Semana da Matemática

3ª Edição

Uma Introdução à Criptografia

PET|MAT

Programa de Educação Tutorial - Matemática
Universidade Federal do Paraná

Uma Introdução à Criptografia

UNIVERSIDADE FEDERAL DO PARANÁ
DEPARTAMENTO DE MATEMÁTICA
PROGRAMA DE EDUCAÇÃO TUTORIAL

Tutor:	Prof. Dr. José Carlos Corrêa Eidam
Estudantes:	Amanda Maciel de Oliveira Beatriz Borba Guergolet Bruno Mielke Schwartsburd Dyckson Ternoski Gabriel Alves de Lima Luana Bankersen Lucas Cabral Port Leonardo Gonçalves Fischer Marcel Thadeu de Abreu e Souza Marina Sayuri Vieira Mateus Balotin Matheus Daniel Galvão de Melo Matheus Kinceski Pires Monique Baptista Fragozo Nil Vinícius Gonçalves de Carvalho Priscilla Pereira de Souza Thais Spannenberg Machado dos Passos Vinicius Medeiros Prantl dos Santos
Site:	www.petmatematica.ufpr.br
Facebook:	www.facebook.com/PetMatUFPR
Instagram:	www.instagram.com/petmatematicaufpr
E-mail:	petmatufpr@gmail.com
Telefone:	(41) 3361-3672
Data do Evento:	11 a 12 de agosto de 2020

Curitiba, agosto de 2020.

Apresentação

Prezado Estudante!

É uma grande alegria tê-lo conosco em mais uma edição da Semana da Matemática da UFPR (SMAT)! Nossa intenção é que este minicurso possa ajudá-lo a alargar suas fronteiras e ampliar seu horizonte na Matemática.

Devido à pandemia que enfrentamos, esta edição da SMAT está acontecendo de forma inteiramente virtual, o que se constitui em um grande desafio para todos nós. Temos certeza que este novo formato alterará nossas percepções sobre o ensino e a aprendizagem e terá muitas consequências importantes em um futuro próximo.

Quero registrar meus agradecimentos e reconhecimento a cada estudante do PET Matemática pelo empenho e dedicação em cada detalhe deste importante trabalho em prol dos alunos do Curso de Matemática e de toda a UFPR.

Prof. José Carlos Eidam
Tutor do PET-Matemática

Sumário

Apresentação **3**

1 Introdução à criptografia **7**
1.1 O que é Criptografia? 7
1.2 História geral da Criptografia 8
1.3 Código de César 9
1.4 Como decifrar o código? 9
1.4.1 Análise de Frequência 9
1.5 Código em Blocos 13
1.6 Chave Privada x Chave Pública 14
1.7 Criptografia em Matrizes 16
1.7.1 Criptografando a mensagem 17
1.7.2 Descriptografando a mensagem 19
1.8 Exercícios 20

2 Introdução à Teoria dos Números **23**
2.1 Critérios de divisibilidade 23
2.2 Números primos 25
2.3 Fatorando números 30
2.4 Relações de equivalência 33
2.5 Congruência modular 36
2.6 Aritmética modular 41
2.6.1 Soma de classes 41

2.6.2 Multiplicação de classes 42
2.6.3 Elemento simétrico 42
2.6.4 Propriedades de equivalência 43
2.7 Divisão Modular 43
2.8 Potências 46
2.9 Função ϕ de Euler 50
2.10 Teoremas de Fermat e de Euler 52
2.11 Exercícios 56

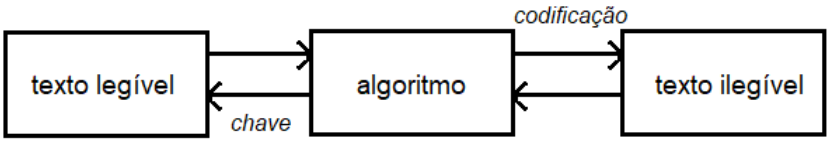
3 Criptografia RSA **59**
3.1 Motivação 59
3.2 Diferenças entre chave pública e chave privada 60
3.3 Criptografia RSA: funcionamento 63
3.3.1 Exercícios 69
3.4 Por que funciona? 70
3.5 A Implementação 70
3.6 Problemas de segurança 71
3.7 Testes de Primalidade 72
3.8 Assinatura digital 74
3.9 Aplicação 76

Capítulo 1

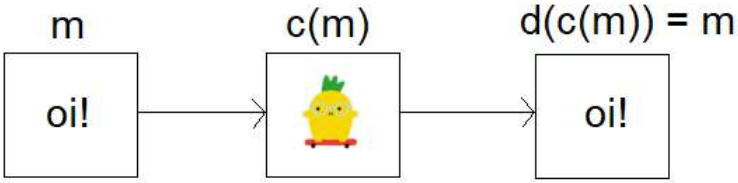
Introdução à criptografia

1.1 O que é Criptografia?

A criptografia é o estudo de métodos e técnicas para transformar um texto legível em algo ilegível, podendo reverter o processo e obter o texto original.



Desse modo, se Bianca quer enviar uma mensagem **m** para João, inicialmente eles combinam como será a "receita" de criptação e decodificação, ou seja, as funções **c** e **d**, respectivamente.



No exemplo acima, a mensagem **m** é *oi!*, a mensagem cifrada **c(m)** é um abacaxi, que decifrada volta a mensagem original: **d(c(m)) = m.**

Interpretando dessa maneira, parece que a criptografia só vale para mensagens escritas, o que não é verdade.

Durante as guerras mundiais e na guerra fria, o *Código Morse* era utilizado; este sistema consiste na conversão de letras em pontos e traços, e posteriormente em sons, de modo a transmitir uma mensagem.

1.2 História geral da Criptografia

A criptografia é tão antiga quanto a escrita e tem um papel muito importante na história e vivência da humanidade. Desde desfechos de guerras à coisas triviais, como mensagens de celular. É a ferramenta que garante a segurança na comunicação.

Vindo do grego, 'KRIPTÓS' significa *escondido* e 'GRÁPHO' significa *escrita.*

Seu objetivo é ocultar o conteúdo de uma mensagem de modo que, caso haja uma tentativa de intercepção, esteja ilegível ou não seja entendível de imediato.

Um dos primeiros registros da criptografia de 1900 a.C., quando um escriba substituiu algumas palavras de um *hieróglifo* (sistema de escrita formal do Egito Antigo) com o intuito de impedir que um futuro ladrão encontrasse o caminho para o tesouro de uma pirâmide.

1.3 Código de César

As técnicas de substituição ou transposição de letras são chamadas de *cifras*.

Em 50 a.C. na cidade de Roma, Júlio César utilizou uma cifra de substituição para proteger comunicações governamentais. Atualmente, este método é conhecido como Cifra de César (ou Código de César).

César alternava seus textos de modo que cada letra fosse substituída por outra três posições a frente no alfabeto. Assim, A virava D, B virava E, e assim por diante, como mostra a tabela abaixo.

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**
D	E	F	G	H	I	J	K	L	M	N	O	P
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Então a mensagem MATEMATICA torna-se PDWHPDWLFD.

Mesmo sendo simples (ou exatamente por isso), a cifra de César foi utilizada por muito tempo, e juntamente com alguns truques, permaneceu indecifrável por séculos.

1.4 Como decifrar o código?

1.4.1 Análise de Frequência

Como diversos conflitos eram decididos pela criptografia, por volta dos anos 600 os árabes criaram a criptoanálise, o estudo para decodificar mensagens.

No século IX, o cientista árabe Al-Kindi descreveu a técnica de estudar a frequência das letras em um texto com o intuito de quebrar códigos. Segue abaixo este método:

1. Pegue um texto qualquer
2. Tabele a frequência de cada letra nele
3. Tabele a frequência de cada letra do texto codificado
4. Associe as letras de mesma frequência em 2) e 3)

Para facilitar, os passos 1 e 2 estão feitos na tabela abaixo, que apresenta a porcentagem de utilização das letras no português brasileiro:

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
14.63	1.04	3.88	4.99	12.57	1.02	1.30	1.28	6.18
J	**K**	**L**	**M**	**N**	**O**	**P**	**Q**	**R**
0.40	0.02	2.78	4.74	5.05	10.73	2.52	1.20	6.53
S	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**	
1.81	4.74	4.63	1.67	0.01	0.21	0.01	0.47	

Essa técnica decodifica as cifras de substituição. Para entender melhor a contagem de frequência, veja o exemplo abaixo.

Exemplo 1.4.1. Vamos decifrar esta mensagem usando análise de frequência:

ILT CPUKV HV IYPUJHUKV KL THALTHAPJV
KLZAH CLG V ALTH L JYPWAVNYHMPH
LZWLYHTVZ XNL CVJL NVZAL

Contando a quantidade de letras, temos 93 caracteres. A frequência de cada letra é dada pela tabela abaixo.

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**
6	0	3	0	0	0	1	10	2	2	3	13	1
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
2	0	5	0	0	0	5	3	9	2	1	4	5

As três letras de maior frequência são L, H e V, então espera-se que elas sejam A, E ou O. Veja que a terceira palavra é HV, então ela pode ser "ae", "ao", "ea",... A única combinação que faz sentido é ao, então pode ser que H = a, V = o e L = e.

IeT CPUKo ao IYPUJHUKo Ke TaAeTaAPJa
KeZZa CeG o AeTa e JYPWAoNYaMPa
eZWeYaToZ XNe CoJe NoZAe

São poucas as palavras de 2 letras no português, como Ke parece conectar as palavras, ela pode significar "de" ou "se".

Se K = d, a palavra *KeZZa* vira *deZZa*, que lembra muito "dessa". Po outro lado, se K = s, não existe uma palavra que satisfaz seZZa. Portanto K = d e Z = s.

IeT CPUdo ao IYPUJHUdo de TaAeTaAPJo
dessa CeG o AeTa e JYPWAoNYaMPa
esWeYaTos XBe CoJe NosAe

No português, a letra "i" tem frequência de 6%, portanto as letras mais próximas são P e T. Se T = i, a primeira palavra terminaria com "ei", o que é quase impossível para uma palavra de 3 letras. Então P = i.

Veja também que A e T estão na faixa de 4 a 6%, podendo ser m, n, r ou t. É razoável que T seja "m" ou "r", porque *IeT* termina com T. E a palavra *NosAe* não permite que A seja r.

Com essas observações, *AeTa* pode formar as palavras "tema" ou "tera". Iremos escolher a primeira opção, mas caso encontremos algum erro por conta dessa escolha, retornaremos a esta etapa para trocar a palava.

Iem CiUdo ao IYiUJHUdo de matematiJo
dessa CeG o tema e JYiWtoNYaMia
esWeYamos XBe CoJe Noste

É fácil ver que J = c, e por restrição de possibilidade, *XBe* significa "que" e N = g.

Iem CiUdo ao IYiUcHUdo de matematico
dessa CeG o tema e cYiWtogYaMia
esWeYamos que Coce goste

É perceptível que C = v, U = n, G = z e Y = r.

Iem vindo ao IrincHndo de matematico
dessa vez o tema e criWtograMia
esWeramos que voce goste

Então a mensagem fica:

Bem vindo ao brincando de matemático,
dessa vez o tema é criptografia.
Esperamos que você goste.

Para decifrar essas mensagens, é recomendado que primeiro identifique quais letras correspondem as vogais A, E e O. Depois é necessário estudar as palavras curtas e suas combinações para intuitivamente chegarmos a opção correta.

Sabendo a análise de frequência, as cifras de substituição não são mais eficientes.

Será que existe uma maneira na qual seja inviável essa contagem de frequência? Sim! Existem os chamados *códigos em bloco* que veremos a seguir.

1.5 Código em Blocos

Esse método consiste em dividirmos a mensagem em blocos e embaralharmos suas letras. Daí o nome *código em bloco* para este processo. Vejamos como ocorre em um exemplo.

Iremos criptografar a mensagem AMO O BRINCANDO DE MATEMÁTICO. Para isso, seguiremos a receita abaixo:

RECEITA DE BOLO:

Ingredientes: mensagem a ser criptografada

Modo de preparo:

- Retire os espaços entre as palavras e adicione um A no final caso haja um número ímpar de letras
- Divida a frase em blocos de duas letras.
- Em cada bloco, permute as letras de lugar.
- Troque as posições dos blocos "ímpares" da seguinte forma: - primeiro com o último

  - terceiro com o antepenúltimo

  $\vdots$

  e assim por diante, deixando os blocos "pares" parados.

Bolo: mensagem criptografada

Seguindo o passo a passo da nossa receita, obtemos no primeiro passo:

AMOOBRINCANDODEMATEMATICOA

Dividindo em blocos,

AM OO BR IN CA ND OD EM AT EM AT IC OA

Permutando as letras,

MA OO RB NI AC DN DO ME TA ME TA CI AO

Trocando as posições dos blocos de lugar

AO OO TA NI TA DN DO ME AC ME RB CI MA

Justapondo os blocos novamente, temos a seguinte mensagem criptografada:

AOOOTANITADNDOMEACMERBCIMA

O código em bloco é um exemplo de criptografia de chave privada.

Mas o que é chave privada?

1.6 Chave Privada x Chave Pública

A criptografia de chave privada, também chamada criptografia simétrica, utiliza apenas uma chave. A mensagem é criptografada com essa chave pelo emissor, e descriptografada com a mesma, pelo receptor. Essa ação é ilustrada abaixo:

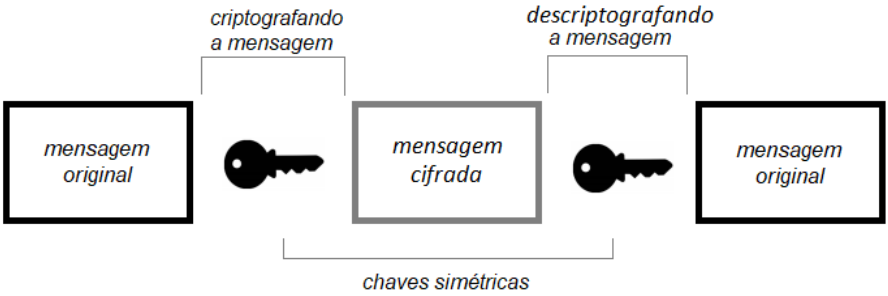


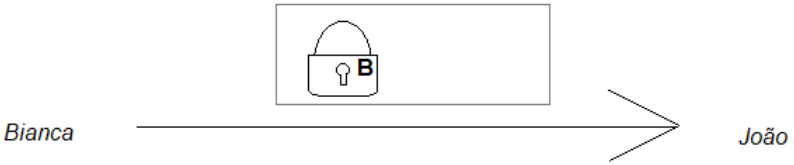
Figura 1.1: Ilustração chave privada

Note que durante o processo de envio da mensagem, um terceiro pode interceptá-la e descobrir a chave, já que é apenas uma.

