

O Algoritmo Quântico de Shor

Paulo Alexandre Atkinson

Supervisora: Cristina Gomes Fernandes

MAC-5701 - TÓPICOS EM CIÊNCIA DA COMPUTAÇÃO

10 DE DEZEMBRO DE 2001.

Notação

Básica

\mathbb{N}	números naturais: $\mathbb{N} = \{1, 2, 3, \dots\}$.
\mathbb{Z}	números inteiros: $\mathbb{Z} = \{0, \pm n : n \in \mathbb{N}\}$.
$\mathbb{Z}_{\geq \varepsilon}$	números inteiros maiores ou iguais a ε . Ex: $\mathbb{Z}_{\geq 0} = \{0, n : n \in \mathbb{N}\}$.
\mathbb{Q}	números racionais: $\mathbb{Q} = \{\frac{a}{b} : a, b \in \mathbb{Z}, b \neq 0\}$.
\mathbb{R}	números reais: $\mathbb{R} = \{n + 0, d_1 d_2 d_3 \dots : n \in \mathbb{Z}, d_i \in \{0, 1, \dots, 9\}, \text{ pelo menos um } d_i \neq 9\}$.
\mathbb{C}	números complexos: $\mathbb{C} = \{a + bi : a, b \in \mathbb{R}, i = \sqrt{-1}\}$.

Capítulo 1

$\mathcal{P} = (\mathcal{I}, \mathcal{R}, \mathcal{X})$ \mathcal{P} é um problema computacional, \mathcal{I} é um conjunto de instâncias, \mathcal{R} é um conjunto de soluções e \mathcal{X} é uma função que a cada elemento de \mathcal{I} associa um elemento de \mathcal{R} .

$\lg \beta$ logaritmo de β na base 2.

$\log \beta$ logaritmo de β em alguma base (usado quando o valor preciso da base é irrelevante).

$\ln \beta$ logaritmo de β na base e .

Capítulo 2

$a \mid b$ a divide b ; b é múltiplo de a .

$a \nmid b$ a não divide b ; b não é múltiplo de a .

$\lfloor x \rfloor$ chão de x : maior inteiro menor ou igual a x .

$\lceil x \rceil$ teto de x : menor inteiro maior ou igual a x .

π_p números primos: $\pi_p = \{n : n \in \mathbb{Z}, n > 1, a \mid n \text{ se e somente se } |a| \in \{1, n\}\} = \{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, \dots\}$.

$x \bmod n$ resto da divisão inteira de x por n : $x \bmod n = x - n \lfloor \frac{x}{n} \rfloor$.

$x \equiv y \bmod n$ os restos das divisões inteiras de x e de y por n são iguais; x e y são ditos congruentes módulo n .

$x \not\equiv y \bmod n$ os restos das divisões inteiras de x e de y por n são diferentes; x e y são ditos não-congruentes módulo n .

$\text{mdc}(a, n)$ máximo divisor comum entre a e n .

$\text{mmc}(a, n)$ mínimo múltiplo comum entre a e n .

\mathbb{Z}_n conjunto dos inteiros módulo n : $\mathbb{Z}_n = \{0, 1, \dots, n-1\}$.

\mathbb{Z}_n^* conjunto dos elementos z de \mathbb{Z}_n não nulos, tais que $\text{mdc}(z, n) = 1$.

$r = \text{ord}_n(a)$ r é o menor natural k tal que $a^k \equiv 1 \bmod n$; r é dito a ordem de a módulo n , ou também a ordem de a em \mathbb{Z}_n^* .

