) ambos os lados da congruência, obtendo:

$-\alpha^p \equiv -\alpha \mod p$, como p é ímpar:

$(-\alpha)^p \equiv -\alpha \mod p$, resultado então que o Teorema também é valido para os números negativos.

Como o caso $a = 0$ é trivial, concluimos assim, a demonstração de que $a^p \equiv a \mod p$ é valido para todo $a \in \mathbb{Z}$ e p primo.

Exemplo 7.

$$4^3 \equiv 4 \mod 3 \Rightarrow 4^3 \equiv 1 \mod 3$$

$$6^2 \equiv 6 \mod 2 \Rightarrow 6^2 \equiv 0 \mod 2$$

É importante observar que o Pequeno Teorema de Fermat diz respeito a congruências cujo módulo são apenas números primos, o que traz uma restrição quanto a sua utilização. Essa restrição já não existe no Teorema apresentado por Euler o qual abrange todos os números inteiros.

Teorema 3. (Teorema de Euler) Sejam $a \in \mathbb{Z}$ e $n \in \mathbb{N}$, se $(a, n) = 1$ então:

$$a^{\phi(n)} \equiv 1 \mod n,$$

sendo $\phi(n)$ a quantidade de elementos do conjunto dos possíveis restos na divisão por n que são primo com n, onde é possível calculá-la escrevendo n como um produto de primos: $n = p_1^{\alpha_1} \cdot ... \cdot p_n^{\alpha_n}$,

com $p_1, p_2, ..., p_n$ números primos distintos e $\alpha_1, \alpha_2, ..., \alpha_n$ números inteiros positivos.Desta forma:

$$\phi(n) = \phi(p_1^{\alpha_1} \cdot ... \cdot p_n^{\alpha_n}) = p_1^{\alpha_1 - 1} \cdot ... \cdot p_n^{\alpha_n - 1} \cdot (p_1 - 1) \cdot ... \cdot (p_n - 1)$$

Observe que se n for um número primo, temos: $\phi(n) = \phi(p) = p^{1-1} \cdot (p-1) = p-1$, e então:
$a^{\phi(n)} \equiv 1 \mod n \Rightarrow a^{p-1} \equiv 1 \mod p$, que nada mais é do que o Pequeno Teorema de Fermat em uma outra versão a qual estudaremos mais adiante.

Para compreender um pouco mais sobre o Pequeno Teorema de Fermat e sobre a ordem de um número n, se faz necessário estudar um pouco sobre Grupos.

1.3 Grupos

Um Grupo é um conjunto que possui uma operação bem definida além de satisfazer algumas propriedades.

Para tal operação estar bem definida ela deve associar a cada dois elementos desse grupo um terceiro elemento também desse grupo:

$$a, b \in G \Rightarrow a * b \in G.$$

Porém para tal conjunto constituir um grupo, precisa também satisfazer as seguintes propriedades:

1. Associatividade: $a * (b * c) = (a * b) * c$
2. Elemento Neutro: $a * e = e * a = a$
3. Elemento Inverso $a * a' = a' * a = e$

Um grupo muito utilizado é o dos Números Inteiros com a operação da adição:
Temos que $a, b \in \mathbb{Z} \Rightarrow a + b \in \mathbb{Z}$ e também que vale:

1. Associatividade $a + (b + c) = (a + b) + c$
2. Elemento Neutro $a + 0 = 0 + a = a$
3. Elemento Inverso $a + (-a) = (-a) + a = 0$

Caracterizando, assim, o conjunto dos Números Inteiros como um grupo para com a adição.

Entretanto, o conjunto dos Números Inteiros não forma um grupo com a operação de multiplicação,pois apesar da operação estar bem definida: $a, b \in Z \Rightarrow a \cdot b \in \mathbb{Z}$, e possuir as propriedades Associativa e Elemento Neutro, ela não possui Elemento Inverso:

1. Associatividade $a \cdot (b \cdot c) = (a \cdot b) \cdot c$;
2. Elemento Neutro $a \cdot 1 = 1 \cdot a = a$.

Porém, $\nexists a' \in \mathbb{Z}$, tal que: $a \cdot a' = a' \cdot a = 1$, e, portanto, podemos concluir que os Números Inteiros não formam um grupo para com a operação de multiplicação, pois ela não possui elemento inverso.

Ao se utilizar congruências, os grupos utilizados são os $\mathbb{Z}_n$, ou seja, Inteiros módulo n, vejamos alguns exemplos:

1) Ao se analisar $\mathbb{Z}_4$, temos como classes residuais: $\bar{0}, \bar{1}, \bar{2}, \bar{3}$ (os possíveis restos na divisão por 4). Logo, temos uma congruência do tipo: $a \equiv b \mod 4$ onde $a, be \in \mathbb{Z}$ e $\bar{a} \in \left\{\bar{0}, \bar{1}, \bar{2}, \bar{3}\right\}$.

Vejamos se $\mathbb{Z}_4$ é um grupo com a operação de adição:

Temos que $x, y \in \mathbb{Z} \Rightarrow x + y \in \mathbb{Z}$ e $x \equiv b_1 \mod n$ e $y \equiv b_2 \mod n \Rightarrow x + y \equiv b_1 + b_2 \mod n$.

Como b_1 *e* $b_2 \in \left\{\bar{0}, \bar{1}, \bar{2}, \bar{3}\right\}$, vejamos todas as possibilidades de $b_1 + b_2 \mod 4$:

+	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

Tabela 1.1: Adição em $\mathbb{Z}_4$

Verifica-se que $b_1 + b_2 \in \mathbb{Z}_4$ e que:

1. Associatividade: $a + (b + c) = (a + b) + c$
2. Elemento Neutro: $a + 0 = 0 + a = a$
3. Elemento Inverso $a + (-a) = (-a) + a = 0$

Desta forma $\mathbb{Z}_4$ é um grupo com a operação de Adição.

Vejamos agora se $\mathbb{Z}_4$ também é um grupo com a operação de Multiplicação:

Seguindo os mesmos passos, temos que $x, y \in \mathbb{Z} \Rightarrow x \cdot y \in \mathbb{Z}$ e $x \equiv b_1 \mod n$ e $y \equiv b_2 \mod n \Rightarrow x \cdot y \equiv b_1 \cdot b_2 \mod n$.

Como b_1 e $b_2 \in \left\{\bar{0}, \bar{1}, \bar{2}, \bar{3}\right\}$, vejamos todas as possibilidades de $b_1 \cdot b_2 \mod 4$.

.	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

Tabela 1.2: Multiplicação em $\mathbb{Z}_4$

Verifica-se que $b_1 \cdot b_2 \in \mathbb{Z}_4$.

Em seguida precisamos analisar se é válido:

1. Associatividade: $a \cdot (b \cdot c) \equiv (a \cdot b) \cdot c \mod 4$

2. Elemento Neutro: $a \cdot 1 \equiv 1 \cdot a \equiv a \mod 4$

3. Elemento Inverso $a \cdot b \equiv b \cdot a \equiv 1 \mod 4$

As propriedades 1 e 2 podem ser facilmente analisadas pelas propriedades da multiplicação, já a propriedade 3 precisa ser analisado com um pouco mais de cuidado. Analisando todas as classes residuais de $\mathbb{Z}_4$ temos:

Estamos analisando se $a \cdot b \equiv b \cdot a \equiv 1 \mod 4$ para todo $a \in \mathbb{Z}_4 \backslash \{0\}$

Repare que, quando $a = 2$, não existe nenhum $b \in \mathbb{Z}_4$ tal que:

$$a \cdot b \equiv b \cdot a \equiv 1 \mod 4$$

Logo, $\mathbb{Z}_4$ não é um grupo com a operação de multiplicação.

A ausência deste inverso multiplicativo muitas vezes é um problema, inclusive no estudo de números primos e de fatoração, onde é utilizada a multiplicação com frequência. Por esse motivo, precisamos utilizar um grupo para com a operação de multiplicação, e como a falta do Elemento Inverso é sempre um problema, comecemos procurando condições para que $a', a \in \mathbb{Z}_n$, tenham a propriedade:

$a \cdot a' \equiv a' \cdot a \equiv 1 \mod n$, o que só acontecerá se: $a \cdot a' + q \cdot n = 1$, com $n \in \mathbb{Z}$, observe que podemos encontrar tal resultado utilizando o algoritmo extendido de Euclides em $(a, n) = 1$, e dessa forma para que o conjunto possua Elemento Inverso precisamos encontrar um n que seja primo com a.

Observe que ao utilizar o conjunto $\mathbb{Z}_p$, sendo p um número primo, a condição $(a, p) = 1$ é respeitada sempre que $p \nmid a$, mas observe que quando $p \mid a$, temos que $a = p.q$ para algum $q \in \mathbb{Z}$, e assim sendo, $a = \bar{0}$, portanto tal conjunto possui Elemento Inverso para todos os seus elementos com excessão é claro do zero (vale lembrar que em um grupo com a operação de multiplicação não teremos Elemento Neutro e nem Elemento Inverso para o zero).

Nas congruências podemos obter um grupo para com a operação de multiplicação ao utilizar o conjunto $\mathbb{Z}_n$ desde que n seja um número primo, já que assim, desta maneira, temos também o inverso multiplicativo (uma demonstração para tal resultado pode ser encontrada em Coutinho 2.013 p.141, 142.)

Vamos analisar o conjunto $\mathbb{Z}_5$ e provar que ele constitui um grupo para com a operação de adição e também para com a operação de multiplicação:

Temos que $a, b, c \in \mathbb{Z}_5 \Rightarrow a + b \in \mathbb{Z}$ e também que vale:

1. Associatividade: $a + (b + c) \equiv (a + b) + c \mod 5$;

2. Elemento Neutro: $a + 0 \equiv 0 + a \equiv a \mod 5$;

3. Elemento Inverso $a + (-a) \equiv (-a) + a \equiv 0 \mod 5$.

Com relação a Multiplicação :

$a, b \in \mathbb{Z} \Rightarrow a \cdot b \in \mathbb{Z}$

1. Associatividade $a \cdot (b \cdot c) \equiv (a \cdot b) \cdot c \mod 5$;

2. Elemento Neutro $a \cdot 1 \equiv 1 \cdot a \equiv a \mod 5$,sendo $a \neq \bar{0}$;

3. Elemento Inverso $a \cdot b \equiv b \cdot a \equiv 1 \mod 5$sendo $a \neq \bar{0}$.

Falta atentarmos com a questão do inverso multiplicativo, observe que verificasse que para todo $a \in \mathbb{Z}_5 \backslash \{0\}$, existe seu inverso multiplicativo $b \in \mathbb{Z}_5$

$a \cdot b \equiv b \cdot a \equiv 1 \mod 5$

$1 \cdot 1 \equiv 1 \cdot 1 \equiv 1 \mod 5$

$2 \cdot 3 \equiv 3 \cdot 2 \equiv 1 \mod 5$

$4 \cdot 4 \equiv 4 \cdot 4 \equiv 1 \mod 5$

E, desta forma, $\mathbb{Z}_5$ é um grupo tanto para a adição quanto para a multiplicação.