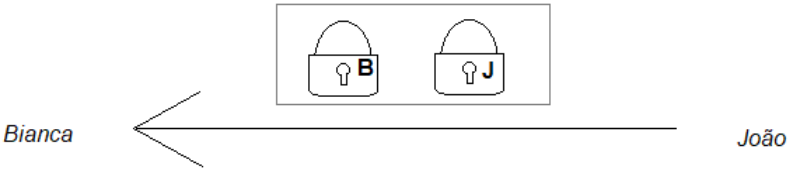
Para isso, temos a criptografia de chave pública, ou chave assimétrica, que utiliza duas ou mais chaves. Vejamos um exemplo abaixo no qual utilizamos duas chaves.

Digamos que Bianca e João desejam se comunicar secretamente. Para tal, ela possui um cadeado **B**, e para abri-lo, uma chave **b**. Analogamente, João possui um cadeado **J** e uma chave **j**.

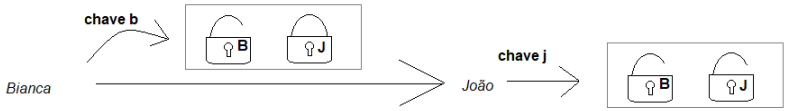
Bianca envia uma caixa com sua mensagem, fechada com o cadeado **B** para João.



Ao recebê-la, João coloca seu cadeado **J** e devolve para Bianca com os dois cadeados, **J** e **B**.



Bianca utiliza sua chave **b** para abrir seu cadeado e devolve para João, agora apenas com o cadeado **J**, cuja chave ele tem.



Assim, finalmente poderá abrir a caixa e ler a mensagem.

O mais conhecido dos métodos de criptografia de chave pública é o RSA, inventado em 1977 por R. L. Rivest, A. Shamir e L. Adleman, cujas iniciais correspondem a sigla RSA.

1.7 Criptografia em Matrizes

Um outro método de criptografia é via matrizes. Neste caso, a nossa *chave* será uma *matriz quadrada* invertível, para garantir que a mensagem possa ser descodificada.

O processo consiste em converter cada letra em um número, e separá-los em vetores coluna. Daí, para criptografar a mensagem, faremos o produto da matriz chave por cada vetor, afim de encontrar um novo vetor coluna. Por fim, converteremos novamente cada número na letra correspondente na tabela de conversão, e encontraremos a mensagem criptografada.

Vejamos um exemplo com a mensagem:

AMO MATEMÁTICA

Para tal, usaremos a seguinte tabela de conversão:

A	B	C	D	E	F	G	H	I	J	K	L	M
1	2	3	4	5	6	7	8	9	10	11	12	13
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	26

1.7.1 Criptografando a mensagem

Cada letra será associada ao número de sua posição no alfabeto;

AMO MATEMATICA
A M O M A T E M A T I C A
1 13 15 13 1 20 5 13 1 20 9 3 1

Agruparemos os números dois a dois, e colocaremos em vetores coluna 2×1. Como juntaremos em duplas e temos um número ímpar de números/letras, adicionamos uma letra no final que não altera o sentido da mensagem. Neste caso adicionaremos mais uma letra "A", que corresponde ao número 1.

$$\underbrace{\begin{pmatrix} 1 \\ 13 \end{pmatrix}}_{v_1} \underbrace{\begin{pmatrix} 15 \\ 13 \end{pmatrix}}_{v_2} \underbrace{\begin{pmatrix} 1 \\ 20 \end{pmatrix}}_{v_3} \underbrace{\begin{pmatrix} 5 \\ 13 \end{pmatrix}}_{v_4} \underbrace{\begin{pmatrix} 1 \\ 20 \end{pmatrix}}_{v_5} \underbrace{\begin{pmatrix} 9 \\ 3 \end{pmatrix}}_{v_6} \underbrace{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}_{v_7}$$

Em seguida, faremos o processo de codificação. Primeiramente escolhemos uma matriz 2×2 que seja invertível.

A matriz $C = \begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}$ possui $det(C) = 3 \cdot 1 - 2 \cdot 2 = -1 \neq 0$, logo possui inversa, então essa é uma possível chave, já que é invertível. Assim, vamos escolhê-la como nossa chave.

Tendo escolhida nossa chave, multiplicaremos a mesma por cada vetor coluna, resultando em:

$$C \cdot v_1 = \begin{pmatrix} 27 \\ 41 \end{pmatrix} C \cdot v_2 = \begin{pmatrix} 41 \\ 69 \end{pmatrix} C \cdot v_3 = \begin{pmatrix} 41 \\ 62 \end{pmatrix} C \cdot v_4 = \begin{pmatrix} 31 \\ 49 \end{pmatrix}$$

$$C \cdot v_5 = \begin{pmatrix} 41 \\ 62 \end{pmatrix} C \cdot v_6 = \begin{pmatrix} 15 \\ 27 \end{pmatrix} C \cdot v_7 = \begin{pmatrix} 3 \\ 5 \end{pmatrix}$$

Agora voltaremos a tabela de conversão para determinar as letras que correspondem a cada número, encontrando assim nossa mensagem criptografada. Note que nem todos os números tem correspondentes na tabela. Os números maiores que 26 correspondem a letra após ter percorrido o alfabeto mais de uma vez. Por exemplo o número $41 = 26 + 15$ percorreu o alfabeto uma vez e parou no 15, que corresponde a letra O.

- $27 = 26 + \mathbf{1} = A$
- $41 = 26 + \mathbf{15} = O$
- $41 = 26 + \mathbf{15} = O$
- $69 = 26 \cdot 2 + \mathbf{17} = Q$
- $41 = 26 + \mathbf{15} = O$
- $62 = 26 \cdot 2 + \mathbf{10} = J$
- $31 = 26 + \mathbf{5} = E$
- $49 = 26 + \mathbf{23} = W$
- $41 = 26 + \mathbf{15} = O$
- $62 = 26 \cdot 2 + \mathbf{10} = J$
- $\mathbf{15} = O$
- $27 = 26 + \mathbf{1} = A$
- $\mathbf{3} = C$
- $\mathbf{5} = E$

Assim a mensagem é: AOOQOJEWOJOACE.

1.7.2 Descriptografando a mensagem

Inicialmente transformaremos nossa mensagem criptografada em números de acordo com a tabela de conversão. Para voltar a mensagem original, utilizaremos a inversa da matriz chave.

A matriz inversa de $C = \begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}$ será:

$$C^{-1} = \frac{1}{(3 \cdot 1) - (2 \cdot 2)} \cdot \begin{pmatrix} 3 & -2 \\ -2 & 1 \end{pmatrix} = \begin{pmatrix} -3 & 2 \\ 2 & -1 \end{pmatrix}$$

Em seguida, multiplicaremos a matriz inversa por cada vetor codificado, encontrando os vetores originais.

$$C^{-1} \cdot \begin{pmatrix} 27 \\ 41 \end{pmatrix} = \begin{pmatrix} 1 \\ 13 \end{pmatrix} \qquad C^{-1} \cdot \begin{pmatrix} 41 \\ 62 \end{pmatrix} = \begin{pmatrix} 1 \\ 20 \end{pmatrix}$$

$$C^{-1} \cdot \begin{pmatrix} 41 \\ 69 \end{pmatrix} = \begin{pmatrix} 15 \\ 13 \end{pmatrix} \qquad C^{-1} \cdot \begin{pmatrix} 15 \\ 27 \end{pmatrix} = \begin{pmatrix} 9 \\ 3 \end{pmatrix}$$

$$C^{-1} \cdot \begin{pmatrix} 41 \\ 62 \end{pmatrix} = \begin{pmatrix} 1 \\ 20 \end{pmatrix} \qquad C^{-1} \cdot \begin{pmatrix} 3 \\ 5 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$C^{-1} \cdot \begin{pmatrix} 31 \\ 49 \end{pmatrix} = \begin{pmatrix} 5 \\ 13 \end{pmatrix}$$

Assim, voltando a tabela, encontramos as letras da codificação, voltando a mensagem original.

1 13 15 13 1 20 5 13 1 20 9 13 1 1
A M O M A T E M A T I C A A

Agora vamos colocar em prática o que aprendemos nesse capítulo!

1.8 Exercícios

Exercício 1.8.1. Cifre o seu nome completo.

Exercício 1.8.2. Quantas chaves podemos fazer com a cifra de substituição?

Exercício 1.8.3. Use a cifra de substituição com -8 posições para cifrar a mensagem: "O Naruto pode ser um pouco duro às vezes".

Exercício 1.8.4. Uma cifra que substitui 365 posições seria equivalente a substituir quantas posições? E para um caso de n posições, sendo n um número qualquer?

Exercício 1.8.5. Use a cifra de César para criptografar a mensagem "Aprender Matemática fica mais fácil quando gostamos dela".

Exercício 1.8.6. Utilizando a análise de frequência, decifre a seguinte mensagem:

WKQSK O WKDOWKDSMK CKY ZKBOMSNKC
ECKWYC Y QSJ MYWY FKBSXRK O MYBKMKY
MOW K BKJKY XYC QESK ZKBK NOCFOXNKB EW
XYFY WSDOBSY

Exercício 1.8.7. Utilizando o algoritmo de criptografia em blocos, decifre a mensagem abaixo:

AHIUSLIVOANEANASHECIQA

Exercício 1.8.8. Verifique se as matrizes abaixo poderiam ser utilizadas como chave para codificar mensagens:

a) $A = \begin{pmatrix} -1 & 3 \\ 2 & 2 \end{pmatrix}$ **c)** $C = \begin{pmatrix} 3 & 6 \\ 1 & 2 \end{pmatrix}$

b) $B = \begin{pmatrix} -3 & 5 \\ -1 & 2 \end{pmatrix}$ **d)** $D = \begin{pmatrix} 8 & 4 \\ 2 & 1 \end{pmatrix}$

Exercício 1.8.9. Crie uma tabela de conversão e com as chaves encontradas no item anterior, codifique a mensagem EU GOSTO DE PASTEL DE FLANGO.

Exercício 1.8.10. Utilizando a chave $C = \begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix}$ e a tabela de conversão apresentada no capítulo, codifique a mensagem FIBONACCI.

Exercício 1.8.11. Utilizando a chave $C = \begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix}$ e a tabela de conversão apresentada no capítulo, codifique a mensagem CINCO EH PRIMO.

Exercício 1.8.12. Decodifique a mensagem

B U E T R Y Z D U J J Q

na qual a chave utilizada para codificar foi $C = \begin{pmatrix} 3 & 5 \\ 1 & 2 \end{pmatrix}$

Capítulo 2

Introdução à Teoria dos Números

Neste capítulo iremos discutir conceitos básicos de teoria de números, estes que serão necessários para entender a criptografia RSA (o algorítmo que será abordado no último capítulo). Desta forma o ritmo será um pouco diferente; recomendamos a leitura completa aos estudantes que não fizeram um curso básico de teoria de números e, para aqueles que já conhecem os conteúdos, o capítulo servirá como uma boa fonte de revisão.

2.1 Critérios de divisibilidade

Você já deve ter se deparado com alguma situação que te fez pensar, "será que esse número é divisível por tal número?" E, embora não existam métodos para todos os números, há critérios para sabermos se um número divide outro. Vejamos alguns desses critérios para números não tão grandes.

Critério de divisibilidade por 2

Um número é divisível por 2 se for par, isto é, se o alga-

rismo da unidade for $0, 2, 4, 6$ ou 8. Por exemplo, 35468441546 é divisível por 2, pois termina em 6.

Critério de divisibilidade por 3

Um número é divisível por 3 caso a soma de todos os seus algarismos for divisível por 3. Por exemplo 848942312544 é divisível por 3, pois $8 + 4 + 8 + 9 + 4 + 2 + 3 + 1 + 2 + 5 + 4 + 4 = 54$ e 54 é divisível por 3. Caso a soma dos algarismos resultar em um número muito grande, podemos usar o critério novamente. Por exemplo, 54 é divisível por 3, pois $5 + 4 = 9$ que é divisível por 3. O critério pode ser aplicado quantas vezes forem necessárias.

Critério de divisibilidade por 4

Um número é divisível por 4 caso termine em 00, ou algum múltiplo de 4. Por exemplo, 4149874148516 é divisível por 4, pois seus dois últimos algarismos são 16, e 16 é múltiplo de 4.

Critério de divisibilidade por 5

Um número é divisível por 5, se o último algarismo deste número for 0 ou 5. Por exemplo, 41416548945 é múltiplo de 5, pois termina em 5.

Critério de divisibilidade por 7

Para descobrir se um número é múltiplo de 7, faça o seguinte procedimento:

Multiplique por 2 o último algarismo do número. Subtraia este valor do número inicial sem o último algarismo, o resultado deve ser múltiplo de 7.

Você pode repetir este procedimento quantas vezes forem necessárias, até encontrar um número que consiga dizer se é múltiplo de 7 ou não. Vamos tomar como exemplo o número 7203:

- Último algarismo: 3;
- Multiplicar o último algarismo por 2: $2 \cdot 3 = 6$;

- Subtrair este resultado pelo número inicial sem este algarismo: $720 - 6 = 714$;
- Número atual: 714, ainda não sabemos se este número é ou não divisível por 7, então fazemos o processo de novo;
- Último algarismo: 4;
- Multiplicar o último algarismo por 2: $2 \cdot 4 = 8$;
- Subtrair este resultado pelo número inicial sem este algarismo: $71 - 8 = 63$;

Como já conseguimos ver que 63 é múltiplo de 7, vemos que 7 divide 7203.