$x^k \bmod n$	exponenciação modular.
\mathcal{G}, \mathcal{H}	grupos.
$\mathcal{G} = (G, \star, \ast, e)$	notação de um grupo;
	G : conjunto de elementos de grupo;
	\star : operação binária;
	\ast : operação de inversão;
	e : elemento identidade.
$\mathcal{G} = (G, +, -g, 0)$	grupo aditivo.
$\mathcal{G} = (G, \cdot, g^{-1}, 1)$	grupo multiplicativo.
$\mathcal{H} \leq \mathcal{G}$	\mathcal{H} é um subgrupo de \mathcal{G} .
$g \star H$	<i>coset</i> do subgrupo \mathcal{H} do grupo \mathcal{G} .
$(g_1 \star H) \star (g_2 \star H) = (g_1 \star g_2) \star H$	operação binária entre <i>cosets</i> de $\mathcal{H} \leq \mathcal{G}$.
$g_1 \star H = g_2 \star H$	g_1 é congruente à g_2 módulo H .
\mathcal{G}/\mathcal{H}	grupo fator; grupo quociente.
$\mathbb{Z}/(n\mathbb{Z})$	$\mathbb{Z}/(n\mathbb{Z}) = (z + n\mathbb{Z}, +, -z + nz, n\mathbb{Z})$, $z \in \mathbb{Z}, n \in \mathbb{N}$: grupo aditivo \mathbb{Z}_n .
$z + n\mathbb{Z}$	<i>coset</i> do subgrupo $(n\mathbb{Z})$ do grupo \mathbb{Z} .
$z_1 + n\mathbb{Z} = z_2 + n\mathbb{Z}$	z_1 e z_2 são congruentes módulo $n\mathbb{Z}$; também denotado $z_1 \equiv z_2 \bmod n$.
$\mathbb{Z}_n = \{0 + n\mathbb{Z}, 1 + n\mathbb{Z}, \dots, n - 1 + n\mathbb{Z}\}$	uma representação do grupo aditivo \mathbb{Z}_n .
$\mathbb{Z}_n = \{0, 1, 2, \dots, (n - 1)\}$	outra representação do grupo aditivo \mathbb{Z}_n .
$(z_1 + n\mathbb{Z})(z_2 + n\mathbb{Z}) = z_1 z_2 + n\mathbb{Z}$	produto de <i>cosets</i> de $n\mathbb{Z}$.
$(z + n\mathbb{Z})^{-1} = z^{-1} + n\mathbb{Z}$	inversão de <i>cosets</i> de $n\mathbb{Z}$.
\mathbb{Z}_n^*	grupo multiplicativo \mathbb{Z}_n^* , denotado $\mathbb{Z}_n^* = (\{z + n\mathbb{Z} : z \in \mathbb{Z}_n, \text{mdc}(z, n) = 1\}, \cdot, z^{-1} + n\mathbb{Z}, 1 + n\mathbb{Z})$.
$G = \{0g, g, 2g, \dots, (m - 1)g\}$	grupo cíclico aditivo de tamanho finito, gerado por g .
$G = \{g^0, g^1, g^2, \dots, g^{m-1}\}$	grupo cíclico multiplicativo de tamanho finito, gerado por g .
$\text{ord}_n(z)$	ordem de $z + n\mathbb{Z}$ em \mathbb{Z}_n^* ; menor natural r tal que $(z + n\mathbb{Z})^r = 1 + n\mathbb{Z}$; também denotado $z^r \equiv 1 \bmod n$.
$\mathcal{G} \cong \mathcal{H}$	o grupo \mathcal{G} é isomorfo ao grupo \mathcal{H} .
Capítulo 3	
H_n	espaço de Hilbert de dimensão n .
$ \psi\rangle$	vetor de estado.

Capítulo 4

$\phi(n)$	função ϕ de Euler, definida na pág. 27.
$ x_1\rangle x_2\rangle$	par de registradores quânticos nos estados x_1 e x_2 .
$ \mathbb{Z}_n^* $	cardinalidade de \mathbb{Z}_n^* , número de elementos de grupo de \mathbb{Z}_n^* .
z_{p_i}	elemento gerador do grupo cíclico $\mathbb{Z}_{p_i}^*$
$\mathcal{G} \cong \mathcal{H}$	\mathcal{G} é isomorfo a \mathcal{H} .
χ	caractere.
$\widehat{\mathcal{G}}$	grupo dual de \mathcal{G} . $\widehat{\mathcal{G}} = (\chi(G), \chi_1(g)\chi_2(g), \chi(g)^{-1}, \chi_0)$
$\widehat{\mathbb{Z}}_n$	grupo dual de \mathbb{Z}_n . $\widehat{\mathbb{Z}}_n = \{\chi_y(0), \chi_y(1), \chi_y(2), \dots, \chi_y(n-1)\}$ ou $\widehat{\mathbb{Z}}_n = \{\chi_0, \chi_y(z)^1, \chi_y(z)^2, \dots, \chi_y(z)^{n-1}\}$.