Voltemos ao Pequeno Teorema de Fermat: ele nos diz que se p é primo, temos: $a^p \equiv a \mod p$, tomemos aqui o inverso multiplicativo de a, ou seja, $b \in \mathbb{Z}$, tal que: $a \cdot b \equiv 1 \mod p$. Podemos multiplicar ambos os lados da congruência $a^p \equiv a \mod p$ por b, obtendo:

$b \cdot a^p \equiv b \cdot a \mod p$, e, como b é inverso de a:

$b \cdot a \cdot a^{p-1} \equiv b \cdot a \mod p$

$1 \cdot a^{p-1} \equiv 1 \mod p$

$a^{p-1} \equiv 1 \mod p$.

Dessa forma, podemos entender com mais alguns detalhes o Pequeno Teorema de Fermat. Já o Teorema de Euler que é uma versão mais geral do Pequeno Teorema de Fermat contém o número $\phi(n)$, que chamamos de fução phi de Euler, esse número é a quantidade de elementos de um sistema reduzido de resíduos módulo n, as vezes também chamado de quantidade de classes reduzidas de resíduos, observemos a congruência: $x \equiv y \mod 14$.

A congruência acima apresenta o sistema completo de resíduos: $0, 1, 2, 3, ..., 13$, que são os possíveis restos de uma divisão por 14.

Podemos encontrar também o sistema reduzido de resíduos:$1, 3, 5, 9, 11, 13$. Esse sistema reduzido de resíduos foi formado apenas com os elementos do sistema completo de resíduos que são primos com 14.

Logo como dito acima, temos que $\phi(14)$ é a quantidade dos elementos do sistema reduzido de resíduos, ou seja: $\phi(14) = 6$.

Generalizando, podemos escrever: $x \equiv y \mod n$, tem como sistema completo de resíduos: $0, 1, 2, 3, ..., n-1$ e como sistema reduzido de resíduos: Os elementos i tais que $(i, n) = 1$ e $i \in \{0, 1, 2, 3, 4, 5, ..., n-1\}$, ou seja, os elementos do sistema completo de resíduos que são primos com n.

Como calcular $\phi(n)$, ou seja, como calcular a quantidade de elementos do sistema reduzido de resíduos?

Começamos com o caso mais simples, quando $n = p$ ou seja se n é um número primo, temos que $(i, p) = 1$ para todos $i \in \{1, 2, 3, 4, 5, ..., p-1\}$, e, portanto, concluímos que o número de elementos desse sistema reduzido de resíduos é $p - 1$, isto é, $\phi(p) = p - 1$.

Agora que já sabemos como calcular a ordem de p primo, podemos com esse resultado calcular a ordem de p^k, com p primo e $k \in \mathbb{N}$.

Utilizando o mesmo pensamento, buscaremos encontrar todos os $(i, p^k) = 1$, onde $i \in \{1, 2, 3, ..., p-1\}$, ou seja, queremos encontrar os $i \in \{1, 2, 3, ..., p-1\}$ que não são divisíveis por p^k. Como estratégia, em primeiro lugar, contaremos todos os que são divisíveis por p^k e, em seguida, tiramos do total, restando assim, os que não são divisíveis por p^k.

Suponhamos $0 \leq a < p^k$, sendo a divisível por p, logo a se escreve: $a = p \cdot b$ e, portanto, $0 \leq b < p^{k-1}$.

Concluímos, então, que existem p^{k-1} inteiros que são divisíveis por p. Como temos um total de p^k números, temos que existem $p^k - p^{k-1}$ números que não são divisíveis por p.

Logo, $\phi(p) = p^k - p^{k-1} = p^{k-1} \cdot (p-1)$.

Falta agora provar a seguinte proposição:

Proposição 1. Se m e $n \in \mathbb{N}$ com $m > 1$ e $n > 1$ sendo $(m, n) = 1$ então: $\phi(m \cdot n) = \phi(m) \cdot \phi(n)$

Demonstração:

Para provar tal resultado será utilizada a abordagem feita por [**?**, p. 133]:

Consideremos a seguinte tabela formada pelos números naturais de 1 a $m \cdot n$:

1	2	...	k	...	n
$n+1$	$n+2$	...	$n+k$	...	$2 \cdot n$
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$(m-1) \cdots (n+1)$	$m-1 \cdots (n+2)$	...	$(m-1) \cdots (n+k)$	...	$m \cdot n$

Como se tem que $(t, m \cdot n) = 1$ se, e somente se, $(t, n) = (t, m) = 1$, para calcular $\phi(m \cdot n)$, devemos determinar os inteiros na tabela acima que são simultaneamente primos com n e m.

Se o primeiro elemento de uma coluna não for primo com n, então todos os elementos da coluna não são primos com n. Portanto, os elementos primos com n estão necessariamente nas colunas restantes que são em número $\phi(n)$, cujos elementos são primos com n, como é fácil verificar. Vejamos agora quais são os elementos primos com m em cada uma dessas colunas.

Como $(m, n) = 1$, a sequência :

$$k, n+k, ..., (m-1)n + k$$

forma um sistema completo de resíduo módulo m e, portanto, $\phi(m)$ desses elementos são primos com m. Logo, o número de elementos simultaneamente primos com n e m é $\phi(m) \cdot \phi(n)$.

Após provado que $\phi(m \cdot n) = \phi(m) \cdot \phi(n)$ tais que $mdc(m, n) = 1$ basta tomarmos a decomposição em fatores primos de n:

Se $n = p_1^{\alpha_1} \cdot ... \cdot p_n^{\alpha_n}$, temos: $\phi(n) = \phi(p_1^{\alpha_1} \cdot ... \cdot p_n^{\alpha_n}) = p_1^{\alpha_1 - 1} \cdot (p_1 - 1) \cdot ... \cdot p_n^{\alpha_n - 1} \cdot (p_n - 1) = p_1^{\alpha_1 - 1} \cdot ... \cdot p_n^{\alpha_n - 1} \cdot (p_1 - 1) \cdot ... \cdot (p_n - 1)$.

Assim sendo, podemos calcular a ordem de um número, bastando para isso conhecermos a sua decomposição em fatores primos. Porém, como vimos até então, saber tal decomposição nem sempre é tarefa fácil, pois, dependendo do tamanho do número em questão, torna-se complicado até mesmo saber se ele é primo. No capítulo a seguir será apresentado alguns testes de primalidade, estes testes são os utilizados nos dias atuais pelos computadores quando se precisa verificar a primalidade de um número.

Capítulo 2

TESTES DE PRIMALIDADE

Nesse capítulo procuraremos falar sobre testes de primalidade atuais, os quais são utilizados pelos computadores para se detectar se um número é primo, analisando as suas vantagens e desvantagens apresentadas. Para tanto, partiremos de uma ideia inicial utilizada pelo matemático Leibniz, a qual não estava totalmente certa, mas representou um avanço na maneira de se pensar e analisar a primalidade de um número. Após o matemático Fermat ter enunciado seu teorema tratado no capítulo anterior, Leibniz pensou que, se um número qualquer p satisfaz tal teorema, isso bastaria para tal número ser primo.

2.1 Teste De Leibniz

Como dito anteriormente, para Leibniz verificar se um número p era primo, ele verificava a congruência: $a^{p-1} \equiv 1 \mod p$, $\forall a$ tal que $(a, p) = 1$, então, se isso acontecesse, seria suficiente para que esse número fosse primo. Sem ao menos provar tal argumento, Leibniz utilizou essa recíproca como seu critério de primalidade, como observaremos abaixo.

Pequeno Teorema de Fermat: Se p for um número primo, então tem-se que p divide $a^{p-1} - 1$ com $(a, p) = 1$, ou seja, p é primo $\Rightarrow a^{p-1} \equiv 1 \mod p$.

Recíproca do Teorema de Fermat: Se p divide $a^{p-1} - 1$ sendo $(a, p) = 1$, então tem-se que p é um número primo, ou seja, $a^{p-1} \equiv 1 \mod p, \forall a \in \mathbb{Z}$ tal que $(a, p) = 1 \Rightarrow p$ é primo.

Um outro detalhe importante em seu teste de primalidade, é que Leibniz utilizava sempre $a = 2$, para tornar seus cálculos menores.

Teste de Primalidade 1. (Teste de Leibniz) Dado um número ímpar $n \in \mathbb{N}$, se $2^{n-1} \equiv 1 \mod n$, então, n é primo.

Então, se Leibniz quizesse descobrir se um número, por exemplo, 229 era primo, bastava para ele verificar se a congruência abaixo era satisfeita:

$2^{229-1} \equiv 2 \mod 229$

$2^{228} \equiv 1 \mod 229$

Com um pouco de cuidado e uma calculadora é possível verificar tal congruência e, portanto, afirmar que 229 é um número primo.

Porém, esse método nem sempre é verdadeiro, ou seja, a recíproca do Pequeno Teorema de Fermat não é válida, pois existem números compostos que também a verificam, exemplo o número 341:

$2^{341} \equiv 2 \mod 341$

Acontece que 341 não é um número primo, pois $341 = 11 \cdot 31$

Pode-se concluir então, que este método não traz a certeza de que o número verificado é primo, já que as vezes ele acerta e as vezes ele erra. Contudo, mesmo assim, ao se verificar uma grande quantidade de números utilizando tal método, foi constatado que ele acerta muito mais do que erra, e portanto um número que passa em tal teste tem grande possibilidade de ser primo. Quando um número passa por este método e não é primo, passa a se denominar pseudo-primo.

2.2 Teste de Miller

A incerteza causada pelo método de Leibniz não é boa quando o que precisamos é saber se um número é primo. Porém, uma mudança neste método realizada por G. L. Miller, em 1976, foi capaz de melhorá-lo aumentando sua precisão e retirando em parte esta incerteza.

O Teste de Miller é realizado da seguinte maneira:

Teste de Primalidade 2. (Teste de Miller)

Seja n o número a ser verificado, podemos admitir n ímpar uma vez que, o único primo par é 2. Se n é ímpar temos que $n-1$ é par e portanto pode ser escrito como:

$n - 1 = 2^k \cdot q$, onde extraímos a maior potência de 2, sendo assim q um número ímpar.

Então escolhe-se uma base b inteira, tal que $1 < b < n-1$, se algum número entre b^q, $b^{2 \cdot q}, \ldots, b^{2^{k-1} \cdot q}$ for congruente a (-1) módulo n, ou $b^{2^{k \cdot q}}$ for congruente a 1 módulo n, dessa forma temos que n é primo.

O exemplo abaixo utiliza um número pequeno para facilitar a compreensão:

Exemplo 8. Seja $n = 5021$, então $n - 1 = 5020 = 2^2 \cdot 1255$, ou seja:

$k = 2$ e $q = 1255$, escolhemos uma base b tal que $1 < b < 5020$, façamos então $b = 2$, se algum número entre:

2^{1255} e 2^{2510} for congruente a -1 módulo 5021, ou 2^{5020} for congruente a 1 módulo 5021, então consideremos 5021 um número primo.