Critério de divisibilidade por 8

Um número é divisível por 8 caso os últimos três algarismos sejam 000 ou algum múltiplo de 8. Por exemplo, 61112 é divisível por 8, pois 112 é divisível por 8. 84591000 é múltipo de 8 pois seus últimos três algarismos são 000.

Critério de divisibilidade por 9

Um número é divisível por 9 caso a soma de seus algarismos seja múltiplo de 9. Por exemplo, 16874964 é divisível por 9, pois $1+6+8+7+4+9+6+4 = 45$, e 45 é divisível por 9.

Já para outros números primos, mais detalhes sobre os critérios e demonstrações de divisibilidade podem ser encontrados em [**6**].

2.2 Números primos

Na matemática, há uma infinidade de números diferentes, com propriedades diferentes a serem estudadas. Existem

números fascinantes que possuem diversas aplicações, uma delas na criptografia, que são os *números primos*.

Definição 2.2.1. (Número primo) Um número inteiro p é dito um *número primo* se, e somente se, dentro do conjunto dos números inteiros, só pode ser dividido por 1, -1, p e $-p$.

Definição 2.2.2. (Números coprimos) Dois números são ditos *coprimos* se eles não possuem divisores em comum.

Observação: Os números primos são todos coprimos entre si.

Os matemáticos sempre quiseram encontrar fórmulas que resultassem em números primos. Algumas que se destacaram foram as exponenciais. Há duas fórmulas exponenciais de enorme importância histórica, ambas intensamente estudadas pelos matemáticos do século $XVII$, sobretudo Fermat. Uma dessas gera os conhecidos números de Mersenne.

Definição 2.2.3. (Números de Mersenne) Um número natural é dito um *número de Mersenne* caso satisfaça a fórmula:

$$M(n) = 2^n - 1,$$

onde n é um número natural.

Dentro do conjunto dos números de Mersenne, existem os chamados **primos de Mersenne**, que são números de Mersenne que também são primos. É importante ressaltar que nem todo número de Mersenne é um número primo.

Os números de Mersenne são assim chamados por conta de uma afirmação do matemático Marin Mersenne, no século XVII, que afirmou que os números de Mersenne seriam primos quando $n = 2, 3, 5, 7, 13, 17, 19, 31, 67, 127$ e 257; e compostos para os outros 44 valores primos de n menores que 257. Apesar disso, Mersenne não apresentou nenhuma justificativa ou demonstração para essa afirmação. Tanto que esta afirmação é na verdade falsa, pois, $M(61)$, $M(89)$ e $M(107)$

são primos de Mersenne, e não foram listados por Mersenne. Além disso, os números $M(67)$ e $M(257)$ na verdade não são números primos.

Uma propriedade muito interessante dos números de Mersenne é que é possível provar que, na fórmula dos números de Mersenne $M(n) = 2^n - 1$, se n não for primo, então $M(n)$ também não será primo, e se $M(n)$ é primo, então n também é primo. No entanto, o raciocínio contrário não necessariamente é verdadeiro, isto é, se n for primo, não significa que $M(n)$ também será primo.

Uma curiosidade é que atualmente se conhecem 51 primos de Mersenne, o último a ser descoberto foi $2^{82.589.933} - 1$, um número com mais de 24 milhões de algarismos, foi descoberto em dezembro de 2018. Não sabemos se existem infinitos primos de Mersenne ou não, este ainda é um problema em aberto na Matemática.

Uma outra fórmula exponencial importante é a fórmula dos números de Fermat.

Definição 2.2.4. (Números de Fermat) Um número natural é dito um *número de Fermat* caso satisfaça a fórmula:

$$F(n) = 2^{2^n} + 1,$$

Em uma carta no ano de 1640, Fermat enumerou os números da forma $F(n) = 2^{2^n} + 1$, para os valores inteiros de n entre 0 e 6 os números são:

$$3, 5, 17, 257, 65537, 4294967297 \quad \text{e} \quad 18446744073709551617$$

Em seguida, afirmou que todos os números desta forma são primos. Porém ele estava errado, pois o último número desses listados não é primo. Até hoje não se conhece nenhum primo de Fermat maior que $F(5)$.

Uma propriedade muito importante sobre os números inteiros que podemos concluir é a seguinte:

Teorema 2.2.5. Todo número inteiro, ou é um número primo, ou pode ser escrito como um produto de números primos, sendo essa escrita única.

Ou seja, isso quer dizer que todo número inteiro possui uma **única** fatoração em números primos. Essa propriedade é tão importante, que é conhecida pelo nome **Teorema Fundamental da Aritmética**. Como é um teorema tão importante, iremos prová-lo.

Demonstração. Faremos a demonstração em duas etapas: primeiro provaremos que existe alguma decomposição em primos para qualquer número natural, e, em seguida, provaremos que tal decomposição é única em todos os casos.

Existe alguma decomposição em primos

Vamos demonstrar que, para qualquer número inteiro maior que 1, ele pode ser decomposto de alguma maneira em algum produto de primos. Vamos começar com $x = 2$, podemos ver que 2 admite uma decomposição em número primos, já que ele próprio é um número primo.

Agora, considerando que $x > 2$, temos o seguinte caso: Se x for primo, então ele pode ser escrito como produto de primos, pois ele próprio é um número primo. Se x não for um número primo, então ele possui divisores dentro dos números inteiros, logo, x pode ser escrito da seguinte forma: $x = d \cdot q$, onde d e q são outros números inteiros que dividem x. Se d e q são números primos, então, x foi escrito como um produto de primos, caso d ou q não sejam primos, então é possível os escrever como um outro produto de dois números naturais. Por exemplo, se $d = r \cdot s$ e $q = v \cdot t$, então $x = r \cdot s \cdot v \cdot t$ e daí analisaremos se r,s,t,v são números primos, ou podem ser escrito como um produto de outros dois números. Po-

demos repetir o processo sucessivamente após cada etapa, até que não seja mais possível, restando apenas primos na decomposição de x. Este raciocínio nos permite ver que, de fato, dado qualquer número inteiro x, existe alguma decomposição de x em um produto de números primos. No entanto, apenas este raciocínio não demonstra que só existe uma decomposição para x.

Tal decomposição em primos é única:

Já que provamos que existe uma decomposição em primos para qualquer número inteiro, resta provar que existe uma única decomposição. Para provar, usaremos prova por indução, que consiste em primeiro provar que determinada propriedade vale para $k = 1$, e depois provar que se assumirmos que tal propriedade vale até um passo k, então a propriedade vale também para $k+1$, e se conseguirmos provar tudo isso, então a propriedade vale para todo o conjunto dos números inteiros.

Vamos supor que um número inteiro x admite duas decomposições diferentes em primos, isto é $x = p_1$, com p_1 primo, e também $x = q_1 \cdot q_2 \ldots q_s$, onde q_n e q_m podem ou não ter o mesmo valor, por exemplo $3{\cdot}3{\cdot}5{\cdot}7{=}q_1{\cdot}q_2{\cdot}q_3{\cdot}q_4$, com $q_1 = q_2 = 3$. Fazemos isso por causa do fato que um mesmo primo pode aparecer várias vezes na decomposição de um mesmo número. Se não existe mais de um q_n assumindo repetidamente o valor 1, então chamaremos s de comprimento de x. Como $x = p_1$ e $x = q_1 \cdot q_2 \ldots q_s$,então $p_1 = q_1 \cdot q_2 \ldots q_s$. Como q_1 divide $q_1 \cdot q_2 \ldots q_s$, então q_1 divide p_1. Mas como p_1 divide q_1, se q_1 também divide p_1, então $q_1 = p_1$. Voltando na igualdade $p_1 = q_1 \cdot q_2 \ldots q_s$ e aplicando que $q_1 = p_1$, dividimos por q_1 de ambos os lados, e concluímos que: $1 = q_2 \ldots q_s$ logo, $s = 1$. Assim, todo número de comprimento 1, admite uma única decomposição. Provamos que vale para 1, resta provar que, se vale para k, vale para $k + 1$ também.

Agora, suponhamos que para um número natural k, todos os números de comprimento k possuem única decomposição. Seja $x = p_1 \cdot p_2 \dots p_{k+1} = q_1 \cdot q_2 \dots q_s$ onde, se $n < m$, então $q_n \leq q_m$, ou seja, os fatores estão em ordem crescente, podendo um mesmo valor se repetir. Vamos provar que números de comprimento $k+1$ admitem uma única decomposição. Temos que q_1 divide $p_1 \cdot p_2 \dots p_{k+1}$, então q_1 divide algum p_i. No entanto, também é verdade que p_1 deve dividir algum q_j, façamos, então, tal divisão dos dois lados. $p_1 \cdot p_2 \dots p_{i-1} \cdot p_{i+1} \dots p_{k+1} = q_1 \cdot q_2 \cdot q_{j-1} \cdot q_{j+1} \dots q_s$. No entanto, podemos ver que, agora, $p_1 \cdot p_2 \dots p_{i-1} \cdot p_{i+1} \dots p_{k+1}$ é um número de comprimento k, e pela hipótese de indução, todos os valores de comprimento k, possuem uma única decomposição.

Concluímos, assim, a prova de que, dado qualquer número inteiro, tal número admite uma única decomposição em números primos. É importante ressaltar que, nesta demonstração, tomamos apenas números naturais no decorrer da prova, mas a prova é análoga para todo o conjunto dos números inteiros, pois para provar que vale para os números negativos também, o raciocínio é inteiramente análogo. □

Nós usamos este resultado para fatorarmos todos os números e, uma vez encontrada essa fatoração, ela é única. Por exemplo, o número 4620 pode ser escrito da forma $2^2 \cdot 3 \cdot 5 \cdot 7 \cdot 11$.

2.3 Fatorando números

Agora que vimos que, para qualquer número inteiro, sempre haverá uma única fatoração em números primos, nos interessa sabermos como fatorar esses números. Há diversos métodos usados para fatorar números, alguns computacio-

nais e outros manuais. O método mais utilizado para fatorar números manualmente é dividir o número por números primos até que reste 1. É um processo similar ao do *mmc*. Para fazermos isso, podemos usar os critérios de divisibilidade que vimos anteriormente. Por exemplo, vamos fatorar 12012:

Vamos dividir o número por um primo até que este número não seja mais divisível por este primo, e então partimos para o próximo primo

- Número:12012;
- Usando o critério de divisibilidade por 2, vemos que é divisível por 2, então dividimos por 2, resultando em 6006;
- 6006 é também divisível por 2, então dividimos por 2 de novo, resultando em 3003;
- 3003 não é mais divisível por 2, então partimos para o próximo primo, no caso, 3.;
- 3003 é divisível por 3? Sim, pois pelo critério de divisibilidade por 3, vamos que $3 + 0 + 0 + 3 = 6$, logo, é divisível por 3, então dividimos por 3 que resulta em 1001;
- Aplicando o critério de divisibilidade por 7, fazemos $100 - 2 = 98$. Como 98 é divisível por 7, 1001 também o é, logo, dividindo 1001 por 7 chegamos em 143;
- 143 não é mais divisível por 7, então vamos para o próximo primo, 11;
- 143 não é um número tão grande, então não há necessidade de aplicarmos o critério de divisibilidade para verificarmos se ele é divisível por 11. É fácil ver que $143 = 11 \cdot 13$ logo, dividimos por 11, restando 13;

- 13 é divisível por 13 e um primo, logo, dividimos 13 por 13 e chegamos em 1, terminando, assim, a nossa fatoração;

- Do mesmo modo, podemos verificar que a forma fatorada de 84084 é $3531528 = 2^2.3.7^2.11.13$.

Nesta fatoração, as divisões que fizemos, foram, respectivamente: $2, 2, 3, 7, 7, 11, 13$. Portanto, a forma fatorada é $12012 = 2^2 \cdot 3 \cdot 7^2 \cdot 11 \cdot 13$.

Outro teorema muito interessante e importante relacionada à numeros primos, é o seguinte:

Teorema 2.3.1. (Teorema de Euclides): Existem infinitos números primos.

Demonstração. Demonstraremos o teorema mostrando que, caso ele fosse falso, isso geraria uma contradição e, portanto, ele é verdadeiro. Vamos supor, por absurdo, que exista uma quantidade finita de números primos. Se há uma quantidade finita de primos, então, é possível os listar dentro de um conjunto finito, por exemplo, seja L o conjunto $L = (p_1, p_2, \ldots, p_n)$ de todos os números primos. Se considerarmos o produto de todos os números dessa lista, isto é, seja $p = p_1 \cdot p_2 \ldots p_n$ o produto de todos os números primos de L, então, podemos notar que se considerarmos o número $p + 1$, este número não será divisível por nenhum primo do conjunto L pois, p e $p+1$ são coprimos entre si e, sendo coprimos, não podem possuir divisores em comum. No entanto, como dentro desse contexto, p possui todos os números primos em sua fatoração e, mesmo assim, $p + 1$ não é divisível por nenhum desses números primos, isso significaria que $p+1$ é primo, o que gera um contradição na hipótese de que L é o conjunto de todos os números primos.

□

Isso significa que existem, de fato, infinitos números primos, contudo, não nos diz nada sobre a distância entre um primo e outro. Contudo, temos um resultado que nos dá pelo menos uma percepção:

Proposição 2.3.2. (Postulado de Bertrand) Existe pelo menos um número primo dentro do intervalo de qualquer número natural e seu dobro. Isto é, dado um número natural n, com $n > 1$, existe um número primo p tal que $n < p < 2n$.

A prova desta proposição é um pouco mais complicada e matematicamente rigorosa, então não a faremos aqui.