Prefácio

O algoritmo quântico probabilístico para fatoração de inteiros em tempo polinomial publicado por Peter W. Shor em 1994 [15, 14] é considerado o primeiro algoritmo quântico combinando relevância prática e eficiência. Dado um inteiro n com pelo menos dois divisores primos distintos, o algoritmo de Shor calcula um divisor não-trivial de n em $O(\log^3 n)$.

O algoritmo de Shor foi projetado para um modelo formal de computação que se baseia nas leis da mecânica quântica. A pouca eficiência encontrada na simulação da dinâmica de sistemas quânticos com o emprego do modelo tradicional de computação [7] deu origem à idéia de que um modelo computacional quântico poderia ser mais eficiente que o modelo tradicional. O algoritmo de Shor é uma evidência de que, sob certas circunstâncias, o modelo computacional quântico pode superar de fato o modelo tradicional. Entretanto ainda não está claro quais são essas circunstâncias, quais são as limitações do modelo quântico e para quais problemas ele é adequado. Essas questões têm sido objeto de intensa pesquisa científica.

Neste texto introdutório são apresentados o problema de fatoração de inteiros em primos¹, a formalização matemática do modelo computacional quântico e o algoritmo quântico de Shor propriamente dito. Os pré-requisitos dos assuntos tratados são noções de teoria dos números, álgebra linear, operadores lineares em espaços de Hilbert, transformada de Fourier e teoria da complexidade de algoritmos. Na medida do possível esses assuntos são cobertos com detalhamento suficiente para que o texto seja auto-contido.

Noções elementares de complexidade de algoritmos são apresentadas no capítulo 1, que tem como objetivo esclarecer quais algoritmos são considerados eficientes dentro de três modelos computacionais: determinístico, probabilístico e quântico. Para tanto, são apresentadas definições algébricas elementares de problemas computacionais e máquinas de Turing, que são exploradas apenas o suficiente para caracterizar classes de complexidade às quais pertencem problemas considerados tratáveis e com isso permitir uma definição básica de algoritmos eficientes.

O capítulo 2 trata do problema da fatoração de inteiros em primos e tem por objetivo deixar claro qual a técnica empregada em sua solução e quais os fundamentos matemáticos em que se baseia essa técnica. Como nos demais capítulos, é apresentado num primeiro momento todo o esquema de solução, acompanhado de exemplos. Em seguida, são discutidos os fundamentos teóricos, que no caso do capítulo 2 são o conceito de grupo e equações modulares.

¹O problema de fatoração de inteiros em primos é considerado ainda mais difícil que o problema de determinar se um dado número é primo [16].

No capítulo 3, o modelo computacional quântico é formulado por meio de operadores lineares em espaços de Hilbert de dimensão finita, sobre números complexos. Esse modelo é suficiente para representar certas características físicas de um sistema mecânico quântico, tais como o fenômeno de superposição de estados, o efeito da medição sobre o sistema, as previsões não-determinísticas, entre outras. O objetivo deste capítulo é explicar como esse modelo permite implementar as características do bit quântico, ou qubit, utilizado na montagem de registradores quânticos.

Detalhar o núcleo do algoritmo de Shor é o objetivo do capítulo 4. Para tanto, são discutidas a transformada quântica de Fourier e o modo como essa transformada é empregada no algoritmo quântico de Shor. Uma análise de complexidade e um sumário sintetizando os assuntos tratados ao longo do texto encerram o capítulo.