Calculando assim tais congruências, teremos:

$2^{1255} \equiv 3658 \mod 5021$

$2^{2510} \equiv 5020 \mod 5021 \Rightarrow 2^{2510} \equiv -1 \mod 5021$ e temos para a base 2 que o teste é conclusivo.

É importante saber que o Teste de Miller também não dá certo sempre. Se acaso utilizarmos a base 2 e testarmos todos os números entre 1 e 1000000000 ele errará 1282 vezes, o que é um número muito pequeno de vezes. Quando um número composto passa pelo Teste de Miller tendo como resultado ser um número primo, dizemos que tal número é um pseudoprimo forte para a base utilizada. A seguir serão apresentados dois exemplos onde é utilizado o Teste de Miller

para verificar a primalidade de um número, para que assim possamos compreender melhor a sua utilidade.

Exemplo 9. Queremos saber se 7561 é um número primo:

- Escolhemos como base o número 17, lembrando que a base tem que estar sempre entre 1 e $n-1$.
- Tiramos uma unidade de 7561, ficando assim com 7560 e em seguida fazemos divisões sucessivas por 2 até encontrar um quociente ímpar.

$$7560 = 2^3 \cdot 945$$

- Calculamos os restos de $b^{q \cdot 2^i}$ por n, sendo no caso $i \in \{0, 1, 2, 3\}$.

$$\begin{aligned} b^{q \cdot i} &\equiv r \mod 7561 \\ 17^{945} &\equiv 1 \mod 7561 \\ 17^{945 \cdot 2} &\equiv 1 \mod 7561 \\ 17^{945 \cdot 4} &\equiv 1 \mod 7561 \\ 17^{945 \cdot 8} &\equiv 1 \mod 7561 \end{aligned}$$

- Os resultados são analisados da seguinte forma: se $i = 0$ e $r = 1$ ou $i \geq 0$ e $r = -1$, temos que 7561 é primo para essa base.

  Como temos $i = 0$ com $r = 1$, nosso resultado é que 7561 é primo para a base 17, ou seja, é um pseudoprimo forte para a base 17.

Exemplo 10. Queremos saber se 1256347 é um número primo:

- Escolhemos como base o número 5, lembrando que a base tem que estar sempre entre 1 e $n-1$;
- Tiramos uma unidade de 1256347, ficando assim com 1256346 e em seguida, fazemos divisões sucessivas por 2 até encontrar um quociente ímpar: $1256346 = 2 \cdot 628173$;
- Calculamos os restos de $b^{q \cdot 2^i}$ por n , sendo no caso $i \in \{0, 1\}$.

  $b^{q \cdot i} \equiv r \mod n$

  $5^{628173} \equiv -1 \mod 1256347$

  $5^{628173 \cdot 2} \equiv 1 \mod 1256347$

- Os resultados são analisados da seguinte forma: se $i = 0$ e $r = 1$ ou $i \geq 0$ e $r = -1$ temos que 1256347 é primo para esta base.

  Como temos $i = 0$ com $r = -1$, nosso resultado é que 1256347 é primo para a base 5, ou seja, é um pseudo-primo forte para a base 5.

Como já dito neste trabalho, ainda não existe um teste de primalidade que determine todos os números primos de modo eficiente. Nos casos dos números 7561 e 1256347, por serem números pequenos, os computadores atuais conseguem através do Teste de Miller trazer a certeza de que se trata de um número primo, basta para isso segundo Coutinho (2013) realizar tal teste para $\frac{1}{4}$ das bases entre 1 e $n-1$. Porém se um número for suficientemente grande, o número de bases entre 1 e $n-1$ também será, e assim será impossível, até mesmo para o computador, realizar este teste para $\frac{1}{4}$ das bases em tempo hábil. O próprio *software xMáxima*, que utilizamos para realizar as contas, possui um comando o qual realiza automaticamente um teste de primalidade e, então, consegue detectar que se trata de números primos.

O que se faz hoje em dia quando se precisa descobrir se um número é primo é utilizar o Teste de Miller para uma quantidade de bases que acredite ser suficiente, fazendo com que a chance de erro diminua ainda mais. Alguns softwares utilizam 10 bases previamente escolhidas, e afirmam que um número é primo quando ele consegue passar pelo Teste de Miller para todas as 10 bases.

A seguir analisaremos o Teste de Lucas, o qual é também um teste determinístico de primalidade com vantagens e desvantagens em relação ao Teste de Miller.

2.3 Teste de Lucas

Esse teste utiliza como estratégia tentar mostrar que a ordem $\phi(n)$ (phi de n) no número n a ser testado é $n-1$. Desta maneira sabemos que existem $n-1$ números menores que n os quais $(a,n)=1$, e sendo assim conclui-se que n é primo.

Teste de Primalidade 3. (Teste de Lucas)
Seja n um inteiro positivo ímpar e b um inteiro tal que $2 \leq b \leq n-1$. Se: $b^{n-1} \equiv 1 \mod n$ e $b^{\frac{n-1}{p}} \not\equiv 1 \mod n$ para cada fator primo p de $n-1$, então n é primo.

O teste de Lucas possui a vantagem de afirmar com certeza que um determinado número é primo, ou seja, se um número passar em tal teste ele com certeza é um número primo, diferentemente dos testes anteriores, onde quando um número passava no teste havia uma grande chance de ser primo, porém em alguns poucos casos o teste falhava.

Exemplo 11. Vamos utilizar o teste de Lucas para saber se o número 7561 é um número primo.

- Escolhemos uma base para realizar o teste, vamos utilizar $b = 888$
- Verificamos se a primeira condição é valida: $b^{n-1} \equiv 1 \mod n$

  $888^{7560} \equiv 1 \mod 7561$
- Verificamos se a segunda condição é válida:

  $b^{\frac{n-1}{p}} \not\equiv 1 \mod n$

  para cada fator primo p de $n-1$, então n é primo.

  Logo, para isso, fatoramos $n-1$. Utilizando o algoritmo da fatoração encontramos $n-1 = 2^3.3^3.5.7$ e, portanto, temos que verificar para 4 fatores primos:

$888^{\frac{7560}{2}} = 888^{3780} \equiv -1 \mod 7561$

$888^{\frac{7560}{3}} = 888^{2520} \equiv 6262 \mod 7561$

$888^{\frac{7560}{5}} = 888^{1512} \equiv 5935 \mod 7561$

$888^{\frac{7560}{7}} = 888^{1080} \equiv 5291 \mod 7561$

Uma vez tendo as duas condições satisfeitas podemos concluir que 7561 é um número primo.

Vale lembrar que, se uma das condições não é satisfeita no teste de Lucas, não significa que o número que temos não é primo, nesse caso a estratégia é trocar a base que estamos utilizando.

Exemplo 12. Queremos provar que 31 é primo utilizando o teste de Lucas:

- Escolhemos uma base para realizar o teste, vamos utilizar $b = 2$
- Verificamos se a primeira condição $b^{n-1} \equiv 1 \mod n$ é valida:
  $2^{30} \equiv 1 \mod 31$;
- Verificamos se a segunda condição $b^{\frac{n-1}{p}} \not\equiv 1 \mod n$ é válida para cada fator primo p de $n-1$:
  Logo, para isso, fatoramos $n - 1 = 30$. Utilizando o algoritmo da fatoração encontramos $30 = 2 \cdot 3 \cdot 5$ e, portanto, temos que verificar para 3 fatores primos.
  $2^{\frac{30}{2}} \equiv 1 \mod 31$

Observe que o teste falhou já no primeiro fator primo de $n - 1$, porém 31 é um número primo muito conhecido. Logo o teste tinha que ter uma resposta positiva, para isso só o que temos a fazer é trocar a base e realizar os passos novamente.

- Escolhemos uma nova base para realizar o teste, vamos utilizar $b = 12$;
- Verificamos se a primeira condição $b^{n-1} \equiv 1 \mod n$ é valida:
  $12^{30} \equiv 1 \mod 31$
- Verificamos se a segunda condição $b^{\frac{n-1}{p}} \not\equiv 1 \mod n$ é válida para cada fator primo p de $n-1$:
  $12^{\frac{30}{2}} \equiv -1 \mod 31$
  $12^{\frac{30}{3}} \equiv 25 \mod 31$
  $12^{\frac{30}{5}} \equiv 2 \mod 31$

Assim, após termos comprovados as duas condições, podemos afirmar que 31 é um número primo.

A escolha de uma base que satisfaça ambas as condições pode ser um tanto trabalhosa. Felizmente, segundo Coutinho (2013, p.171) "Em 1.975 Brilhart, Lehmer e Selfridge observaram que não é necessário usar uma base única no teste de Lucas. Podemos nos dar ao luxo de escolher uma base distinta para cada primo. Com isto obtemos um teste um pouco mais eficiente".

Seja $n > 0$ um inteiro tal que: $n - 1 = p_1^{e_1} \cdots p_r^{e_r}$, onde $p_1 < \cdots < p_r$ são primos. Se, para cada $i = 1, \cdots, r$, existirem inteiros positivos b_i com $2 \leq b_i \leq n - 1$ que satisfaçam:

$b_i^{n-1} \equiv 1 \mod n$

$b_i^{\frac{n-1}{p_i}} \not\equiv 1 \mod n$, então n é primo. Não sendo necessário que os b_i's sejam todos distintos.

Vejamos como utilizar tal versão aprimorada deste teste de primalidade:

Exemplo 13. Vamos utilizar o teste de Lucas para provar que o número 7 é um número primo.

- Escolhemos uma nova base para realizar o teste, valor utilizar $b = 2$
- Verificamos se a primeira condição $b^{n-1} \equiv 1 \mod n$ é valida:

  $2^6 \equiv 1 \mod 7$;
- Verificamos se a segunda condição é válida:

  $b^{\frac{n-1}{p}} \not\equiv 1 \mod n$ para cada fator primo p de $n - 1$. Assim sendo, n é primo.

  Logo, para isso fatoramos $n-1 = 6$. Utilizando o algoritmo da fatoração encontramos $6 = 2 \cdot 3$ e, portanto, temos que verificar para 2 fatores primos.

  $b^{\frac{n-1}{p}} \not\equiv 1 \mod n$

  $2^{\frac{6}{2}} \equiv 1 \mod 7$, e o teste falha, desta forma mudamos a base para 3.

  $3^{\frac{6}{2}} \equiv -1 \mod 7$

  $3^{\frac{6}{3}} \equiv 2 \mod 7$,

  tendo como saida assim que o número 7 é primo.

Apesar do teste de Lucas possuir a grande vantagem de ter como saída a certeza de que um determinado número é primo, ele padece também de um grande problema. Já que ao se observar com atenção seu enunciado percebemos que ele tem que ser verificado **para cada fator primo de** $n - 1$, ou seja, precisamos fatorar $n - 1$, e como já sabemos fatorar um número suficientemente grande pode levar tanto tempo que se torna inviável.