2.4 Relações de equivalência

Na matemática, é muito útil conseguirmos estabelecer como diferentes elementos se relacionam entre si, para assim conseguirmos estabelecer padrões entre esses elementos. Uma espécie de relação particularmente importante são as relações de equivalência. Essas relações são usadas para denotar quando dois elementos, ou conjuntos, são equivalentes dentro de algum contexto. Isto é, dentro deste contexto, são, para todos os efeitos, "iguais".

Definição 2.4.1. (Relação de equivalência) Seja C um conjunto no qual esteja definida uma relação, denotada por $\sim$. Esta relação é dita de *equivalência* se, dados elementos x, y e z do conjunto C, as seguintes propriedades são satisfeitas:

1. $x \sim x$, isto é, todo elemento está relacionado consigo mesmo, chamada de *propriedade reflexiva.*

2. Se $x \sim y$, então $y \sim x$, chamada de *propriedade simétrica.*

3. Se $x \sim y$ e $y \sim z$, então $x \sim z$, chamada de *propriedade transitiva.*

Um exemplo bem direto de relação de equivalência é a relação de igualdade dentro do conjunto de números inteiros. Podemos ver que, por exemplo, caso n,m e q estejam contidos nos números inteiros, vale:

1. $n = n$;
2. se $n = m$, então $m = n$;
3. se $n = m$ e $m = q$, então $n = q$.

Como dentro do conjunto dos números inteiros a relação de igualdade satisfaz as três propriedades de relação de equivalência, então a relação de igualdade é uma forma de relação de equivalência. Um exemplo de uma relação dentro dos números inteiros que não é uma relação de equivalência, é a relação de menor que. Por exemplo, $3 < 3$ não é verdade, e também, temos que $3 < 4$, mas não $4 < 3$. Note que mesmo que a relação menor que satisfaça a propriedade transitiva, por exemplo $2 < 3$ e $3 < 6$, portanto $2 < 6$, como a relação menor que não satisfaz as duas primeiras propriedades, ela não é uma relação de equivalência. Para uma relação ser chamada de equivalência, as três propriedades devem ser satisfeitas.

Exemplos de relações de equivalência não faltam: igualdade nos inteiros, a relação de bolas de uma mesma cor em um conjunto de bolas coloridas, a relação de mesmo número de lados em um conjunto de polígonos, a relação de alunos com mesma idade em uma sala de aula e assim por diante.

As relações de equivalência são usadas para classificar elementos de um conjunto em subconjuntos com propriedades semelhantes. E se há uma mania humana, é a mania de classificar tudo. As subdivisões de um conjunto produzidas por uma relação de equivalência são conhecidas como classes

de equivalência. Podemos pensar nelas como pequenos conjuntos dentro de um conjunto maior, cujos elementos desses pequenos conjuntos assumem propriedades semelhantes. Em termos mais formais, seja C um conjunto e $\sim$ uma relação de equivalência definida em C. Se $x \in C$, então a classe de equivalência de x é o conjunto dos elementos de C que são equivalentes a x por $\sim$. Denotando por $\overline{x}$ a classe de equivalência de x temos em símbolos:

$$\overline{x} = \{y \in C : y \sim x\},$$

Ou seja, a classe de equivalência de x em C são todos os elementos em C que estão relacionados a x. Por exemplo, escolha uma bola vermelha dentro de um conjunto B de bolas coloridas. A classe de equivalência desta bola pela relação "bolas de mesma cor" é o subconjunto de todas as bolas vermelhas contidas em B.

Há uma propriedade das classes de equivalência que é tão importante que vamos enunciá-la como um princípio: **qualquer elemento de uma classe de equivalência pode ser escolhido como um representante para toda a classe**. Isto é, se conhecemos um elemento da classe, podemos reconstruir a classe inteira. Imagine que, no exemplo das bolas coloridas, alguém nos diz que o subconjunto que temos diante de nós é uma classe de equivalência. Para sabermos que classe é esta, basta tomarmos uma bola e vermos qual é a sua cor, isto é, com um só elementos podemos identificar a propriedade em comum que é compartilhada por todos os outros elementos da classe.

Voltemos ao conjunto C com a relação de equivalência $\sim$. Para o conjunto C, o princípio acima nos diz que se y é um elemento da classe de x então as classes de x e y são iguais. Isto é:

$$\text{se } \; x \in C \; \text{ e } \; y \in \overline{x}, \; \text{ então } \; \overline{x} = \overline{y}.$$

É importante estudarmos mais duas propriedades do conjunto C com a relação de equivalência $\sim$:

1. C é a união de todas as classes de equivalência.

2. Duas classes de equivalência distintas não podem ter um elemento em comum.

O conjunto das classes de equivalência de $\sim$ em C tem um nome especial, é chamado de **conjunto quociente** de C por $\sim$. Observe que os elementos do conjunto quociente são subconjuntos do conjunto das classes de equivalência de C. No exemplo das bolas, o conjunto quociente de B pela relação de equivalência de bolas de uma mesma cor, seria o conjunto dos subconjuntos das bolas coloridas, onde cada um desses subconjuntos representaria o conjuntos de bolas de uma determinada cor.

2.5 Congruência modular

Já parou para pensar sobre o porquê de contabilizarmos os dias de 24 em 24 horas, sendo que, na verdade, o tempo é um fluxo contínuo? Podemos descrever isso matematicamente do seguinte modo: Fixamos um momento inicial a partir do qual o tempo é contabilizado. Em seguida, estabelecemos uma relação de equivalência: dois momentos que diferem por 24 horas correspondem a horas análogas em dias diferentes. Sim, a forma com que contabilizamos horas em diferentes dias, matematicamente, pode ser interpretado como uma relação de equivalência.

Queremos fazer uma coisa semelhante, não com a sucessão das horas, mas sim com o conjunto dos números inteiros. O conjunto dos inteiros já vem com um marco inicial natural, o número zero. Vamos escolher um número inteiro positivo, que estará fixado a partir deste momento. Para que não precisamos nos comprometer nesta escolha, chamemos de n este número. O número n será chamado de **módulo** dentro desta construção.

Vamos agora construir uma relação de equivalência no conjunto dos inteiros. Diremos que, pulando de n em n, todos os inteiros são equivalentes; ou ainda, dois inteiros cuja diferença é um múltiplo de n são equivalentes. Formalmente, diremos que dois ineiros a e b são **congruentes módulo n** se $a-b$ é um múltiplo de n. Se a e b são congruentes módulo n, escrevemos

$$a \equiv b(mod \ \ n).$$

Para ilustrar melhor, faremos alguns exemplos numéricos. Vamos escolher $n = 5$ como módulo, então

$$10 \equiv 0(mod \ \ 5) \ \text{ e } \ 14 \equiv 24(mod \ \ 5).$$

Neste caso, 10 é congrunte a 0 módulo 5, pois tanto 0 quanto 10 são múltiplos de 5. Já 14 e 24 são congruentes entre si no módulo 5 porque ambos 14 e 24 são um número à menos de algum múltiplo de 5. No caso, 14 é $15-1$ e 24 é $25-1$.

Outro exemplo, com outro módulo; digamos que $n = 7$, então:

$$10 \equiv 3(mod \ \ 7) \ \text{ e } \ 14 \equiv 0(mod \ \ 7),$$

10 é congruente a 3 módulo 7 porque $10-7$ é 3. Poderíamos pensar em 17. 17 é congruente a 3 módulo 7 pois, $17-7 = 10$ e $10 - 7 = 3$, logo $17 - (2.7) = 3$. Desta forma, todos

os números: $3, 10, 17, 24, 31, 38, ...$ são todos congruentes a 3 módulo 7, pois todos são números da forma: $3 + n \cdot 7$, para algum n pertencendo os números inteiros.

Observe que quem é congruente a quem depende do módulo que foi escolhido.

Proposição 2.5.1. Congruência módulo n é uma relação de equivalência

Demonstração. Começamos com a propriedade reflexiva, devemos provar que congruência módulo n satisfaz a propriedade reflexiva. Seja a um inteiro. Para mostrar que $a \equiv a(mod\ \ n)$, temos que verificar, por definição, que a diferença $a - a$ é um múltiplo de n. Mas isto é claro, pois 0 é múltiplo de qualquer inteiro. A propriedade simétrica também é fácil de se verificar. Se $a \equiv b(mod\ \ n)$, então $a - b$ é um múltiplo de n. Mas $b - a = -(a - b)$. logo $b - a$ também é múltiplo de n. Portanto $b \equiv a(mod\ \ n)$.

Quanto à propriedade transitiva, suponhamos que $a \equiv b(mod\ \ n)$ e $b \equiv c(mod\ \ n)$; onde a, b e c são inteiros. A primeira congruência nos diz que $a - b$ é múltipo de n; a segunda que $b-c$ é múltiplo de n. Somando múltiplos de n temos de volta múltiplos de n; logo $(a-b)+(b-c) = (a-c)$ é um múltiplo de n. Portanto $a \equiv c(mod\ \ n)$. Assim, verificamos as três propriedades e podemos concluir que a congruência módulo n é uma relação de equivalência.

□

O conjunto que de fato nos interessa é o conjunto quociente de $\mathbb{Z}$ pela relação de congruência módulo n. Este conjunto tem uma notação própria, $\mathbb{Z}_n$; e um nome especial, *conjunto dos inteiros módulo n*. Precisamos identificar os elementos de $\mathbb{Z}_n$. Sabemos, por definição, que são subconjuntos de $\mathbb{Z}$: as classes de equivalência da congruência módulo n. Seja $a \in \mathbb{Z}$. A classe de a é formada pelos $b \in \mathbb{Z}$

que satisfazem $b - a$ é múltiplo de n isto é, são números da forma $b - a = k \cdot n$, para algum $k \in \mathbb{Z}$. Podemos assim descrever a classe de a na forma

$$\overline{a} = a + k \cdot n, \quad \text{para } k \in \mathbb{Z}.$$

Em particular, $\overline{0}$ é o conjunto dos múltiplos de n.

Isto produz uma situação curiosa. Se $a \in \mathbb{Z}$, então podemos dividi-lo por n, obtendo q e r inteiros tais que

$$a = n \cdot q + r \quad e \quad 0 \leq r \leq n - 1.$$

Logo, $a - r = n.q$ é um múltiplo de n. Portanto $a \equiv r(mod \quad n)$. Isto é, um inteiro qualquer é congruente módulo n a algum inteiro no intervalo que vai de 0 a $n - 1$. Em outras palavras, o conjunto quociente $\mathbb{Z}_n$ é formado pelas classes $\overline{0}, \overline{1}, \overline{2}, ..., \overline{n-1}$. Além disso, duas destas classes não podem ser iguais: a única maneira de dois números entre 0 e $n-1$ serem congruentes módulo n é se eles forem o mesmo número. Resumindo

$$\mathbb{Z}_n = \{\overline{0}, \overline{1}, ..., \overline{n-1}\}.$$

Quando uma classe de $\mathbb{Z}_n$ estiver representada na forma $\overline{a}$ com $0 \leq a \leq n-1$, diremos que está na **forma reduzida**.

Por exemplo, vamos olhar para o conjunto quociente de $\mathbb{Z}$ pela relação congruência módulo 5. Seria, então, descrito pelo conjunto:

$$\mathbb{Z}_5 = \{\overline{0}, \overline{1}, \overline{2}, \overline{3}, \overline{4}\}$$

onde

- $\overline{0}$ = {0,5,10,15,20,25,...}
- $\overline{1}$ = {1,6,11,16,21,26,...}

- $\overline{2} = \{2,7,12,17,22,27,...\}$
- $\overline{3} = \{3,8,13,18,23,28,...\}$
- $\overline{4} = \{4,9,14,19,14,29,...\}$

Onde, percebemos que não haveria necessidade de colocar outras classes além dessas, porque qualquer outra classe acabaria sendo uma dessas 5, já que qualquer número inteiro se encontra em alguma dessas 5 classes.

Note que podemos sempre tomar os representantes de cada classe de congruência módulo n entre 0 e $n-1$. Você pode se perguntar "por que precisaríamos escolher outros representantes?". No entanto, para alguns casos dos quais trataremos mais adiante, pode ser conveniente que tomemos representantes distintos para cada uma das classes. Quando construímos um conjunto de números inteiros com n elementos, escolhendo exatamente um elemento de cada classe de congruência módulo n, denominamos esse conjunto *sistema completo de resíduos* módulo n.

Como vimos, $\{0, 1, 2, 3, 4\}$ forma um sistema completo de resíduos módulo 5, no entanto, também podemos construir outros sistemas completos de resíduos, como $\{5, 11, 17, 23, 29\}$.

Como devemos imaginar $\mathbb{Z}_n$? Geralmente pensamos em $\mathbb{Z}$ como sendo o conjunto dos pontos marcados ao longo de uma reta horizontal, de uma em uma unidade. Imagine agora que enrolamos esta reta em uma circunferência colocando o ponto n ao ponto 0. Como a reta é infinita, continuamos a enrolá-la na circunferência. Desta maneira os pontos cujas coordenadas são múltiplos de n coincidem com o ponto zero. A imagem geométrica correspondente a $\mathbb{Z}_n$ é, portanto, a de uma circunferência, onde estão marcados n pontos equidistantes. Cada ponto corresponde a uma das classes de equivalência de $\mathbb{Z}_n$.

2.6 Aritmética modular

Esta imagem geométrica é muito útil quando se trata de definir a soma de elementos de $\mathbb{Z}_n$. Vamos imaginar que temos um relógio de n horas, com um ponteiro. Isto é, imagine as classes $\overline{0}$ a $\overline{n-1}$ dispostas ao longo de uma circunferência, a intervalos iguais, no sentido horário; e vamos dotar este relógio de um ponteiro, fixado na circunferência. Desta forma, construímos uma máquina de calcular mecânica para operar em $\mathbb{Z}_n$.