No teste de Lucas a estratégia utilizada é provar que a ordem de n é igual a $\phi(n)$, significando assim que se $a < n$ é um inteiro positivo, temos $(a, n) = 1$ o que significaria que n é primo. Para encontrar a ordem de um número n, ou seja, encontrar a quantidade de números menores que n e que também são primos com n, em um primeiro pensamento, pode se tentar contar os elementos um a um, tarefa essa impossível se o número em questão for grande o suficiente. Para contornar esse problema, faz se uso do Teorema da Raiz Primitiva.

Quando $\phi(n)$ é a ordem de a com respeito a n, então dizemos que a é uma raiz primitiva módulo n. a é raiz primitiva com respeito a $n \Leftrightarrow$ ordem de a com respeito a n é igual $\phi(n)$.

Teorema 4. (Teorema da Raiz Primitiva)

Se a é raiz primitiva de p, então:

$a \mod p, a^2 \mod p, a^3 \mod p, ..., a^{p-1} \mod p$, são distintos e pertencem a $\left\{\bar{1}...p\bar{-}1\right\}$, ou seja, a raiz primitiva gera todos os elementos do grupo multiplicativo $\mathbb{Z}_p$.

A seguir é apresentado um exemplo para melhor ilustrar o Teorema:

Exemplo 14. Vamos calcular a ordem de 5 com respeito a 9, o que podemos denotar como: $ord_9(5)$.

Como sabemos que $\phi(9) = \phi(3^2) = 3^{2-1} \cdot (3-1) = 6$ e que $ord_9(5)|\phi(9) = 6$, basta calcular as potências dos divisores de 6 para encontrarmos tal ordem:

$$5^1 \equiv 5 \mod 9$$
$$5^2 \equiv 7 \mod 9$$
$$5^3 \equiv 8 \mod 9$$
$$5^6 \equiv 1 \mod 9$$

Desta forma $ord_9 5 = 6$. Como $ord_9 5 = 6 = \phi(9)$, dizemos que 5 é uma raiz primitiva módulo 9.

O Teorema da raiz primitiva nos garante que sempre existirá um número b que cumpra as condições descritas acima se n for realmente um número primo.

No capítulo a seguir será abordado métodos criptográficos os quais se utilizam dos teoremas que foram apresentados até então, juntamente com as fragilidades dos testes de primalidade existente. Será possível analisar como a dificuldade em se fatorar um número serve de apoio para a Criptografia RSA e também como o Teorema da raiz primitva é útil para o método criptográfico Diffie-Hellman os quais estudaremos mais adiante.

Capítulo 3

MÉTODOS CRIPTOGRÁFICOS

Será abordado neste capítulo, uma aplicação matemática que nos últimos anos passou a tirar proveito da dificuldade em se fatorar um número e, também, da dificuldade em descobrir se um número ($\in \mathbb{N}$) é primo, a criptografia.

A criptografia é muito utilizada nos dias atuais para proteger transações financeiras realizadas através do computador. Um exemplo muito comum são as compras realizadas via Internet e pagas com cartão de crédito, nelas o comprador digita o número e alguns dados de seu cartão de crédito no site da loja e somente com estas informações é realizado o pagamento da compra. Se esses dados caíssem nas mãos de um terceiro, ele conseguiria utilizar este cartão para novas compras em seu nome, sem nenhuma dificuldade. Por esse motivo, essas informações do cartão de crédito precisam ser codificadas de maneira que somente o destinatário real (a loja) consiga compreender. Essa codificação precisa atender alguns requisitos para que possa ser eficaz, o primeiro deles é que o destinatário (em nosso exemplo, a loja) saiba como decodificar os dados. Portanto, a codificação não pode ocorrer de forma aleatória, significando assim que tanto a loja quanto o comprador precisam saber como codificar tais dados. Mas, nesse ponto temos um problema, pois a comunicação entre a loja e o comprador está sendo realizada via *internet* e, da mesma maneira que os dados do cartão de crédito podem ser interceptados por um terceiro (e por este motivo precisam ser codificados), a codificação utilizada também corre este risco. Assim quando a loja comunica ao comprador o código que ele utilizará para codificar, esse código não é mais seguro.

Tal fato era verdade até algum tempo atrás, pois antigamente as maneiras conhecidas de se codificar tinham uma relação muito forte com as de decodificar, bastando para isso “desfazer"a codificação (voltar os passos realizados para codificar). O que acontece é que quando vamos codificar alguma informação, existe uma “receita"contendo o passo a passo do processo de codificação, chegando assim ao terminar esse processo no resultado final que é a mensagem codificada. Se essa maneira de codificar (“receita") caísse nas mãos de um terceiro, juntamente com a mensagem codificada, bastava para ele voltar os passos realizados para codificar, conseguindo decodificar tal mensagem.

Como dito anteriormente isto era verdade até algum tempo atrás, o que acontece é que existe uma maneira de se codificar a qual não sofre deste mal, pois ao tentar realizar tal processo inverso

da codificação, esse processo se torna tão trabalhoso a ponto de ser impossível ser realizado, ou seja, um processo fácil de ser feito mais muito difícil de ser desfeito, garantindo assim que mesmo se um terceiro soubesse como a codificação foi realizada, este terceiro não conseguiria decodificar os dados. Essa nova maneira de se codificar é conhecida como Criptografia de Chave Pública, levando em conta que o processo de codificação pode se tornar público sem comprometer a segurança do método.

Na Criptografia de Chave Pública a codificação é realizada utilizando uma chave (método de codificação) pública, porém para ser decodificada se faz necessário o uso de uma outra chave (método), chave esta que, fica privada somente ao destinatário da mensagem, ou seja, aquele que decodificará tal mensagem. O que resolve o problema descrito acima, haja visto que desta forma a pessoa que irá codificar os dados não precisa conhecer o método de decodificação e, dessa forma, mesmo que exista um terceiro que encontre a mensagem criptografada junto com o método de codificação, este terceiro não conseguirá decodificar tais dados.

A Criptografia de Chave Pública utilizada no exemplo da compra via internet paga com cartão de credito é conhecida como Criptografia RSA, sigla essa que leva as iniciais de seus criadores R.L.Rivest, A. Shamir e L. Adleman, como homenagem. Vejamos a seguir como tal método se utiliza da matemática para seu funcionamento.

3.1 Criptografia RSA

Aqui vamos entender um pouco como funciona o método RSA. A chave de codificação utilizada neste método é composta pelo número n formado pelo produto de dois números primos distintos a e b, tal chave é conhecida como chave pública. A chave de decodificação é composta pelos dois números primos a e b, essa é a chave privada. Embora exista aparentemente uma proximidade entre ambas, não é possível encontrar a chave privada (os dois números primos a e b) a partir da chave pública n, pois, para isso, precisaríamos decompor a chave n. Como vimos até agora, se um número for grande o suficiente, se torna impossível até mesmo com a utilização de um computador decompor esse número, em tempo hábil.

Dados necessários para a utilização da criptografia RSA:

- escolhe-se dois números primos distintos p e q grandes o suficiente e calcula-se o produto $p \cdot q$, o resultado chamamos de n $(n = p \cdot q)$:
- n é utilizado para codificar a mensagem (n é a chave pública);
- p e q são utilizados para decodificar a mensagem (p e q são as chaves privadas);
- para quebrar o código RSA é necessário fatorar n descobrindo assim os números p e q, porém ao se utilizar números grandes o suficiente (de cerca de 100 digitos), fatorá-lo se torna um processo inviável, devido ao tempo que os métodos atuais levariam, sendo assim impossível de colocá-los em prática.

Assim sendo, as chaves de codificação e de decodificação são números, e a codificação é realizada por meio do cálculo de uma potência módulo n (chave pública de codificação). Para que isso seja

possível, a mensagem a ser codificada precisa ser um número inteiro e, portanto, quando tal mensagem for um texto devemos converter tal texto em uma sequência de números para assim codificá-la.

1. Na primeira etapa, converteremos as letras em números de dois dígitos para evitar ambiguidades, conforme a tabela a seguir:

A	B	C	D	E	F	G	H	I	J	K	L	M
10	11	12	13	14	15	16	17	18	19	20	21	22
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
23	24	25	26	27	28	29	30	31	32	33	34	35

   Na conversão, o espaço entre duas palavras é substituido pelo número 99 e, desta forma, temos a mensagem convertida para uma sequência de números.

2. Feito toda a conversão da mensagem a ser codificada para números, o próximo passo é a escolha dos dois números primos distintos p e q, os quais chamaremos de parâmetros e o cálculo de n, tal que $n = p \cdot q$.

3. O último passo antes de iniciarmos a codificação em si, é quebrar a sequência de números em blocos, de maneira que cada bloco produzido seja menor que o número n. Existem diversas maneiras em que pode-se quebrar os blocos, porém, eles não podem ser iniciados com 0, a fim de evitar problemas na hora de decodificar. Esses blocos não correspondem a letras, símbolos ou qualquer outra unidade linguística e, portanto, elimina a possibilidade da utilização da decodificação por frequência.

4. Após esta etapa, já pode-se dar início a codificação da mensagem que esta divida em blocos. Para codificar a mensagem é utilizada a chave pública a qual é composta pelos números n e e, sendo e um inteiro positivo inversível módulo $\phi(n)$, ou seja, $mdc(e, \phi(n)) = 1$. Observe que o cálculo de $\phi(n)$ é possível se soubermos a chave de decodificação, pois $n = p.q$ e assim sendo $\phi(n) = \phi(p \cdot q) = (p-1) \cdot (q-1)$. Dessa forma, juntamente com n temos como parte da chave de codificação o número e.

5. Seja b o bloco a ser codificado, logo b é um inteiro positivo menor que n. Denotando o bloco codificado como $C(b)$, calculamos $C(b)$ através do seguinte cálculo: $b^e \equiv C(b) \mod n$, sendo $0 \leq C(b) < n$, dessa forma podemos dizer que $C(b)$ é o resto da divisão de b^e por n.

   Feita a codificação de todos os blocos, blocos esses que, após terem sido codificados, não podem mais serem reunidos de modo a formar novamente um grande número, pois se isto ocorrer será impossível sua decodificação.

6. Para realizar a decodificação se faz necessário a posse de dois números: n e d , onde d é o inverso de e módulo $\phi(n)$. Os números d e e formam a chave de decodificação.

   De posse da chave de decodificação juntamente com os blocos já codificados, precisamos compreender quais são os passos que nos permitem reconstruir os blocos originais de antes da codificação.

7. Seja a o bloco codificado, o qual queremos decodificar, e $D(a)$ o resultado do processo desta decodificação. Calculamos $D(a)$ da seguinte forma: $a^d \equiv D(a) \mod n$, sendo $0 \leq D(a) < n$, dessa forma podemos dizer que $D(a)$ é o resto da divisão de a^d por n.

   Desta forma temos que encontrar d. Observe que ele pode ser calculado com facilidade através do algoritmo euclidiano estendido, desde que $\phi(n)$ e e sejam conhecidos.