2.6.1 Soma de classes

Vejamos como funciona a máquina. Por exemplo, para somar duas classes $\overline{a}$ e $\overline{b}$, colocamos o ponteiro na classe a e depois movemos o ponteiro b unidades no sentido horário. O resultado da soma é a classe para a qual o ponteiro está apontando depois desta operação. Por exemplo, vamos considerar $\mathbb{Z}_7$, e vamos somar, $\overline{5}$ e $\overline{4}$ em $\mathbb{Z}_7$. Colocamos o ponteiro em 5 e vamos mover ele quatro casas. Como no nosso relógio imaginário existem sete horas, pois estamos em $\mathbb{Z}_7$, ao movermos duas casas a partir de 5, chegaremos em $\overline{0}$, e depois moveremos mais duas casas e chegaremos em $\overline{2}$. Logo, $\overline{5}+\overline{4}=\overline{2}$ em $\mathbb{Z}_7$.

Essa abstração imaginando a operação soma de classes como um relógio é muito útil para conseguirmos ter uma intuição sobre como funciona, no entanto, para descrevermos essa operação matematicamente, fazemos o seguinte. Precisamos descrever esta mesma operação de modo matemático. Isto é fácil. Sejam $\overline{a}$ e $\overline{b}$ as classes de $\mathbb{Z}_n$ que desejamos somar. A fórmula para a operação é a seguinte:

$$\overline{a}+\overline{b}=\overline{a+b}.$$

É preciso interpretar esta fórmula corretamente. À esquerda temos a soma de duas classes; à direita temos a classe da soma de dois números inteiros. Assim definimos a operação soma de classes usando uma operação que já nos é conhecida, a soma de inteiros. Voltando ao exemplo em $\mathbb{Z}_7$. De acordo com esta fórmula, para somar $\overline{5}$ a $\overline{4}$, somando os inteiros 4 e 5 temos 9; logo $\overline{5}+\overline{4}=\overline{9}$. Como $9-7=2$ temos que 9 e 2 estão na mesma classe de equivalência de $\mathbb{Z}_7$, isto é, $\overline{9}=\overline{2}$. Portanto, $\overline{5}+\overline{4}=\overline{2}$. É importante notar que isto é válido independente da escolha de representantes para a classe. Por exemplo, $\overline{75}+\overline{53}=\overline{5}+\overline{4}$, pois 75 e 5, 53 e 4 respectivamente fazem parte da mesma classe de equivalência em $\mathbb{Z}_7$, portanto, qualquer um pode servir como representante da sua classe toda. A subtração de classes funciona exatamente igual à soma.

2.6.2 Multiplicação de classes

A definição matemática é similar, isto é, a multiplicação de duas classes $\overline{a}$ e $\overline{b}$ em $\mathbb{Z}_n$ é definida por:

$$\overline{a}\cdot\overline{b}=\overline{a\cdot b}.$$

E, assim como na soma, no produto o resultado também segue igual, independentemente da escolha de representantes.

2.6.3 Elemento simétrico

O elemento $\overline{-a}$ é chamado de o *simétrico* de $\overline{a}$ para a operação soma. Supondo que $\overline{a}$ está na forma reduzida, a forma reduzida de $\overline{-a}$ é $\overline{n-a}$. Por exemplo, em $\mathbb{Z}_5$, temos $\overline{-3}=\overline{5-3}=\overline{2}$.

2.6.4 Propriedades de equivalência

Agora que vimos como funciona a soma, a subtração e o produto modular, vamos listar, agora, algumas das propriedades dessas operações:

- $(\overline{a} + \overline{b}) + \overline{c} = \overline{a} + (\overline{b} + \overline{c})$;
- $\overline{a} + \overline{b} = \overline{b} + \overline{a}$;
- $\overline{a} + \overline{0} = \overline{a}$;
- $\overline{a} + \overline{-a} = \overline{0}$;
- $(\overline{a}\overline{b}) \cdot \overline{c} = \overline{a} \cdot (\overline{b}\overline{c})$;
- $\overline{a} \cdot \overline{b} = \overline{b} \cdot \overline{a}$;
- $\overline{a} \cdot \overline{1} = \overline{a}$;
- $\overline{a} \cdot (\overline{b} + \overline{c}) = \overline{a} \cdot \overline{b} = \overline{a} \cdot \overline{c}$;

2.7 Divisão Modular

Foram apresentadas até aqui as operações de adição, subtração e multiplicação em $\mathbb{Z}_n$. Resta agora avaliar como definimos uma operação de "divisão" nesse conjunto. A analogia com números reais é um bom ponto de partida. Assim como podemos pensar que a subtração $a - b$ equivale à soma entre a e o *oposto* de b, nos números reais podemos conceber a divisão a/b como uma multiplicação entre a e o *inverso* de b com relação à multiplicação, *desde que*, é claro, $b \neq 0$.

Assim, parece natural à primeira vista definir a divisão de $\overline{a}$ por $\overline{b}$ em $\mathbb{Z}_n$ como a multiplicação $\overline{a} \cdot \overline{\beta}$, em que $\overline{b} \cdot \overline{\beta} = \overline{1}$. No entanto, o problema é um pouco mais delicado que isso. No caso de números reais, como vimos, basta que $b \neq 0$ para que

seu inverso exista. Em $\mathbb{Z}_n$, a condição $\overline{b} \neq \overline{0}$, embora obviamente necessária, não é suficiente para garantir a existência de um inverso. Considere, por exemplo, o elemento $\overline{2}$ em $\mathbb{Z}_6$. Note que 6 é um número par, e portanto o resto da divisão de qualquer número par por 6 será $0, 2$ ou 4. Em linguagem de classes de equivalência isso significa $\overline{2} \cdot \overline{b} = \overline{0}, \overline{2}$ ou $\overline{4}$ para qualquer $\overline{b} \in \mathbb{Z}_6$. Em cada caso $\overline{2 \cdot b} \neq \overline{1}$, e portanto, não existe elemento inverso de $\overline{2}$ em $\mathbb{Z}_6$. Isso nos motiva a seguinte definição em $\mathbb{Z}_n$:

Definição 2.7.1. Dizemos que um elemento $\overline{b}$ de $\mathbb{Z}_n$ é *inversível* se existe $\overline{\beta} \in \mathbb{Z}_n$ tal que $\overline{b \cdot \beta} = \overline{1}$ em Z_n. Nesse caso, $\overline{\beta}$ é dito *inverso* de $\overline{b}$.

Traduzindo para linguagem de congruências, podemos dizer que b é *inversível* módulo n se existe inteiro β tal que:

$$b \cdot \beta \equiv 1 (mod\ n).$$

Para determinarmos quando um elemento é inversível, olhemos com mais atenção para a congruência

$$a \cdot \alpha \equiv 1 (mod\ n).$$

Note que isso equivale a afirmar que n divide $a \cdot \alpha - 1$, e portanto, existe $k \in \mathbb{Z}$ tal que:

$$a\alpha + kn = 1.$$

Seja $d = mdc(a, n)$. Pela equação acima, sabemos que d divide $a\alpha + kn$. Isto é, sejam $a = d \cdot a', n = d \cdot n'$, então:

$$1 = a\alpha + kn = d(a'\alpha + kn').$$

Portanto d é um inteiro positivo divisor de 1. Disso concluímos que $mdc(a, n) = 1$, isto é, se a possui inverso em $\mathbb{Z}_n$, então a e n são coprimos. É possível verificar a

recíproca dessa afirmação sem grandes dificuldades. Seja $mdc(a, n) = 1$. Então, aplicando o Algoritmo de Euclides, conseguimos encontrar inteiros α, β, tais que:

$$a \cdot \alpha + n \cdot \beta = 1.$$

Disso, concluímos que n divide $a \cdot \alpha - 1$, logo, em $\mathbb{Z}_n$, temos que $\overline{a \cdot \alpha} = \overline{1}$. Com isso, provamos o seguinte resultado:

Proposição 2.7.2. A classe $\overline{a}$ tem inverso em $\mathbb{Z}_n$ se, e somente se, a e n são coprimos.

Note que a demonstração que fizemos também nos dá um caminho para calcular o inverso de $\overline{a}$ em $\mathbb{Z}_n$, quando este existe. De fato, o número α que encontramos usando o Algoritmo de Euclides nos fornece precisamente a classe inversa à de a. Por exemplo, calculemos o inverso multiplicativo de $\overline{3}$ em $\mathbb{Z}_{32}$. Ao dividirmos 32 por 3 no Algoritmo de Euclides, temos:

$$\begin{aligned} 32 &= 10 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1. \end{aligned}$$

Assim, obtendo 1 como combinação de 3 e 32, temos:

$$1 = 3 - 1 \cdot 2 = 3 - 1 \cdot (32 - 10 \cdot 3) = -1 \cdot 32 + 11 \cdot 3.$$

Disso, concluímos que $\overline{11}$ é o inverso multiplicativo de $\overline{3}$ em $\mathbb{Z}_{32}$.

O conjunto dos elementos de $\mathbb{Z}_n$ que admitem inverso multiplicativo será bastante útil para nós mais adiante. Por isso, fixaremos uma notação para ele, do seguinte modo:

$$U(n) := \{\overline{a} \in \mathbb{Z}_n : mdc(a, n) = 1\}.$$

Se p é um número primo, podemos facilmente determinar $U(p)$. De fato, se $mdc(a, p) \neq 1$, temos justamente que p

divide a, logo $\overline{a} \equiv \overline{0}(mod\ p)$. Assim, para todas as classes $\overline{a}$ diferentes de $\overline{0}$ em $\mathbb{Z}_p, mdc(a,p) = 1$ e existe classe inversa. Portanto:

$$U(p) = \mathbb{Z}_p - \{\overline{0}\}.$$

Caso n seja composto, determinar $U(n)$ exige um pouco mais de atenção. Por exemplo, $U(12) = \{\overline{1}, \overline{5}, \overline{7}, \overline{11}\}$.

Cabe destacar ainda uma propriedade de $U(n)$, que é o fato de que este conjunto é *fechado para multiplicação*. Com isso, queremos dizer que sempre que multiplicamos dois elementos de $U(n)$, o resultado ainda pertence a $U(n)$. Provemos que, se $\overline{a}$ e $\overline{b}$ são inversíveis em $\mathbb{Z}_n$, então $\overline{a \cdot b}$ também admite elemento inverso em $\mathbb{Z}_n$. De fato, sejam $\overline{\alpha}, \overline{\beta}$ os respectivos inversos de $\overline{a}, \overline{b}$. É fácil verificar que $\overline{\alpha \cdot \beta}$ será o inverso que procuramos:

$$(\overline{a \cdot b})(\overline{\alpha \cdot \beta}) = (\overline{a \cdot \alpha})(\overline{b \cdot \beta}) = \overline{1 \cdot 1} = \overline{1}.$$

Assim como no caso de um sistema completo de resíduos módulo n, pode ser conveniente em alguns casos separarmos um conjunto com exatamente um representante de cada uma das classes inversíveis em $\mathbb{Z}_n$. Denominaremos esse conjunto *sistema de resíduos reduzido* módulo n. Por exemplo, $\{1, 5, 7, 11\}$ é um sistema de resíduos reduzido módulo 12, bem como $\{5, 25, 35, 55\}$.

2.8 Potências

Graças às propriedades da multiplicação de classes de congruência, as congruências módulo n também permitem simplificar o cálculo dos restos da divisão de potências de um inteiro a qualquer por n. Como exemplo, vamos pensar em como calcular o resto 10^{135} na divisão por 7.

Primeiro, podemos tomar um representante mais simples para as contas, $10 \equiv 3(mod\ 7)$. Se fizermos as congruências das potências, obtemos:

$$\begin{aligned}
3^2 &\equiv 3 \cdot 3 \equiv 2(mod\ 7)\\
3^3 &\equiv 3 \cdot 2 \equiv 6 \equiv -1(mod\ 7)\\
3^4 &\equiv 3 \cdot (-1) \equiv -3 \equiv 4(mod\ 7)\\
3^5 &\equiv 3 \cdot 4 \equiv 5(mod\ 7)\\
3^6 &\equiv 3 \cdot 5 \equiv 1(mod\ 7).
\end{aligned}$$

Observe que $10^6 \equiv 1(mod\ 7)$ e, a partir da sexta potência, todos os restos se repetem ciclicamente. Por sua vez, dividindo 135 por 6, obtemos $135 = 22 \cdot 6 + 3$. Logo, as congruências de 10^{135} em módulo 7 são dadas por:

$$10^{135} \equiv (10^6)^{22} \cdot 10^3 \equiv (1)^{22} \cdot 10^3 \equiv 10^3 \equiv 6(mod\ 7).$$

Note que, no caso de módulo 7, conseguimos rapidamente encontrar todos os restos possíveis por inspeção. Nem sempre será viável tal cálculo de todos os restos. Considere agora o problema de calcular o resto da divisão de 3^{64} por 31. A princípio, não sabemos qual potência de 3 terá resto 1. Proceder por inspeção seria bem demorado nesse caso. Tentaremos encontrar alguma potência de 3 que seja mais conveniente para as contas. Podemos verificar, sem grande esforço, que $3^3 \equiv -4(mod\ 31)$, e $64 = 21 \cdot 3 + 1$. Note que é possível fazer a seguinte substituição:

$$-4^{21} = (-1)^{21} \cdot (2^2)^{21} = -2^{42},$$

e, portanto:

$$3^{64} \equiv (3^3)^{21} \cdot 3 \equiv (-4)^{21} \cdot 3 \equiv -2^{42} \cdot 3(mod\ 31).$$

Pode não ter ficado claro em um primeiro momento o porquê de essa substituição em particular ser conveniente, mas note que $2^5 = 32$, o que, em congruências módulo 31, significa $2^5 \equiv 1(mod\ 31)$. Uma vez que $42 = 8 \cdot 5 + 2$, temos:

$$3^{64} \equiv -2^{42} \cdot 3 \equiv -(2^5)^8 \cdot 2^2 \cdot 3 \equiv -4 \cdot 3 \equiv -12(mod\ 31).$$

Para concluir, tomamos o menor representante positivo da classe $\overline{-12}$, que é $31 - 12 = 19$. Portanto, o resto da divisão de 3^{64} por 31 é 19.