8. Após este passo precisamos provar que, sendo b um bloco da mensagem original, temos $D(C(b)) = b$, ou seja, decodificando um bloco da mensagem codificada, sempre encontramos o bloco da mensagem original correspondente. Em seguida, basta colocar novamente os blocos em uma sequência longa de números e convertê-los na mensagem original.

A seguir um exemplo de como a Criptografia RSA pode ser utilizada:

Exemplo 15. Vamos codificar a mensagem "NÚMEROS INTEIROS".

1. Num primeiro momento devemos converter as letras em números com auxílio da tabela:

A	B	C	D	E	F	G	H	I	J	K	L	M
10	11	12	13	14	15	16	17	18	19	20	21	22
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
23	24	25	26	27	28	29	30	31	32	33	34	35

   Assim sendo, conseguimos a sequência numérica a seguir:

   2330221427242899182329141827242 8

2. O próximo passo é a escolha dos parâmetros, ou seja, dos dois números primos p e q. Para simplificar um pouco, escolheremos números primos pequenos a fim de facilitar as contas, as quais podem ser realizadas até mesmo com a calculadora científica do computador, tornando mais evidente o método que está sendo utilizado.

   Seja então, $p = 11$ e $q = 13$, sendo assim temos $n = 143$

3. Temos que quebrar esta sequência numérica em pequenos blocos, de maneira que cada bloco seja menor que o número n, em nosso caso menor que 143.

   2330221427242899182329141827242 8

   Assim procedendo, temos os blocos separados por um traço:

   $23 - 30 - 22 - 142 - 72 - 42 - 89 - 91 - 82 - 32 - 91 - 41 - 82 - 72 - 42 - 8$

4. Em seguida calculamos $\phi(n)$ a fim de encontrar um número e tal que $mdc(e, \phi(n)) = 1$, onde: $\phi(n) = \phi(p.q) = \phi(13.11) = 12.10 = 120$

   Também, a fim de facilitar os cálculos, visto que, o objetivo deste exemplo é evidenciar o método que está sendo utilizado, utilizaremos $e = 7$, podemos fazer isto pois $mdc(7, 120) = 1$.

   Produzimos assim os itens necessário da chave pública (de codificação), que são os números "e" e "n".

5. De posse da chave de codificação e dos blocos convertidos em números, podemos dar início à codificação:

   $23 - 30 - 22 - 142 - 72 - 42 - 89 - 91 - 82 - 32 - 91 - 41 - 82 - 72 - 42 - 8$

   $b^e \equiv C(b) \mod n$, sendo $C(b) < n$

   $23^7 \equiv 23 \mod 143$

   $30^7 \equiv 134 \mod 143$

   $22^7 \equiv 22 \mod 143$

   $142^7 \equiv 142 \mod 143$

   $72^7 \equiv 19 \mod 143$

   $42^7 \equiv 81 \mod 143$

   $89^7 \equiv 67 \mod 143$

   $91^7 \equiv 130 \mod 143$

   $82^7 \equiv 69 \mod 143$

   $32^7 \equiv 98 \mod 143$

   $91^7 \equiv 130 \mod 143$

   $41^7 \equiv 24 \mod 143$

   $82^7 \equiv 69 \mod 143$

   $72^7 \equiv 19 \mod 143$

   $42^7 \equiv 81 \mod 143$

   $8^7 \equiv 57 \mod 143$

   Feito os cálculos, temos os blocos agora codificados:

   $23 - 134 - 22 - 142 - 19 - 81 - 67 - 130 - 69 - 98 - 130 - 24 - 69 - 19 - 81 - 57$

6. Para realizar a decodificação da mensagem, se faz necessário os números n e d, então devemos calcular o valor de d para que possamos prosseguir com a decodificação:

   $d.e \equiv 1 \mod \phi(n)$

   $d.7 \equiv 1 \mod 120$

   $103.7 \equiv 1 \mod 120$, encontramos $d = 103$, utilizando aqui o algoritmo de Euclides estendido com os números 7 e 120.

   Temos assim a chave privada (de decodificação), composta por d e n.

7. De posse da chave privada ($n = 143, d = 103$) e dos blocos codificados, podemos iniciar a decoficação que acontecerá da seguinte maneira:

   $23 - 134 - 22 - 142 - 19 - 81 - 67 - 130 - 69 - 98 - 130 - 24 - 69 - 19 - 81 - 57$

   $a \equiv D(a)^d \mod n$, sendo $D(a) < n$, a um bloco codificado e $D(a)$ o bloco já decodificado.

$a \equiv D(a)^d \mod n$

$23^{103} \equiv 23 \mod 143$

$134^{103} \equiv 30 \mod 143$

$22^{103} \equiv 22 \mod 143$

$142^{103} \equiv 142 \mod 143$

$19^{103} \equiv 72 \mod 143$

$81^{103} \equiv 42 \mod 143$

$67^{103} \equiv 89 \mod 143$

$130^{103} \equiv 91 \mod 143$

$69^{103} \equiv 82 \mod 143$

$98^{103} \equiv 32 \mod 143$

$130^{103} \equiv 91 \mod 143$

$24^{103} \equiv 41 \mod 143$

$69^{103} \equiv 82 \mod 143$

$19^{103} \equiv 72 \mod 143$

$81^{103} \equiv 42 \mod 143$

$57^{103} \equiv 8 \mod 143$

E assim, conseguimos realizar toda a decodificação, encontrando novamente os blocos numéricos de antes da codificação:

$23 - 30 - 22 - 142 - 72 - 42 - 89 - 91 - 82 - 32 - 91 - 41 - 82 - 72 - 42 - 8$

8. Como último passo, devemos então, juntar os blocos de maneira a formar uma grande lista de números e em seguida converter novamente os números nas letras conforme tabela descrita anteriormente.

   233022142724289918232914182724 28

   $23 - 30 - 22 - 14 - 27 - 24 - 28 - 99 - 18 - 23 - 29 - 14 - 18 - 27 - 24 - 28$

   NÚMEROS INTEIROS

Ao observar a codificação e a decodificação do método de Criptografia RSA observamos que, para codificar, utilizamos:

$b^e \equiv C(b) \mod n$, com $C(b) < n$, e, para decodificar, utilizamos:

$a^d \equiv D(a) \mod n$, com $D(a) < n$.

Porém, observamos que na etapa de codificação utilizamos $C(b)$ para o bloco já codificado e na etapa de decodificação este mesmo bloco codificado passou a se denominar a, logo $C(b) = a$.

Logo, temos que provar que $D(C(b)) = b$ para $1 \leq b \leq n - 1$.

Desta forma podemos utilizar congruências, pois tanto $D(C(b))$ quanto b são menores que n : $D(C(b)) \equiv D(b^e) \equiv (b^e)^d \equiv b^{e \cdot d} \mod n$.

Lembrando que: $e \cdot d \equiv 1 \mod \phi(n)$, ou ainda, $e \cdot d = 1 + k \cdot \phi(n)$, com $k \in \mathbb{Z}$, segue que:
$b^{e \cdot d} \equiv b^{1+k \cdot \phi(n)} \mod n$.
$b^{1+k \cdot \phi(n)} \equiv b^1 \cdot \left(b^{\phi(n)}\right)^k \mod n$
Como sabemos fatorar n, podemos calcular $\phi(n)$ ($\phi(n) = \phi(p \cdot q) = (p-1) \cdot (q-1)$), substituindo em nossa congruência:
$b^{1+k \cdot \phi(n)} \equiv b^1 \cdot \left(b^{((p-1) \cdot (q-1))}\right)^k \mod n$
$b^1 \cdot b^{(p-1) \cdot (q-1) \cdot k} \equiv b^1 \cdot \left(b^{(p-1)}\right)^{(q-1) \cdot k} \mod n$
Temos até o momento:
$D(C(b)) \equiv D(b^e) \equiv (b^e)^d \equiv b^{e \cdot d} \equiv b^1 \cdot \left(b^{(p-1)}\right)^{(q-1) \cdot k} \mod n$
Pelo Pequeno Teorema de Fermat sabemos que:
$b^1 \cdot \left(b^{(p-1)}\right)^{(q-1) \cdot k} \equiv b \cdot 1^{(q-1) \cdot k} \mod p$
Analogamente encontramos que :
$b^1 \cdot \left(b^{(q-1)}\right)^{(p-1) \cdot k} \equiv b \cdot 1^{(p-1) \cdot k} \mod q$
Como p e q são primos distintos , temos $(p, q) = 1$, das congruências acima temos:
$p | b^{(e \cdot d)} - b$ e $q | b^{(e \cdot d)} - b$, e assim podemos concluir que:
$p \cdot q | b^{(e \cdot d)} - b$, como $n = p \cdot q$, podemos escrever em termos de congruência:
$b^{(e \cdot d)} \equiv b \mod n$
Concluimos assim, que:
$D(C(b)) \equiv b \mod n$.

Mostramos, que esse método funciona, mas precisamos agora entender um pouco sobre o que torna o RSA seguro. A segurança deste método reside no fato de que a pessoa que codifica os dados tem acesso apenas a chave pública (os números n e e) enquanto que para decodificar os dados faz-se necessário conhecer a chave privada (os números n e d), ou seja o RSA se torna seguro devido a grande dificuldade em calcular d conhecendo apenas n e e.

Sabemos que d é o inverso de e módulo $\phi(n)$, desta forma bastaria aplicar o algoritmo de Euclides estendido aos números e e $\phi(n)$, porém assim se faz necessário conhecer $\phi(n)$ e, para isso, devemos fatorar n, o que já vimos ser muito difícil se n for grande o suficiente. Logo, só é possível calcular d se soubermos os valores de p e q tal que $n = p \cdot q$ e tais valores só estão disponíveis a quem decodificará a mensagem, tornando o método de Criptografia RSA seguro.

Assim sendo, podemos entender como e porquê este método de Criptografia funciona. Em seguida será apresentada uma aplicação mais prática que a anterior, utilizando números maiores para codificar e decodificar os dados de um cartão de crédito, que foi utilizado em uma compra via *internet*. As contas realizadas no exemplo a seguir foram feitas utilizando o *software xMaxima*.

Exemplo 16. João está realizando uma compra via *internet*, a qual decidiu pagar com seu cartão de crédito. Ao efetivar sua compra, João deve preencher os seus dados pessoais bem como seu endereço para o qual será enviado seu produto e, em algum momento, após esta etapa ele será encaminhado a uma página na qual deverá efetuar o pagamento pela compra realizada. Nessa página João deverá informar os dados do seu cartão de crédito e, ao digitar tais dados, seu computador codificará esses dados com a chave pública que foi enviada pelo site da loja.

A loja é a responsável pela distribuição da chave pública, dessa maneira vários compradores podem utilizar a mesma chave pública sem nenhum problema, pois somente ela tem a chave privada a qual permite decodificar tais dados e, assim, conseguir compreender os dados enviados. Desse

modo, a loja envia ao computador de João a chave pública composta dos números $n = 11.413$ e $e = 3$.

João digita o número de seu cartão de crédito: 1234567812345678, o seu computador os separa em blocos e em seguida os codifica:

$123 - 4567 - 8123 - 456 - 78$

$123^3 \equiv 548 \mod 11413$

$4567^3 \equiv 1558 \mod 11413$

$8123^3 \equiv 10.120 \mod 11413$

$456^3 \equiv 11.025 \mod 11413$

$78^3 \equiv 6.619 \mod 11413$

O Computador após codificar os blocos em que foram separados os dados, os envia para a loja:

$548 - 1558 - 10120 - 11025 - 6619$

A loja, de posse da chave privada composta pelos números $d = 7467$ e $n = 11413$, recebe os blocos devidamente codificados e então realiza os cálculos para decodificar os dados contidos nos blocos.

$548^{7467} \equiv 123 \mod 11413$

$1558^{7467} \equiv 4567 \mod 11413$

$10120^{7467} \equiv 8123 \mod 11413$

$11025^{7467} \equiv 456 \mod 11413$

$6619^{7467} \equiv 78 \mod 11413$

Encontrando assim os blocos já decodificados:

$123 - 4567 - 8123 - 456 - 78$

Bastando agora juntá-los, voltando assim a forma original dos dados do cartão de crédito. Uma vez com os dados necessários em mãos, a loja efetua o pagamento e envia a mercadoria, concretizando a compra.

Os números primos utilizados no exemplo acima possuem apenas 3 casas decimais e, mesmo assim, já se faz necessário o uso de um *software* matemático capaz de calcular com números grandes, em operações reais os números primos utilizados possuem cerca de 100 dígitos.

3.2 Criptografia Diffie-Hellman

Uma outra criptografia que também utiliza chave pública é a criptografia de Diffie-Hellman, esse método tem sua segurança baseada na dificuldade em se encontrar logarítmos discretos e foi criado para solucionar o seguinte problema:

Imagine que duas pessoas Alice e Bob precisam se comunicar via *internet* de maneira segura, ou seja, de forma que somente Alice leia as mensagens enviadas por Bob e de que somente Bob leia as mensagens enviadas por Alice. Para ter segurança tais mensagens devem ser codificadas a fim de evitar que um terceiro as interceptem obtendo acesso as informações descritas nas mensagens.

Pensando assim, quando Alice mandar uma mensagem para Bob, ela deve codificá-la de maneira que somente Bob consiga decodificar e ler a mensagem, e também quando ele mandar uma mensagem para Alice ele deve condificá-la de maneira que somente ela consiga decodificar e ler a

mensagem. O Problema aqui é que, quando um deles transmite a mensagem codificada deve também enviar uma chave contendo as informações necessárias para que o receptor possa decodificá-las e, se essa informação for interceptada, um terceiro pode ter acesso as mensagens trocadas, deixando assim de serem secretas.

Esse problema teria solução simples se Alice e Bob pudessem se encontrar e combinar com antecedência a chave que utilizariam, porém o problema aqui apresentado trata de situações onde o único meio de comunicação possível é a internet e, portanto, eles devem encontrar uma maneira segura de combinarem uma chave a ser usada, podendo assim se comunicarem de maneira segura.

Segundo Pires (2010), Whitfield Diffie e Martin Hellman propuseram uma solução para este problema, a qual ficou conhecida como protocolo Diffie-Hellman para a troca de chaves. O Protocolo Diffie-Hellman é uma maneira segura de compartilhar uma chave, através de uma série de procedimentos, duas pessoas conseguem compartilhar uma chave de codificação e decodificação sem que mais ninguém tenha acesso, podendo utilizá-la para codificar as mensagens que serão enviadas ao parceiro e decodificar as mensagens que serão recebidas pelo mesmo.

Vejamos como funciona tal procedimento para troca de chaves, talvez a melhor maneira de se compreender o funcionamento deste método Criptografico seja utilizando um “esquema de cores". Vamos voltar ao nosso problema de início, no qual duas pessoas precisavam se comunicar via *internet* de maneira segura, a Alice e o Bob.

Antes de iniciar a conversa Alice e Bob devem utilizar a troca de chaves, para isto eles seguem os passos a seguir:

1. Combinam uma cor que será de conhecimento de ambos (c), além dessa cor cada um também escolhe uma outra cor secreta a qual não mostrarão a ninguém.

2. Alice mistura a cor comum (c) a sua cor secreta (a) produzindo uma nova cor (ca) e então a envia para Bob.

3. Bob também mistura a cor comum (c) a sua cor secreta (b) produzindo também uma nova cor (cb) e então a envia para Alice.

4. Alice recebe a cor obtida da mistura feita por Bob (cb) e então adiciona a esta mistura sua cor secreta obtendo uma nova cor (cba).

5. O mesmo é feito por Bob, ele pega a mistura feita por Alice e adiciona sua cor secreta, obtendo assim a mesma nova cor (cab) obtida por Alice.

Mesmo que em ordem diferente, a cor encontrada por Bob e a cor encontrada por Alice são iguais, pois foram produzidas com os mesmos ingredientes, esta nova cor é o segredo comum de Alice e Bob.

Se alguém interceptar as cores enviadas entre Alice e Bob, essa pessoa não conseguirá descobrir os ingredientes necessários para obter a nova cor que é o segredo a ser utilizado por Alice e Bob. Isso devido ao fato de que antes de enviar a cor secreta, cada um a misturou a cor comum formando assim uma nova cor, e após ter misturado duas cores obtendo uma terceira é impossível desfazer o processo para descobrir as cores secretas de Alice e de Bob as quais são ingredientes para a cor secreta.

Ao compararmos o procedimento das cores ao método Diffie-Hellman temos que a cor secreta obtida será a chave secreta que Alice e Bob utilizariam para conversarem em segurança, vejamos como tal método funciona utilizando números ao invés de cores.

1. Alice e Bob combinam dois números $\alpha, p \in \mathbb{N}$, sendo p primo e α uma raiz primitiva de p, esses números são a chave pública do método.

2. Em seguida, Alice escolhe um número que seja menor que p e o mantém em segredo, ou seja, $x_a < p$. Bob também escolhe um número menor que p e o mantém em segredo, ou seja, $x_b < p$.

3. O próximo passo de Alice é misturar seu número secreto com a chave pública, para isso calcula y_a da seguinte forma:

   $\alpha^{x_a} \equiv y_a \mod p$, sendo $y_a < p$.

   Bob também executa esse passo misturando seu número secreto com a chave pública, produzindo assim y_b da seguinte forma:

   $\alpha^{x_b} \equiv y_b \mod p$, sendo $y_b < p$.

4. Nesta etapa Alice envia para Bob o número obtido de seu cálculo: y_a e Bob também envia para Alice o número obtido de seu cálculo: y_b.

5. Alice então recebe o número y_b enviado por Bob e faz o seguinte cálculo para obter o número k:

   $y_b^{x_a} \equiv k \mod p$.

   Bob também realiza seus cálculos para obter k:

   $y_a^{x_b} \equiv k \mod p$.

Com isto Alice e Bob chegam ao mesmo número k, sem terem compartilhado tal número com mais ninguém, essa é a chave privada do método Diffie-Hellman.

Vejamos um exemplo numérico para melhor entendermos o funcionamento deste método Criptográfico:

Exemplo 17. 1. Alice e Bob combinam a chave pública que será utilizada: 5 e 113, sendo que 5 é raiz primitiva de 113.

2. Alice escolhe seu número privado: 16.

   Bob também escolhe seu número privado: 73.

3. Alice mistura seu número privado com a chave pública:

   $5^{16} \equiv 30 \mod 113$, e encontra assim o número 30, Bob também executa esse passo misturando seu número privado com a chave pública:

   $5^{73} \equiv 76 \mod 113$, encontrando assim o número 76.

4. Alice envia para Bob o número 30, e Bob envia para Alice o número 76.

5. Alice então recebe o número 76 enviado por Bob e faz o seguinte cálculo para obter o número k :

   $76^{16} \equiv k \mod 113$, sendo $0 \leq k < 113$

   $76^{16} \equiv 106 \mod 113$

   Encontrando $k = 106$.

6. Bob recebe o número 30 enviado por Alice e faz o seguinte cálculo para obter o número k :

   $30^{73} \equiv k \mod 113$, sendo $0 \leq k < 113$

   $30^{73} \equiv 106 \mod 113$

   Encontrando $k = 106$.

Alice e Bob agora possuem a chave privada 106, que será utilizada por ambos para codificarem e decoficarem suas mensagens e assim se comunicarem em segurança.

O próximo passo é entender o porque deste método ser considerado seguro, apesar de toda a combinação da chave a ser utilizada ter ocorrido via internet, sendo assim passível de interceptação. Vejamos:

- Ao iniciar Alice e Bob combinaram dois números $a = 5$ e $p = 113$ que formariam a chave pública. Esses números foram enviados de um para o outro e portanto poderiam cair nas mãos de um interceptador que no caso em questão chamaremos de Eva.

- Após Alice ter "misturado"seu número secreto com a chave pública ela envia o número encontrado $y_a = 30$ para Bob. Ao ser enviado esse número também pode cair nas mãos de Eva.

- Bob também envia o número $y_b = 76$ encontrado da "mistura"da chave pública com seu número secreto. Esse número ao ser enviado também pode ser interceptado por Eva.

Desta forma Eva ao investigar a conversa entre Alice e Bob consegue interceptar os números a, p, y_a e $y_b (5, 113, 30, 76)$, porém nenhum deles corresponde a chave secreta k que será utilizada por Alice e Bob.

O número $k = 106$ foi calculado das seguintes forma:
$(5^{16})^{73} \equiv (30)^{73} \equiv 106 \mod 113$
$(5^{73})^{16} \equiv (76)^{16} \equiv 106 \mod 113$
Para que Eva consiga descobrir o valor de k ela precisa resolver:
$5^{x_a} \equiv 30 \mod 113$ ou
$5^{x_b} \equiv 76 \mod 113$
Neste caso temos que x_a é chamado de logaritmo discreto de 30 módulo 113 na base 5, temos também que x_b é o logaritmo discreto de 76 módulo 113 na base 5. Podemos denotá-los por:
$dlog_{5,113}(30) = x_a$ e $dlog_{5,113}(76) = x_b$.

Os logaritmos discretos são os responsáveis por garantir a segurança do método Diffie-Hellman, isso devido a complexidade em resolver tais congruências como as descritas acima. Obviamente existem *softwares* computacionais capazes de resolver tais congruências, mas isso é devido a estarmos utilizando apenas números pequenos. Porém, na prática, o Algoritmo Diffie-Hellman utiliza um número primo de centenas de dígitos, aumentando de tal maneira as possibilidades para tal logaritmo discreto que os *softwares* conhecidos para se calcular demorariam anos para conseguir encontrar uma resposta, tornando assim tal busca por encontrar a chave utilizada um fracasso.

Os *softwares* computacionais inventados para calcular estes logaritmos discretos operam pelo método da tentativa e erro. Portanto, quanto maiores forem os números que estão sendo utilizados, mais tempo este *software* demorará para encontrar uma resposta, até que em determinado ponto os números utilizados sejam grande o suficiente para que o tempo gasto torne estes *softwares* ineficazes e garantindo a segurança deste método criptográfico.