Você deve ter percebido que, para qualquer inteiro a, se alguma de suas potências tiver resto 1 na divisão por n, então os restos das divisões de todas as potências seguintes se repetirão ciclicamente. Isto significa que, se para algum r inteiro, $a^r \equiv 1(mod\ n)$, então, para todo k, $a^{r+k} \equiv a^k(mod\ n)$. Determinar tais valores de r será bastante pertinente para simplificar as operações de potências modulares.

Definição 2.8.1. (Ordem de um inteiro modular) : Sejam a e n dois inteiros positivos. Dizemos que a *ordem* de a módulo n é o *menor* inteiro positivo $k > 0$ tal que:

$$a^k \equiv 1(mod\ n).$$

Seja k a ordem de a módulo n. Para cada múltiplo kl de k, é fácil verificar que $a^{kl} \equiv 1(mod\ n)$. É importante deixar registrado que a recíproca também é verdadeira, isto é, se $a^m \equiv 1(mod\ n)$, então k divide m. De fato, seja $m = q \cdot k + r, 0 \leq r < k$. Então:

$$1 \equiv (a^k)^q \cdot a^r \equiv a^r(mod\ n).$$

Mas, como k é o menor positivo tal que $a^k \equiv 1(mod\ n)$ e $r < k$, temos que $r = 0$, e portanto, k divide m.

Contudo, como já foi observado na seção de divisibilidade modular, dado $\overline{a}$ em $\mathbb{Z}_n$, nem sempre existe um elemento

inverso para $\overline{a}$. Se $\overline{a}$ não for inversível em $\mathbb{Z}_n$, então, para qualquer elemento $\overline{b} \in \mathbb{Z}_n, \overline{a \cdot b} \neq \overline{1}$. Em particular, isso significa que nenhuma potência de a será congruente a 1 módulo n. Por exemplo, se tomarmos $\overline{2}$ em $\mathbb{Z}_6$, veremos que as congruências de suas potências serão:

$$\begin{aligned} 2^1 &\equiv 2(mod\ 6) \\ 2^2 &\equiv 4(mod\ 6) \\ 2^3 &\equiv 2(mod\ 6) \\ 2^4 &\equiv 4(mod\ 6) \\ &\vdots \end{aligned}$$

Assim, as potências de $\overline{2}$ sempre se alternarão entre $\overline{2}$ e $\overline{4}$, sem nunca atingir $\overline{1}$.

Com isso, podemos concluir que, se a tem alguma ordem módulo n, então $\overline{a}$ é inversível nesse módulo, o que equivale a afirmar $\overline{a} \in U(n)$. Como veremos mais adiante, sempre que $\overline{a} \in U(n)$, existe algum inteiro positivo k tal que $a^k \equiv 1(mod\ n)$, e, portanto, existe alguma ordem para a módulo m.

Antes de prosseguir, cabe destacar ainda um tipo de elemendo de $U(n)$:

Definição 2.8.2. (Raiz Primitiva) : Dizemos que $\overline{a}$ é uma *raiz primitiva* de $U(n)$ se podemos obter todo elemento de $U(n)$ a partir de potências de $\overline{a}$.

Por exemplo, para $U(7)$, temos que $\overline{3}$ é uma raiz primitiva. De fato, como foi verificado anteriormente, cada um dos seis elementos de $U(7)$ é obtido a partir de alguma potência de $\overline{3}$.

Contudo, nem todo conjunto $U(n)$ possui raiz primitiva. Considere o conjunto $U(8) = \{\overline{1}, \overline{3}, \overline{5}, \overline{7}\}$. Para cada elemento $\overline{a} \in U(8)$, temos que $\overline{a^2} = \overline{1}$, logo, nenhum elemento diferente de $\overline{1}$ é potência de outro elemento.

2.9 Função ϕ de Euler

Definiremos agora a seguinte função sobre os inteiros positivos:

Definição 2.9.1. (Função ϕ de Euler): Seja n um número inteiro positivo. A *função ϕ de Euler* em n, denotada $\phi(n)$, é a quantidade de números inteiros positivos menores ou iguais a n que são relativamente primos com n.

Anteriormente, nós definimos $U(n)$ como o conjunto de todos os elementos inversíveis de $\mathbb{Z}_n$. Essa definição, como foi visto, equivale a:

$$U(n) := \{\overline{a} \in \mathbb{Z}_n : mdc(a, n) = 1\}.$$

Como podemos associar a cada elemento de $\mathbb{Z}_n$ um representante dado entre 0 e $n-1$, podemos dizer, equivalentemente, que a função $\phi(n)$ é exatamente a quantidade de elementos de $U(n)$.

A princípio, para algum n qualquer, precisamos avaliar $\phi(n)$ por inspeção. No entanto, algumas propriedades permitem que calculemos $\phi(n)$ mais facilmente, como vemos a seguir.

O caso mais simples a considerar é, naturalmente, $\phi(p)$, quando p é primo. Neste caso, sabemos que, para qualquer a, com $0 < a < p, mdc(a, p) = 1$, logo $\phi(p) = p - 1$. Da mesma forma, se p^k é potência de algum primo p, a função ϕ de Euler será dada por:

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

De fato, sabemos que, se $mdc(p, a) \neq 1$, então p divide a. Logo, basta desconsiderarmos apenas os múltiplos de p entre 1 e p^k. Cada múltiplo de p é da forma $q \cdot p$, e para $1 \leq q \leq p^{k-1}$ temos que $p \leq q \cdot p \leq p^k$. Logo, devemos ter

exatamente p^{k-1} múltiplos de p entre 1 e p^k. E portanto, os restantes $p^k - p^{k-1}$ números serão relativamente primos a p.

Enunciamos agora outra propriedade da função ϕ de Euler, que permite generalizá-la para qualquer inteiro positivo.

Proposição 2.9.2. Sejam m e n inteiros positivos tais que $mdc(m, n) = 1$. Então:

$$\phi(m \cdot n) = \phi(m) \cdot \phi(n).$$

Demonstração. Para uma melhor visualização, vamos distribuir os números de 1 a mn em uma matriz $m \times n$, da seguinte forma:

$$\begin{bmatrix} 1 & m+1 & 2m+1 & \dots & (n-1)m+1 \\ 2 & m+2 & 2m+2 & \dots & (n-1)m+2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ m & 2m & 3m & \dots & nm \end{bmatrix}$$

Agora, se para algum $i, 1 \leq i \leq m$ tivermos $mdc(i, m) = d > 1$, então nessa linha inteira $mdc(i, mn) \geq d > 1$, e nenhum elemento será relativamente primo a mn. Logo, apenas $\phi(m)$ linhas terão todos elementos coprimos com m. Consideremos então as i-ésimas linhas, em que $mdc(i, m) = 1$. Cada linha possui n elementos, e como $mdc(m, n) = 1$, os números $i, m+i, 2m+i, \dots, (n-1)m+i$ formam o sistema completo de raízes módulo n, isto é, cada classe de congruência módulo n aparece exatamente uma vez. Disso resulta que cada linha terá $\phi(n)$ elementos coprimos com n. Sendo os números coprimos com mn aqueles que são simultaneamente coprimos com m e com n, a quantidade destes será precisamente a multiplicação das $\phi(m)$ linhas coprimas com m pelos $\phi(n)$ elementos de cada linha coprimos com n. Logo:

$$\phi(m \cdot n) = \phi(m) \cdot \phi(n).$$

□

2.10 Teoremas de Fermat e de Euler

Restam ainda dois resultados fundamentais sobre potências modulares, que abordaremos agora: o Pequeno Teorema de Fermat, que trata de módulos primos, e o Teorema de Euler, que nada mais é que uma generalização do primeiro resultado. Pois bem, vamos enunciá-los:

Teorema 2.10.1 (Pequeno Teorema de Fermat)**.** Seja p um número primo e a um inteiro que não é múltiplo de p. Então $a^{p-1} \equiv 1(mod\ p)$.

Demonstração. Inicialmente, vamos observar o seguinte conjunto:

$$A = \{a, 2a, 3a, \ldots, (p-1)a\}.$$

Se tomamos quaisquer dois elementos desse conjunto, temos que eles *não* são congruentes entre si módulo p. Se tivéssemos $xa \equiv ya(mod\ p)$, para $1 \leq x, y \leq p-1$, como a é inversível módulo p, podemos multiplicar os dois lados da congruência por seu inverso, e obtemos $x \equiv y(mod\ p)$, o que é impossível, pois sendo x, y possíveis restos distintos da divisão por p, suas classes de congruência são necessariamente distintas. Também temos que nenhum elemento de A é congruente a 0 módulo p, pois para $xa \in A$, tanto x quanto a são relativamente primos a p.

Assim, podemos constatar que A é um *sistema de resíduos reduzido* módulo p. Portanto, para alguma reordenação $x_1, x_2, \ldots, x_{p-1}$ dos inteiros $1, 2, \ldots, p-1$, encontramos as seguintes congruências:

$$\begin{aligned} a &\equiv x_1 (mod\ p) \\ 2a &\equiv x_2 (mod\ p) \\ &\vdots \\ (p-1)a &\equiv x_{p-1} (mod\ p). \end{aligned}$$

A partir disso, podemos simplesmente multiplicar todos os elementos de cada lado, e a congruência se mantém. Como todos os números entre 1 e $p-1$ ocorrem em cada lado, o resultado da nossa multiplicação será:

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

Por fim, como $(p-1)!$ é um produto de números coprimos com p, sua classe de equivalência é inversível em $\mathbb{Z}_p$ e podemos simplesmente multiplicar os dois lados da congruência pelo seu inverso. Isso nos dá o resultado:

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

□

Corolário 2.10.2. Seja p um número primo qualquer, e a um inteiro qualquer. Então:

$$a^p \equiv a(mod\ p).$$

Demonstração. Se p não divide a, então basta multiplicar por a os dois lados da congruência $a^{p-1} \equiv 1(mod\ p)$ e o resultado segue.

Se p divide a, então $a \equiv 0(mod\ p)$ e o resultado é imediato. □

Como pudemos notar, o Pequeno Teorema de Fermat vale *especificamente* para módulos primos. Você pode estar se perguntando o porquê de não podermos generalizá-lo para qualquer módulo. Pois bem, vamos olhar com um pouco mais de atenção para a demonstração que fizemos. Note que do começo até quase o fim da demonstração, conseguimos repetir todos os passos. No entanto, ao final, dependemos fortemente do fato de que $(p-1)!$ é inversível módulo p. Pois bem, se n é um número composto, então $mdc(n,(n-1)!) \neq 1$, e por isso não conseguimos inverter $(n-1)!$ no módulo n, e consequentemente não podemos cancelar o elemento na congruência final.

Agora, o que aconteceria se nós escolhêssemos apenas os números menores que n que são inversíveis módulo n? Será que conseguiríamos algum tipo de resultado análogo? Sem nos demorarmos muito, a resposta é afirmativa, e é justamente o que afirma o Teorema de Euler. Porém, antes de passarmos a ele, precisamos firmar brevemente um resultado.

Seja $U(n) = \{\overline{x_1}, \ldots, \overline{x_{\phi(n)}}\}$, em que $x_1, \ldots x_{\phi(n)}$ são representantes de cada uma das $\phi(n)$ classes distintas que são inversíveis módulo n. Isto significa que o conjunto $B = \{x_1, \ldots, x_{\phi(n)}\}$ forma um sistema reduzido de resíduos. Se $mdc(a,n) = 1$, o que acontece se multiplicarmos todos os elementos do conjunto B por a? Sabemos que $a \cdot x_i$, para qualquer valor de i entre 1 e $\phi(n)$, continua sendo inversível módulo n. No entanto, será que conseguimos novamente representantes de todas as classes inversíveis, formando um outro sistema reduzido de resíduos?

Vejamos. Se acontecer $a \cdot x_i \equiv a \cdot x_j (mod\ n)$, como a é inversível módulo n, podemos cancelá-lo dos dois lados da congruência, obtendo $x_i \equiv x_j (mod\ n)$. No entanto, ao tomarmos dois elementos quaisquer de B, eles não são congruentes entre si. Logo, cada $a \cdot x_i$ corresponde a uma classe

de congruência distinta das demais, e assim todas as classes são representadas, e obtemos novamente um sistema reduzido de resíduos. Com esse fato em mente, podemos passar ao Teorema de Euler.

Teorema 2.10.3 (Teorema de Euler)**.** Sejam a e n inteiros, com $n > 1$, tais que $mdc(a, n) = 1$. Então:

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

Demonstração. A prova será muito parecida com a que já fizemos no Pequeno Teorema de Fermat. Seja $B = \{x_1, \ldots, x_{\phi(n)}\}$ um sistema de resíduos reduzido de n. Podemos reordenar os elementos de B de alguma forma $x_{i_1}, \ldots, x_{i_{\phi(n)}}$, tal que as seguintes congruências sejam válidas:

$$\begin{aligned} a \cdot x_1 &\equiv x_{i_1}(mod\ n) \\ a \cdot x_2 &\equiv x_{i_2}(mod\ n) \\ &\vdots \\ a \cdot x_{\phi(n)} &\equiv x_{i_{\phi(n)}}(mod\ n). \end{aligned}$$

Como na demonstração anterior, simplesmente multiplicamos todos os elementos em cada lado das congruências, obtendo:

$$x_1 \cdot x_2 \cdot \ldots \cdot x_{\phi(n)} \cdot a^{\phi(n)} \equiv x_1 \cdot x_2 \cdot \ldots \cdot x_{\phi(n)}(mod\ n).$$

Como cada x_i nessa multiplicação é inversível, podemos simplesmente cancelá-los um a um. No fim do processo, temos:

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

□

2.11 Exercícios

Exercício 2.11.1. Encontre a fatoração em primos de: 56, 94, 260, 78 e 196.

Exercício 2.11.2. Qual é o menor número que devemos adicionar a 25013 para que a soma seja divísivel ao mesmo tempo por 3 e por 7?

Exercício 2.11.3. Se um número n for dividido por 27, o resto da divisão será igual a 7. Se dividirmos o número $n+50$ também por 27, qual será o resto obtido?

Exercício 2.11.4. Sabendo que $k \equiv 1(mod\ 4)$, mostre que $6k + 5 \equiv 3(mod\ 4)$.