Após ter comprovado a segurança da chave utilizada no método Diffie-Hellman, pode-se dar início a etapa de codificação. Nesta etapa o método Diffie-Hellman já foi utilizado, na verdade tal método é util apenas para o acordo da senha que sera utilizada durante a conversa para codificação e decodificação das mensagens, e se faz necessário utilizar um método de codificação e decodificação das mensagens.

Para concluírmos o nosso exemplo da Alice e do Bob, utilizaremos um método simples, onde mudamos a posição do alfabeto conforme a chave obtida, lembrando que a chave que Alice e Bob tinham combinado era 106. tabela. Na verdade tem mais opções, mas estas são as mais simples.

/	a	b	c	d	e	f	g	h	i	j	k	l	m
1	B	C	D	E	F	G	H	I	J	K	L	M	N
0	A	B	C	D	E	F	G	H	I	J	K	L	M
6	G	H	I	J	K	L	M	N	O	P	Q	R	S

/	n	o	p	q	r	s	t	u	v	w	x	y	z
1	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
0	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
6	T	U	V	W	X	Y	Z	A	B	C	D	E	F

Alice envia para Bob a seguinte mensagem:

"Qual mesmo o dia do aniversário de sua mãe?"

Codificando a mensagem utilizando alternadamente os alfabetos permutados de ordem $(1, 0$ e $6)$ obtém-se:

RUGNEYNOUEIGEOGOIBFRYBROPDKTUGNAK

Como Bob possui a chave de codificação e decodificação, ele decodifica, compreende a mensagem e, então, responde à Alice:

"Será dia nove"

Que é enviada até a Alice codificada:

TEXBDOBNUWE

Alice, por possuir o código, compreende a resposta enviada por Bob. Qualquer outra pessoa que intercepte a conversa obterá a conversa criptografada:

Alice: RUGNEYNOUEIGEOGOIBFRYBROPDKTUGNAK

Bob: TEXBDOBNUWE,

não sendo possível descobrir qual o significado.

Na prática existem *softwares* capazes de aplicar o Algoritmo Diffie-Hellman, bem como realizar toda a parte de codificação e decodificação, ficando por parte do usuário somente escrever o texto em tal *software* munido de sua senha privada. Toda a informação enviada de um computador para o outro acontecerá de maneira segura. Porém, esse método sofre de uma fraqueza se comparado a criptografia RSA.

Se o computador da Alice ou o do Bob for invadido por um *hacker* e este por sua vez obtiver as informações contidas no computador, o método Diffie-Hellman perde sua segurança assim, tal *hacker* consegue acesso direto à senha de codificação e decodificação, não sendo necessário ele interceptar os dados que estão sendo enviandos pela *internet* e descobrir o logaritmo discreto, o qual comentamos anteriormente. Como a senha de codificação e decodificação é uma só, se tal hacker conseguir acesso a um dos computadores utilizados para a comunicação, a segurança do método Diffie-Hellman ficará comprometida.

Dizemos assim, que, para que o método Diffie-Hellman ser realmente eficaz, ele deverá ocorrer em um ambiente em que se possa garantir a segurança dos computadores contra vírus e invasões de *hackers*. Tal mal já não ocorre na criptografia RSA, pois mesmo que o hacker obtenha todo o acesso ao computador de quem está codificando, conseguindo acesso à chave de codificação, essa chave não é util para decodificar os dados.

A criptografia RSA possui a vantagem de somente o computador do destinatário precisar segurança. Por esse motivo, o método criptográfico utilizado nas operações financeiras é o RSA, o que o torna mais utilizado do que o método Diffie-Hellman, porém isso não significa que o último não tenha suas aplicações, ele pode ser usado em conversas entre duas pessoas, por exemplo, na troca de *e-mails* confidenciais entre duas pessoas.

A seguir será apresentado um modelo de sequência didática, no qual aplicaremos os conceitos vistos aqui, no cotidiano escolar.

Capítulo 4

SEQUÊNCIA DIDÁTICA

Será apresentada a seguir uma sequência didática idealizada tendo como intuito o ensino dos conteúdos matemáticos contidos neste trabalho. Ela será apresentada através de uma abordagem investigativa em sala de aula, na qual será abordada a dificuldade em se encontrar números primos e em se fatorar números inteiros. Dificuldades essas que servirão de suporte para uma aplicação: a Criptografia de chave pública. Vale lembrar que esta é uma sequência didática idealizada e, desse modo, deverá ser adaptada a cada realidade escolar visando suprir sempre às necessidades da escola em questão.

A sequência didática apresenta alguns conteúdos matemáticos referentes aos aprendidos pelos alunos durante o Ensino Fundamental e Médio, sendo que os conteúdos restantes podem ser ensinados durante o caminhar da atividade, deste modo ela poderá ter como público-alvo os alunos que estão cursando qualquer um dos três anos do Ensino Médio. Porém, visto que ela apresenta conteúdos como: potências, logarítmos e análise do resto de divisões com números inteiros, que são assuntos tratados no durante o 1ž ano do Ensino Médio, seria de bom tom o desenvolvimento dessa atividade ao final desta etapa de ensino.

Em um primeiro momento, será abordada a dificuldade em se encontrar números primos. Nessa etapa os alunos apresentarão conhecimentos já adquiridos sobre números primos e, juntamente com o professor, irão estudar os testes de primalidades existentes, se atentando as dificuldades que esses métodos apresentam conforme aumenta-se o tamanho dos números a serem analisados. É justamente nessa etapa que o professor ficará encarregado de aproximar os alunos do conhecimento e, para tanto, deverá apresentar o conhecimento matemático que se faz necessário na utilização desses testes de primalidades.

Feito isso, os alunos utilizarão tais testes de modo a analisarem até onde conseguem ser eficientes e, em seguida, será apresentado através de vídeos uma aplicação que se utiliza da fragilidade dos testes de primalidade, a Criptografia de chave pública. Os alunos após terem contato com tal aplicação, aprenderão a utilizá-las. O professor formará pequenos grupos de alunos, os quais trocarão mensagens codificadas. É importante frisar que o professor deve ser sempre o mediador, ou seja, deve ser aquele que instiga e aproxima o aluno da busca pelo conhecimento.

4.1 Introdução

Este plano de aula terá como intuito estimular a curiosidade e o interesse dos alunos por meio de aplicações práticas dos conteúdos abordados durante esse trabalho. As atividades aqui preparadas permitirão ao aluno analisar e compreender melhor o funcionamento dos métodos criptográficos estudados bem como a importância que a matemática tem para o funcionamento deles.

Os educandos realizarão uma investigação na busca de descobrir métodos eficazes para encontrar números primos e fatorar números inteiros, para que possam compreender a dificuldade existente nesses dois itens quando estamos tratando de números grandes, dificuldade essa, que é responsável pela segurança do método Criptográfico RSA. Em um segundo momento, eles serão convidados a utilizarem os métodos criptográficos RSA e Diffie-Hellman para que possam se comunicar em segredo, contextualizando assim, o conteúdo aprendido em sala de aula. A abordagem apresentada neste trabalho tem como objetivo ser uma ponte para que o aluno se aproxime e identifique a importância da matemática aprendida na escola em aplicações práticas, promovendo uma participação mais ativa a com o ensejo de construir um conhecimento matemático com significado.

Essa sequência didática poderá proporcionar o aumento da capacidade de percepção do aluno, uma vez que motivados e convencidos da importância da matemática em seu dia a dia, poderão estudá-la com novos olhares, ampliando assim o raio de utilização dos seus conhecimentos matemáticos, uma vez que está ligada à resolução de problemas da realidade, ajudando-os a compreender melhor a importância da matemática em sua vida cotidiana, conforme concepções apresentadas por Araújo (2009). O aluno poderá adquirir uma compreensão mais ampla dos conceitos aqui discutidos, já que buscaremos focar mais no despertar da criatividade do discente, para que ele consiga utilizar o pensamento lógico matemático em um raio maior de situações.

4.2 Objetivos

A abordagem apresentada neste trabalho tem como objetivo ser uma ponte para que o aluno se aproxime e identifique a importância da matemática aprendida na escola em aplicações práticas, promovendo assim uma participação mais ativa e visando construir um conhecimento matemático com significado.

A escolha de uma aplicação presente nos dias atuais e ligada a tecnologia, tem por objetivo evidenciar a importância da matemática na sociedade e, assim, ser um motivador para o estudo dessa disciplina, promovendo uma maior dedicação, comprometimento e interesse do aluno.

O maior objetivo deste trabalho é dar ao aluno uma compreensão mais ampla dos conceitos aqui discutidos, na busca de instigá-los ao pensamento lógico, bem como despertar sua criatividade nas mais diversas situações nas quais se é possível aplicar seus conceitos.

4.3 Metodologia e Apresentação de Materiais

Por meio de uma atividade aplicada, será possível mostrar aos educandos aplicações da matemática aprendida na escola. Tal atividade tem como intenção mostrar como a matemática aliada com ideias criativas são úteis na resolução de problemas que pareciam não ter solução.

Através de uma problemática inicial o professor instigará o aluno a refletir sobre seu grau de complexidade, promovendo assim uma maior atenção e comprometimento na construção do conhecimento necessário para resolver tal problemática.

A avaliação acontecerá durante todo o projeto, com o intuito de observar a autonomia e a participação de cada educando, para que possam compreender e participar de maneira ativa das aulas. Também deverão ocorrer observações, mediações e se necessário futuras intervenções e mudanças no caminhar do projeto, visando se adequar a realidade da sala. A avaliação se dará por meio da:

- Observação do professor nas interações entre aluno-aluno e professor-aluno;
- Relatório;
- Aulas expositivas;
- Irabalho em grupo.

4.4 Roteiro detalhado da proposta

Na atividade em questão, o sócio-interacionismo terá papel de destaque, pois é interagindo com seus colegas e professor que o adolescente construirá suas conclusões necessárias a cada etapa do projeto. O construtivismo também terá sua relevância no que diz respeito à sequência de ideias e pensamentos que resultarão na solução do problema apresentado.

Percebe-se também neste projeto, que a busca pelo conhecimento será algo contínuo, construído por etapas, e que cada passo dado será de grande importância para àqueles que virão.

1. Levantamento da Problemática

   Numa primeira etapa o professor deverá apresentar uma problemática para os alunos, a qual servirá como questão norteadora de seus estudos além de ter a função de motivá-los durante os estudos.

   Tal problemática será iniciada tendo em vista a dificuldade existente até hoje em se encontrar números primos e também de se fatorar números inteiros e como matemáticos famosos souberam utilizar tal dificuldade para seu benefício, utilizando-a em uma aplicação que é utilizada cada vez com maior frequência a criptografia RSA e de Diffie-Hellman, que são responsáveis por proteger comunicações via-*internet* tornando tal meio de comunicação um ambiente seguro.

   Para isso, o professor fará uma abordagem a fim de levantar o conhecimento dos alunos sobre como encontrar números inteiros e, a partir de tal, através de uma investigação mediada pelo professor, os alunos deverão aprimorar tal método de se encontrar números primos, passando desde o crivo de Erastótenes até os mais atuais como o teste de Miller e o de Lucas, para os quais os alunos precisarão ter conhecimento sobre alguns conteúdos matemáticos como congruências e alguns teoremas, que precisarão ser trabalhados durante essa investigação.

Após a familiarização com os testes de primalidades mais atuais, será apresentada sua principal aplicação utilizada nos dias atuais, a Criptografia de chave pública. Para isso, pode ser utilizado vídeos como os listados nas referências, os quais exemplificam o funcionamentode tais métodos criptográficos, evidenciando a matemática utilizada. É importante ressaltar que o professor deve agir como mediador escolhendo momentos que julgar mais importantes, fazendo comentários que auxiliem os alunos a se aproximarem mais do tema e, assim, ajudando a fixar a ideia que está sendo utilizada.

Após a exibição dos vídeos e explicação teórica sobre o assunto, deve se abrir um espaço para um debate, no qual através de perguntas o professor tenta aproximar os alunos cada vez mais do conhecimento matemático necessário, deixando claro assim quais os conteúdos que serão aprendidos e os quais são necessário para utilizar tais técnicas Criptográficas, o que além de incentivá-los, deixará claro o objetivo a ser alcançado.

2. Pressupostos Teóricos

   Para a segunda etapa o professor deverá recordar conteúdos já estudados em anos anteriores pelos alunos sobre os números inteiros como a definição de números primos, o algoritmo da divisão, o algoritmo fatoração e o crivo de Erastóstenes. Nesta etapa, pode se lançar como desafio para os alunos alguns números ímpares grandes de até 10 algarismos para que eles tentem fatorar tais números, e caso não consigam, peça que já que não se é possível fatorá-los para que provem que tais números são primos.

   Exemplos de números a serem aplicados:

   $4051 - 6967 - 3739 - 12547 - 654369 - 1256347 - 25365351 - 231258657$

   Após terem vivenciado a dificuldade em se determinar se um número é primo ou não, o professor pode argumentar sobre a existência de testes de primalidade que surgiram ao longo da história matemática, bem como o avanço de sua eficiência em detectar se um número é primo. Após essa etapa, o professor pode introduzir o conceito de divisibilidade e de congruências, assim como os Teoremas de Fermat e Euler que servirão de ferramentas para os testes de primalidade que aprenderão a seguir.

3. Testes de Primalidade

   Nesta etapa serão apresentados alguns testes de primalidade que surgiram ao longo da história da matemática:

   - Divisão por tentativa
   - Crivo de Eratóstenes
   - Teste de primalidade de Leibniz
   - Teste de primalidade de Miller
   - Teste de primalidade de Lucas

O professor poderá abordar as vantagens e desvantagens da utilização de cada um dos testes de primalidade, bem como compará-los, evidenciando assim a evolução do pensamento humano durante a construção do conhecimento.

Além do conhecimento sobre os testes de primalidade essa etapa tem por objetivo mostrar a dificuldade em se fatorar um número e, até mesmo, de detectar se tal número é primo.

É de suma importância que o aluno compreenda tal dificuldade, pois ela será utilizada nos métodos criptográficos da próxima etapa.

4. Criptografia

   Nesta etapa os alunos se dividirão em grupos com quantidade pré-estabelecido pelo professor, e terão que se comunicar com um outro grupo em segredo, para isso utilizarão os métodos criptográficos apresentados neste trabalho.

   Em um primeiro momento será utilizada a criptografia Diffie-Hellman. Para isso, cada grupo de aluno deverá se comunicar com um outro grupo, aos pares, o professor será responsável por analisar e ver se o método esta sendo empregado corretamente. É importante aqui, que o professor auxilie os alunos na escolha da chave pública, definindo o tamanho dos números a serem utilizados, o que depende diretamente dos instrumentos que serão utilizados para realizar os cálculos (calculadora científica, calculadora do computador, *softwares* matemáticos).

   Após os alunos terem trocado mensagens utilizando a criptografia Diffie-Hellman, será a vez de utilizarem a criptografia RSA. Para isso, o professor será o destinatário das mensagens. Portanto, o professor escolherá a chave privada e criará a chave pública do método RSA, repassando somente a chave pública para os outros grupos e ficando de posse da chave privada, a qual utilizará para decodificar as mensagens enviadas a ele pelos grupos.

4.5 Considerações finais sobre a aplicação

Os alunos poderão realizar uma redação sobre as aulas apresentadas a fim de identificarem o conhecimento adquirido e expressarem suas opiniões sobre o modelo de atividade realizada. Será importante buscar destacar o quanto a criatividade acompanha o desenvolvimento da Matemática. Espera-se que o aluno aumente sua capacidade de interligar os conteúdos vistos em sala de aula a outras aplicações, conseguindo identificar situações onde possa utilizar tais conhecimentos.

Araújo(2009) nos diz que o aprendizado é algo que precisa ser interiorizado, ou seja, precisa estar numa zona proximal de conhecimento, para que assim seja possível unir o conhecimento que está descobrindo com os que já possui, fazendo sentido tal aprendizado. Desta forma, sempre que descobrimos algo ele se tornará alicerce para novas descobertas, e que o aprender faz parte do ser humano.

O uso do vídeo e das histórias deverá ajudar o aluno a entrar no mundo imaginário, a fim de vivenciar a problemática vivida ao longo do tempo, reviver etapas da construção do conhecimento e perceber como o raciocínio pode ser utilizado para resolver situações complicadas. Com isso,

construir um conhecimento mais amplo, no intuito de conseguir utilizá-lo não só em situações "programadas", mas sim, em situações reais com naturalidade e dinamismo.

Conclusão

Durante este trabalho foi possível estudar propriedades fundamentais sobre Números Inteiros, através de uma aplicação muito utilizada no dia a dia a Criptografia de chave pública, a qual só se tornou possível através de ideias criativas que utilizam tecnologia aliada a conhecimentos matemáticos.Tivemos como norte o estudo dos números primos e da Criptografia a medida que avançavamos por entre conteúdos matemáticos que tal aplicação se utiliza.

Foi possível observar ainda, de maneira lógica, o avanço da eficiência dos métodos utilizados para detectar números primos, conseguindo distinguir com eficiencia um números primos com tamanho considerável. O trabalho apresentou a evolução de alguns testes de primalidades, evidenciando as ideias e o processo de construção de cada um, até alcançar os testes de primalidades atuais: o teste de primalidade de Lucas Lehmer o qual consegue detectar com exatidão números primos até um certo tamanho, e o Teste de Miller cujo qual devida sua tamanha eficiência é utilizado por softwares computacionais nos dias atuais.

Porém, constatou-se que apesar de tal melhora nos testes de primalidades, ainda não existe um método que consiga detectar se um número qualquer é primo, pois os testes de primalidade apresentados funcionam até certo ponto, sendo que se o número a ser testado for muito grande, tal teste perde sua eficiência devido a capacidade limitada do computador, o qual pode levar anos para terminar tal teste, tornando assim o mesmo inviável.

Com o passar dos tempos, vimos que o problema matemático de se detectar a primalidade de um número, e até mesmo de se encontrar fatores primos de números inteiros teve um grande avanço, porém, quando se trata de números muito grande, esse problema ainda continua sem solução. Vimos também, que o fato desse problema continuar sem solução, passou a ser a chave para resolver um outro problema: o problema da troca de chaves, pois tal dificuldade em se detectar números primos e fatorar números Inteiros, se torna então solução para a troca de informações de maneira segura via internet, através da criptografia de chave publica (RSA e Diffie-Hellman), aplicação essa que se tornou essencial para o desenvolvimento de nossa sociedade. Desta maneira, pode-se perceber que se usada de maneira criativa, a matemática torna-se suporte para novas aplicações.

A Escrita em Códigos (Criptografia) desperta a curiosidade humana em explorar o desconhecido, motivando assim a busca pelo saber . A Criptografia presente neste trabalho faz uso de conhecimentos matemáticos importantes, assim sendo foi possível durante a abordagem aqui apresentada, estudar desde conceitos básicos como números primos, divisibilidade e fatoração até conceitos mais avançados como o estudo de grupos, congruências e testes de primalidades para números grandes. A utilização desta aplicação a qual esta presente atualmente no meio digital teve por objetivo apresentar a importância da matemática no meio em que vivemos, analisando e compreendendo seu método de funcionamento, o que consequentemente promoveu um interesse

maior pela matemática, incentivando e despertando para um aprendizado mais significativo.

No último capitulo foi apresentada uma sequência didática, visando a aplicação em sala de aula dos conteúdos, conceitos e aplicações estudados ao longo dessa dissertação para alunos do ensino médio, a fim de ilustrar uma proposta que busca despertar o raciocínio lógico dos alunos através de uma abordagem investigativa cuja qual instiga o aluno na busca pelo conhecimento.

Ao longo de todo este trabalho foi possível notar também, a evolução da matemática através dos tempos, ao estudarmos os testes de primalidade pode-se notar como o conhecimento matemático existente se torna alicerce para que um novo passo possa surgir, através da evolução de ideias e da maneira de pensar. Essa evolução do pensamento humano permite o surgimento de novas aplicações como a criptografia de chave pública, a qual se tornou indispensável no mundo financeiro atual.

Referências Bibliográficas

[1] Khan Academy. Documentário: Criptografia diffie-hellman. http://www.youtube.com/watch?v=YEBfamv-_do (acesso em 24/02/2014).

[2] Khan Academy. Documentário: Criptografia rsa. https://www.youtube.com/watch?v=wXB-V_Keiu8 (acesso em 24/02/2014).

[3] Site do *software* matemático maxima. http://MAXIMA.sourceforge.net (acesso em 13/03/2014).

[4] SPM. Documentário: Números primos. http://www.youtube.com/watch?v=FCMyFgUK5jM (acesso em 24/02/2014).

[5] SPM. Documentário: A chave das chaves. http://www.youtube.com/watch?v=Vm1Y6Kr9DCw (acesso em 24/02/2014).

[6] HEFEZ A. Elementos de aritmética. *Coleção Textos Universitários, SBM*, 2011.

[7] SANTOS B. Introdução ao software maxima. *Centro de Matemática da Universidade do Porto, CMUP*, 2009.

[8] COUTINHO S. C. Números inteiros e criptografia rsa. *Coleção Matemática e Aplicações, IMPA*, 2013.

[9] ARAÚJO J. L. Uma abordagem sócio-crítica da modelagem matemática: a perspectiva da educação matemática crítica. *ALEXANDRIA Revista de Educação em Ciência e Tecnologia*, 2(2):55–68, Julho 2009.

[10] PIRES R. M. Aplicação do algoritmo diffie-hellman no compartilhamento de volumes criptografados do truecrypt. *Departamento de Ciência da Computação, Universidade Federal de Goiás*, 2010.