Exercício 2.11.5. Determine o resto da divisão de 5^{60} por 26.

Exercício 2.11.6. Encontre os inversos módulo 11 dos seguintes valores: 122, 37, 52, 65, 86, 79, 102, 16, 117, 215

Exercício 2.11.7. Resolva a congruência $4x \equiv 9(mod\ 13)$

Exercício 2.11.8. Calcule a ordem de:

- 3 módulo 7
- 2 módulo 11
- 5 módulo 31
- 7 módulo 43

Exercício 2.11.9. Encontre o valor da função ϕ de Euler para os números 21, 35 e 55

Exercício 2.11.10. Calcule os restos das seguintes divisões utilizando o Teorema de Fermat.

- 3^{98745} por 43

- 3^{1034^2} por 1033
- 2^{41048^2} por 41047

Exercício 2.11.11. Calcule os restos das seguintes divisões utilizando o Teorema de Euler.

- 2^{495} por 15841
- 2^{41045} por 41041
- 2^{77} por 2465

Capítulo 3

Criptografia RSA

3.1 Motivação

Iremos entender um dos algoritmos que garante a sua segurança quando navega na internet, realiza compras utilizando o cartão de crédito, utiliza o internet banking do seu banco pessoal, ou até mesmo quando precisa mandar mensagens pelo aplicativo do celular. A **Criptografia Assimétrica RSA**, ou simplesmente **criptografia RSA** consiste em um algoritmo de chave pública, o que significa que funciona com duas *chaves*: a pública, que é utilizada para criptografar as mensagens; e a *privada*, quel tem o papel de descriptografar os dados, o acesso a esta segunda é limitado.

Afinal, qual a grande importância da criptografia RSA? Já que dedicaremos um capítulo inteiro para discuti-lá, é um questionamento válido. A grande vantagem de utilizar esse algoritmo é a dificuldade para descriptografar uma mensagem. Veremos que o processo de criptografia depende do produto de dois números primos. Realizar esse produto não gera muito trabalho computacionalmente, mas uma vez realizado, é inviável descobrir quem eram os primos originais

sem ter a chave. Os exemplos apresentados usarão números primos pequenos, para o leitor compreender com mais facilidade, mas na prática, os números primos utilizados passam dos 100 dígitos, e embora computacionalmente seja possível multiplicar dois números desse tamanho, o resultado será um número com muito mais dígitos e a tarefa de descobrir quem são os primos originais se torna inviável.

Agora que já sabemos da importância da criptografia RSA, vamos entender as diferenças entre chave pública e privada e o que isso muda no algoritmo.

3.2 Diferenças entre chave pública e chave privada

Vamos considerar como exemplo a **Cifra de César**, um dos métodos mais simples de criptografia e que pode até ser quebrado usando lápis e papel. Apesar da simplicidade, é um dos primeiros algoritmos de criptografia que se tem registro e era utilizado pelo imperador romano Júlio César, como meio de transmitir mensagens militares. O seu funcionamento é bem simples, basta pegar o alfabeto e transladar as letras um número entre 1 e 25, na imagem abaixo temos um exemplo:

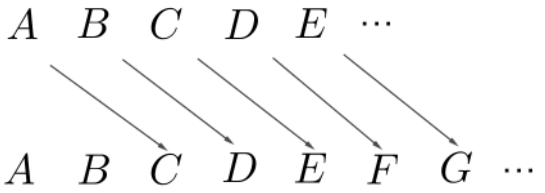


Figura 3.1: Aplicação da Cifra de César

No exemplo acima, transladamos cada letra 2 casas (da esquerda para a direita), então se α é uma letra, sua codificação será $\alpha + 2$, aqui estamos associando a letra a com o número 1, b com 2, e assim por diante.

Observação 3.2.1. No exemplo acima, a letra X será equivalente à Z, ou seja, para saber à quanto Y será equivalente, será necessário voltar o alfabeto ao começo, assim Y será equivalente à B. Complete a figura 3.1 para ter uma noção melhor.

A **chave** neste exemplo é saber qual número foi usado para a translação e assim, só temos uma chave (neste caso, privada), logo tanto quem emite a mensagem quanto quem recebe precisa sabê-la.

Exercício 3.2.2. Marcus mandou uma mensagem criptografada para a Ana. A mensagem é muito importante mas Ana não se lembra da chave para descriptografar. Sabendo que foi utilizado a Cifra de César, ajude Ana a revelar o conteúdo da mensagem:

Rl Dqd, r vhx glqkhlur hvwd qr phx dupdulr.

Dica: Tente procurar elementos familiares (tamanho das palavras e quais letras poderiam estar ocupando essa palavra, além de ver qual letra mais aparece e comparar com as letras de uma mensagem normal) e lembre-se que se você achar uma letra correta, você consegue deduzir toda a mensagem a partir dela.

Vimos em prática um algoritmo que usa uma chave privada. No caso da criptografia RSA, duas chaves são usadas, uma pública e uma privada. Iremos considerar os objetos que trocarão as mensagens como cliente e servidor, idealmente tanto o cliente quanto o servidor possuem chaves cada um, uma chave pública e uma privada. O processo funciona da

seguinte forma:

- O cliente e o servidor trocam suas chaves públicas.
- O cliente usa a chave pública do servidor e criptografa a mensagem.
- Com a mensagem criptografada em mãos, o servidor utiliza a sua chave privada e descriptografa a mensagem do cliente.
- Após descriptografada, o servidor irá criptografar o que será enviado para o cliente usando a chave pública do cliente.
- Por fim, o cliente irá descriptografar a mensagem usando sua chave privada.

O processo acima, por exemplo, pode acontecer em um aplicativo de celular.

Note que qualquer um tem acesso a chave pública, o que significa que é possível mandar mensagens tanto para o cliente quanto para o servidor, mas para interceptar uma mensagem e saber seus dados seria necessário ter uma chave privada.

O processo que usaremos neste material será um pouco mais simplificado. Iremos usar somente uma chave pública e uma chave privada (ambas do servidor). Assim, a comunicação servidor-cliente se dará da seguinte forma:

- O cliente criptografa a mensagem usando a chave pública e envia para o servidor.
- Com a chave privada, o servidor descriptografa a mensagem do cliente.

Observação 3.2.3. Nesse método de criptografia, a chave pública só tem a função de criptografar e a chave privada de descriptografar. Devido as características da chave pública, alguém facilmente poderia enviar uma mensagem com más intenções, portanto existe o conceito de assinatura digital, que certifica quem está mandando a mensagem. Iremos discutir sobre assinaturas com mais detalhes ao fim do material.

3.3 Criptografia RSA: funcionamento

O processo de criptografia RSA funciona basicamente em três etapas:

1. **Pré-codificação:** Aqui serão feitas as conversões necessárias da linguagem padrão (português, no nosso caso) para uma linguagem numérica que possibilite ser utilizada para o algoritmo. Além disso, há a escolha de alguns parâmetros, os quais serão discutidos com mais detalhes em breve.

2. **Codificação:** Nesta etapa, com a mensagem escolhida, o processo de codificar será feito em blocos, ou seja, quebra-se a mensagem em partes menores e codifica-se cada pedaço separadamente. A mensagem encriptada será a junção destes blocos.

3. **Decodificação:** O receptor irá receber uma mensagem criptografada (feita na etapa anterior) e para ter acesso ao conteúdo, terá que saber a chave (nesse caso, privada).

Iremos agora descrever o passo a passo do algoritmo RSA. Vamos começar entendendo como funciona a **pré-codifi-**

cação. Para isso precisamos converter as mensagens para sequências de números. Vamos supor, por simplicidade, que as mensagens somente são compostas por letras. Escolheremos então números para representar cada letra do alfabeto. Além disso precisamos de um número para representar os espaços entre cada palavra (lembre-se que a mensagem inteira tem que estar representada como uma sequência numérica). Essa escolha pode ser feita de várias maneiras, entretanto algumas são mais viáveis que outras, como veremos adiante. Segue abaixo uma tabela de conversão:

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

Tabela 3.1: Conversão para números

Observação 3.3.1.

Iremos utilizar o número **99** quando quisermos representar um espaço entre palavras.

Exercício 3.3.2. O que aconteceria se tivéssemos escolhido 1 para representar A, 2 para representar B, 3 para representar C e assim por diante?

O exemplo a seguir irá nos ajudar a entender como a tabela funciona.

Exemplo 3.3.3. Vamos converter a frase *"Eu amo matemática"* usando a tabela 3.1.

A sequência numérica equivalente será:

14	30	99	10	22	24	99	22	10	29	14
E	U	-	A	M	O	-	M	A	T	E
22	10	29	18	12	10					
M	A	T	I	C	A					

Destacando somente os números, temos:

14309910222499221029142210291812 10

Para prosseguir, precisaremos de dois números primos distintos, digamos p e q que serão chamados de *parâmetros* e tomaremos o produto destes, digamos $n = p \cdot q$. A última etapa da pré-codificação é "quebrar" a mensagem em blocos. Vamos continuar com o exemplo anterior e quebrar a mensagem em blocos.

Observação 3.3.4. Os blocos escolhidos devem ser menores que n. Por exemplo, se $n = 11 \cdot 3 = 33$, cada bloco tem que ser menor que 33.

Exemplo 3.3.5. Vamos utilizar $p = 11$ e $q = 13$, assim $n = p \cdot q = 11 \cdot 13 = 143$. Devemos quebrar

14309910222499221029142210291812 10

em blocos menores, onde cada bloco não exceda 143. Digamos:

14 30 99 102 2 24 99 22 102 91 42 2 10 29 18 12 10

Note que nenhum bloco começa com zero (isto traria algumas dificuldades na hora de computar o algoritmo). Além disso iremos evitar deixar blocos somente com o número 1 (novamente, traria alguma dificuldade).

Com isso, terminamos esta etapa. Vejamos agora a parte principal do algoritmo, a **codificação**. Antes de codificar a mensagem, precisamos saber alguns valores. Vamos finalmente utilizar o valor de n discutido anteriormente, e, além

dele, precisaremos de um número inteiro que seja invertível módulo $\phi(n)$, ou seja, um número que vamos chamar de e tal que $\text{mdc}(e, \phi(n)) = 1$ (revise a parte de números invertíveis módulo n, será muito importante!).

Observação 3.3.6. Lembre-se que, como $n = p \cdot q$, teremos $\phi(n) = \phi(p \cdot q) = \phi(p) \cdot \phi(q) = (p-1)(q-1)$, pois p e q são primos.

Chamaremos o par (n, e) de *chave de codificação* do sistema RSA.

Denotando por b um dos blocos obtidos anteriormente, iremos escrever $C(b)$ para denotar o bloco codificado e a "receita de bolo" para codificar um bloco será a seguinte:

$$C(b) \doteq \text{resto da divisão de } b^e \text{ por } n$$

Também podemos ver $C(b)$ como a forma reduzida de b^e módulo n.

Deste modo, a mensagem codificada será a sequência dos blocos codificados.

Exemplo 3.3.7. Continuando com o exemplo anterior, temos $p = 11$ e $q = 13 \Rightarrow n = 143$ e $\phi(n) = \phi(11) \cdot \phi(13) = 10 \cdot 12 = 120$.

Precisamos de um valor para e de modo que $\text{mdc}(e, 120) = 1$. Note que $120 = 10 \cdot 12 = 2 \cdot 5 \cdot 2 \cdot 6 = 2 \cdot 5 \cdot 2 \cdot 2 \cdot 3 = 2^3 \cdot 3 \cdot 5$, e assim o menor primo que não divide 120 é 7. Portanto, iremos escolher $e = 7$ (qualquer outro valor que satisfaça $\text{mdc}(e, 120) = 1$ poderia ser escolhido, mas vamos pelo menor, por simplicidade).

Lembre-se dos blocos obtidos anteriormente: 14 30 99 102 2 24 99 22 102 91 42 2 10 29 18 12 10

Vamos codificar, por exemplo, o bloco 102. Para isso precisamos saber a forma reduzida de 102^e módulo n, neste caso 102^7 módulo 143. Mas

$102^7 \equiv (-41)^7 \equiv -41^7 \equiv (-1)(41^2)^3 \cdot 41 \equiv (-1)1681^3 \cdot 41 \equiv (-1)(11 \cdot 143 + 108)^3 \cdot 41 \equiv (-1)108^3 \cdot 41 \equiv (-1)(-35)^3 \cdot 41 \equiv 35^3 \cdot 41 \equiv 35^2 \cdot 1435 \equiv 35^2(143 \cdot 10 + 5) \equiv 35^2 \cdot 5 \equiv 35 \cdot 175 \equiv 35 \cdot 32 \equiv 1120 \equiv (143 \cdot 7 + 119) \equiv 119 \pmod{143}$

Fazendo o mesmo tipo de conta para os outros blocos, a mensagem codificada será:

PC	14	30	99	102	2	24	99	22	102
C	53	134	33	119	128	106	33	22	119
PC	91	42	2	10	29	18	12	10	
C	130	81	128	10	94	138	12	10	

Tabela 3.2: PC: Pré-codificado e C: Codificado.

Ou seja,

53 134 33 119 128 106 33 22 119 130 81 128 10 94 138 12 10

Observação 3.3.8. Repare que para alguns blocos, a codificação $C(b)$ coincide com b. Isso não é problema (desde que esses blocos não revelem informações importantes). Blocos como esses são chamados de *invariantes* pelo RSA.

Assim, se encerra a parte mais importante do algoritmo. Seguindo os passos descritos anteriormente pode-se codificar qualquer mensagem e transmiti-la com segurança. Veremos que para saber o conteúdo original, será necessário uma *chave privada*.

Observação 3.3.9. Como veremos a seguir, para decodificar a mensagem, será necessário decodificar bloco por bloco, ou seja, não podemos unir os blocos $C(b)$ e formar um novo número.

Finalmente, iremos **decodificar** a mensagem que foi codificada nos passos anteriores. Para tal, precisaremos de dois

números, o primeiro é o n, valor já conhecido. Já o segundo será denotado por d, onde $d \doteq$ "inverso de e módulo $\phi(n)$".

O par (n, d) é chamado de *chave de decodificação*.

Agora já temos todos os "ingredientes" para decodificar nossa mensagem, então denotando por a um bloco codificado, denotaremos por $D(a)$ o resultado da decodificação e a "receita de bolo" dessa vez será:

$$D(a) \doteq \text{resto da divisão de } a^d \text{ por } n$$

Ou equivalentemente, $D(a)$ é a forma reduzida de a^d módulo n.

Com isso, o processo de criptografia RSA está completo. Vamos ver um exemplo de como decodificar uma mensagem.

Exemplo 3.3.10. Estamos utilizando $n = 143$ e $e = 7$. Precisamos calcular d. Para isso vamos utilizar o *algoritmo euclidiano estendido*, isto é, vamos dividir $\phi(143) = 120$ por 7. Iremos obter como resultado:

$$\begin{aligned} 120 &= 7 \cdot 17 + 1 \\ 1 &= 120 + (-17) \cdot 7 \end{aligned}$$

Ou seja, -17 seria o valor de d, mas como queremos um valor positivo iremos tomar uma valor equivalente módulo 120. Basta pegar o número $(-17) + 120 = 103$, portanto $d = 103$.

Relembremos a mensagem codificada:

53 134 33 119 128 106 33 22 119 130 81 128 10 94 138 12 10

Vamos escolher um bloco e decodificar, tome por exemplo o bloco 119. Sua decodificação será:

$$119^d \pmod{n} \equiv 119^{103} \pmod{143} \equiv 102 \pmod{143}$$

Se você voltar aos exemplos anteriores, verá que o resultado está correto, basta repetir o processo para todos os blocos e terá a mensagem original.

Observação 3.3.11. Note que a última conta que fizemos seria muito complicada com lápis e papel, e, portanto, é necessário o auxílio de um computador.

3.3.1 Exercícios

Exercício 3.3.12. Tomando $p = 11$ e $q = 3$, siga os passos descritos no capítulo para codificar o seu nome. Lembre-se de usar a tabela para conversão de letras em números e ignore acentos. Além disso escolha um valor para e conveniente. Após a codificação, decodifique a mensagem e confira o resultado.

Exercício 3.3.13. Sabendo-se que $n = 3552377$ é igual ao produto de dois números primos e que $\phi(n) = 3548580$; fatore n.

Exercício 3.3.14. A chave pública utilizada pelo Banco de Toulouse para codificar suas mensagens é a seguinte: $n = 10403$ e $e = 8743$. Recentemente os computadores do banco receberam, de local indeterminado, a seguinte mensagem:

4762 - 8214 - 9372 - 9009 - 4453 - 8198

O que diz a mensagem mandada ao Banco Toulouse?

Exercício 3.3.15. A mensagem 6355 - 5075 foi codificada pelo método RSA usando a senha $n = 7597$ e $e = 4947$. Além disso, sabe-se que $\phi(n) = 7420$. Decodifique a mensagem.

3.4 Por que funciona?

Agora que sabemos como RSA funciona, nos resta entender o por que dele funcionar. Queremos que um bloco de uma mensagem quando codificado, possa ser decodificado e volte para o bloco original pelo processo exposto. Ou seja queremos aplicar a codificação b^e e a decodificação $(b^e)^d$ e obter o resultado original, matematicamente:

$$b^{ed} = b$$

observa-se que como $b^{ed}, b < n$, demonstrar essa igualdade é análogo a demonstrar

$$b^{ed} \equiv b \ (mod\ n)$$

Inicialmente , devemos observar que

$$ed \equiv\ 1\ (mod\ \phi(n)) \implies ed = 1 + k\phi(n)\ para\ k \in \mathbb{Z},\ então$$

$$ed = 1 + k(p-1)(q-1) \implies b^{ed} \equiv b.(b^{p-1})^{k(q-1)}\ (mod\ p)$$

Ademais, supondo que p não divide b podemos usar o teorema de Fermat para obter $b^{p-1} \equiv b\ (mod\ p)$. O caso onde p divide b é simples dado que p é primo, supondo que p divide b temos $b \equiv 0\ (mod\ p)$ e então $b \equiv b^{ed}\ (mod\ p)$. Como não especificamos p no raciocínio anterior podemos realiza-lo analogamente para q e obter $b \equiv b^{ed}\ (mod\ q)$, ou seja $b^{eq} - b$ é divisível por p e por q, e como $mdc(p, q) = 1$ pois , são primos distintos , $b^{eq} - b$ é divisível por pq, assim $b^{eq} - b \equiv 0\ (mod\ n)$ e por fim $b^{eq} \equiv b\ (mod\ n)$.

3.5 A Implementação

Em geral , dada a chave pública , queremos que só seja possível decifrar a mensagem com a chave privada.

Na Criptografia RSA ao sabermos a chave pública , teremos um primo muito grande e um número aleatório coprimo com $\phi(n)$. A dificuldade é a fatoração do número primo, o qual tem um custo computacional grande.

Portanto, escolhendo dois primos grandes, é impossível os calcular somente obtendo a sua multiplicação, mas também é necessário garantir que seja impossível encontrar "e", já que no processo de descriptografar , é o único número realmente necessário. Tome d o inverso multiplicativo de "e", mod n ou seja $de \equiv 1 \mod \phi(n)$ porém para isso é necessário calcular $\phi(n)$ e para isso fatorar n .

Em geral , é quase impossível quebrar RSA, sem conseguir fatorar números muito grandes. Porém, há alguns problemas na segurança do RSA quando é feita uma assinatura digital, a qual já foi quebrada.

3.6 Problemas de segurança

A segurança de um sistema RSA não se da somente pela chave privada. Há outros problemas que podem comprometer a segurança, dentre eles podemos citar:

- **Necessidade de primos grandes**: Por mais que um computador não consiga fatorar primos com muitos dígitos, é preciso garantir que os primos escolhidos tenham pelo menos 100 dígitos e isso ocasiona um gasto computacional. Assim, leva-se em consideração o tempo necessário para a criação de uma chave privada segura.

- **Chave privada pequena**: Além de garantir dois primos relativamente grandes, é necessário garantir um d

suficientemente grande.

- **Escolha de primos**: Por mais que primos grandes resolvam todos os problemas computacionais, há um problema na escolha dos primos. Se por exemplo, p e q são primos grandes mas $|p - q|$ não , então é fácil fatorar n pelo algoritmo de Fermat.

Por mais que existam problemas de segurança na criptografia RSA ela é uma das mais usadas diariamente, dado que haja uma boa escolhas de primos e uma chave privada segura, teremos uma criptografia segura.

Em geral, podemos dizer que quebrar RSA é análogo desvendar a fatoração de grandes números computacionalmente.

3.7 Testes de Primalidade

Para podermos aplicar a criptografia em computadores ,precisaremos ter uma forma de testar se um número muito grande é primo, assim podemos criar um método computacional para encontrar esses números. Para isso, iremos enunciar o teste de miller e aplica-lo a um teste e torna-lo aplicável à criptografia RSA.

O teste de Miller: Queremos testar se n é divisível por um inteiro b tal que $1 < b < n - 1$, para isso, seguiremos o algoritimo:

- divida n-1 sucessivamente por 2 até encontrar um ímpar q e k tais que n-1 = $2^k q$
- defina i = 0 e r = resto de b^q por n
- se i $\geq$ 0 e r = 1 ou r = n-1 , então o teste é inconclusivo

- incremente i de 1 e substitua r pelo resto da divisão de r^2 por n
- se $i < k$ volte à etapa 3 . se não n é composto.

O teste definitivo: Seja $n > 0$ um inteiro, e

$$n - 1 = p_1^{e_1} ... p_r^{e_r}$$

onde p_i para cada i = 1 , ... r são primos. Se, para cada i, existem $2 \leq b_i \leq n - 1$ que satisfaçam

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

$$b_i^{\frac{n-1}{p_i}} \not\equiv 1 (mod \ n)$$

Então, para testar se n é primo, iremos utilizar este teste e o de Miller visto anteriormente. Esta estratégia consiste em 3 etapas:

1. Verifique se n é divisível por um primo $p < 5000$
2. Se n não é divisível por estes primos, aplique o teste de Miller a n usando como base os primeiros 10 primos
3. Supondo que o teste de Miller não concluiu nada com todas estas bases, aplique o teste acima a n.

Entretanto, o gasto computacional para cada caso particular de n é grande.

O próximo processo se resume a diminuir a quantidade de voluntários entre dois números: x e $x + 10^4$. Então, para adaptar a estratégia e poder testar o maior número de primos no menor tempo possível, iremos pegar os números no intervalo $(x + 10^4)$, e eliminar todos os inteiros ímpares que são divisíveis por primos menores que 5000, então aplicaremos

os passos 2 e 3 aos que sobraram. O processo continua até encontrar um primo Um dos problemas é o caso de não ter nenhum número primo. Em decorrência disso, é necessário garantir de que haja pelo menos um primo neste intervalo, para qual sera usada a função $\pi(x)$.

Para x muito grande, podemos tomar ϵ pequeno e verificar quanto vale $\pi(x+\epsilon) - \pi(x)$, ou seja, uma aproximação para o número de primos de x até x+ ϵ.

Das propriedades do logaritmo e do teorema dos números primos, temos que $\pi(x+\epsilon) - \pi(x)$ é aproximadamente igual

$$\frac{x+\epsilon}{logx + log(1+x^{-1}\epsilon)} - \frac{x}{logx}$$

e supondo $x^{-1}\epsilon$ pequeno (o que podemos garantir já que x é muito grande e ϵ muito pequeno), podemos substituir $log(1+x^{-1}\epsilon)$ por 0.

Logo, temos uma aproximação para a quantidade de primos no intervalo $\frac{\epsilon}{logx}$. Com isso, podemos fazer uma aproximação para números de 127 dígitos, temos

$$\frac{10^4}{log10^{127}} = 34$$

Desse modo, conseguimos garantir que entre 10 000 números de aproximadamente 127 dígitos, encontramos em média 37 primos.

Exercício 3.7.1. Use o teste para verificar que 991 é um primo

3.8 Assinatura digital

Digamos que uma empresa se comunique com um banco usando a criptografia RSA, podendo requisitar diversas atividades para o banco, como fazer uma transferência.

Decifrar que a empresa está pedindo uma transferência ao banco, como já vimos, é muito difícil.

Entretanto, é muito fácil um hacker, por exemplo, enviar uma mensagem ao banco no lugar da empresa. Para isso existe a assinatura digital.

A ideia é mudar a mensagem codificada de forma que quem está recebendo possa confirmar o remetente. Assim, vamos denotar, e_e, d_e e e_b ,d_b nossos "e" e "d" da empresa e do banco respectivamente.

Suponhamos que a empresa queira mandar um pedido de transferência em uma mensagem m, a qual normalmente teria a codificação m^{e_b}.

Note que para criar uma assinatura é necessário que a empresa antes de codificar a mensagem com a chave publica do banco faça a operação m^{d_e}, e entrega ao banco $s = (m^{d_e})^{e_b}$.

Desta forma, o banco consegue descriptografar a mensagem fazendo $(s^{d_b})^{e_e}$. Ou seja, primeiramente o banco iria descriptografar a mensagem, que não faria sentido pois teria m^{d_e} e então, com e_e que é público, iria encontrar uma mensagem legível.

Como somente a empresa tem d_e então o banco saberia que a mensagem veio dela, caso o banco aplicasse e_e e encontra-se uma mensagem estranha ou ilegível, ficaria claro que a mensagem não veio da empresa. Assim é possível garantir o remetente da mensagem, embora o mal uso leve a falta de segurança.

Em 1995 foi descoberto que, com base na velocidade que o sistema demorava para confirmar a assinatura com mensagens de tamanhos ligeiramente diferentes, era possível desvendar a chave privada.

Exercício 3.8.1. Tomando novamente p = 11 e q = 3 ,codifique seu nome, desta vez você também tera que realizar uma assinatura para um destinatário com e = 7

3.9 Aplicação

Finalmente temos um sistema de criptografia e garantimos que ele funciona, agora o que nos resta é como aplica-lo.

Já vimos que RSA é usado em bancos, mas afinal como isso acontece? Em geral, numa situação formal se utilizaria computadores, sistemas potentes e complexos para poder criar números primos e assim uma chave privada mais segura. Evidentemente, o tempo gasto para criar as chaves depende do quão seguro tem que ser o sistema.

Uma troca de mensagem entre amigos, por exemplo, que estão brincando não precisa ter uma chave com primos com mais de 3 dígitos. Enquanto que um banco que necessita proteger as informações dos seus clientes necessita de muito tempo para criar uma chave privada segura.

Assim há maneiras, como as citadas neste capitulo, de aplicar RSA em computadores. Formas de testar primos, e fazer calculos das chaves são aprimoradas a cada dia, e com isso a facilidade de fatoração de primos também, o que da tanta importancia aos estudos em cima do assunto. E além da escolha dos primos, temos outras adaptações feitas do papel para uma máquina, por exemplo, normalmente é escolhido de forma aleatória dentro dos requisitos necessários. A escolha da representação numérica de cada letra também é feita com outras linguagens, como por exemplo ASCII.

Por fim, temos diversas maneiras de se aplicar criptografia RSA em computadores, e estudos matemáticos importantes para aprimorar a segurança deste metodo de codificação.

Referências Bibliográficas

[1] COUTINHO, S. C. **Números Inteiros e Criptografia RSA**. Rio de Janeiro, IMPA, 2014.
[2] COUTINHO, S. C. **Criptografia**. Rio de Janeiro, IMPA, 2015.
[3] COMPARITECH. **What is RSA encryption and how does it work?**.
[4] MARINHO, T. **Criptografia Assimétrica RSA**, 2017.
[5] MARQUES, D. **Teoria dos Números Transcendetes**. SBM, 2013. Rio de Janeiro. 1ª edição.
[6] HEFEZ, A. **Curso de Álgebra**, Volume 1, IMPA, 2016.