A cada dia, a viabilidade da computação quântica se torna mais próxima. Seu potencial na solução em tempo polinomial de certos tipos de problemas computacionais complexos foi demonstrado por Deutsch e Jozsa [5], Shor [15], Brassard e Hoyer [3] e outros pesquisadores. Mesmo que se venha a demonstrar que o modelo computacional quântico, da mesma maneira que o modelo tradicional, é insuficiente para resolver problemas NP-completos em tempo polinomial, ou mesmo para obter esquemas de aproximação polinomial para problemas de otimização APX-completos, a aceleração no processamento, obtida com esse modelo, poderá justificar por si o investimento em computação quântica e o estudo e projeto de algoritmos quânticos.

Sumário

Notação	i
Prefácio	iv
1 Noções de Complexidade Computacional	1
1.1 Algoritmos e Modelos de Computação	1
1.2 Algoritmos Eficientes	2
1.3 Algoritmos de Decisão Eficientes	4
1.4 Algoritmos Probabilísticos de Decisão	4
1.5 Máquinas de Turing	5
2 Fatoração de Inteiros	7
2.1 Fatoração de Inteiros em Primos	7
2.1.1 Identificação de Fator Primo	8
2.1.2 Exemplos de Fatoração	10
2.1.3 Exercícios	11
2.2 O Grupo Aditivo \mathbb{Z}_n	13
2.2.1 Noções de Teoria dos Grupos	13
2.2.2 Grupo Aditivo \mathbb{Z}_n	14
2.2.3 Grupo Multiplicativo \mathbb{Z}_n^*	15
2.2.4 Grupos Cíclicos	17
2.2.5 Conceito de Ordem em \mathbb{Z}_n^*	19
2.2.6 Decomposição de \mathbb{Z}_n^* em Subgrupos Cíclicos	20
3 Modelo Computacional Quântico	23
3.1 Introdução	23
3.2 Visão geral	23
3.3 Mecânica Quântica em espaços de Hilbert	24
4 Algoritmo de Shor	25
4.1 Introdução	25
4.2 Visão geral	25
4.3 Determinação da Ordem	27
4.3.1 Definição do Problema de Determinação da Ordem	27

4.3.2	Probabilidade de Obtenção de uma Ordem Adequada	27
4.3.3	Ordem como Período da Função $f(k) = a^k \bmod n$. . .	29
4.4	Transformada de Fourier Discreta	29
4.4.1	Caracteres	30
4.4.2	Transformada de Fourier Discreta Clássica	33
4.4.3	Transformada de Fourier e Periodicidade	34
Bibliografia		35
Índice		36

Capítulo 1

Noções de Complexidade Computacional

1.1 Algoritmos e Modelos de Computação

Um processo é dito computacional¹ quando suas transições de estado podem ser descritas por meio de um modelo de computação, que é um objeto conceitual com precisa definição matemática. O modelo elementar de computação é a máquina de Turing básica que, embora simples, acredita-se capaz de realizar todo e qualquer procedimento algorítmico, isto é, que envolva sequências finitas de instruções ou operações com objetivo definido, em geral a solução de um problema computacional. Se um procedimento for algoritmicamente computável, então ele poderá ser realizado por uma máquina de Turing básica. Essa hipótese é conhecida como a tese de Church-Turing.

Outros modelos de computação fundamentados na máquina de Turing básica têm por objetivo facilitar a implementação e análise de tipos específicos de procedimento algorítmico, porém a capacidade desses modelos de atingir o objetivo de cada procedimento é igual à da máquina de Turing básica. Neste trabalho estamos interessados em três modelos de computação em particular: a máquina de Turing básica, BTM, a máquina de Turing probabilística, PTM, e a máquina de Turing quântica, QTM. Nosso objetivo é deixar claras as diferenças entre esses modelos, e não utilizá-los como instrumento para descrição de algoritmos. Após algumas definições elementares serão tratados esses três modelos de computação.

Um alfabeto é um conjunto finito de elementos denominados caracteres, que quando concatenados formam palavras. As palavras são o meio usado para descrever objetos e ações do procedimento algorítmico, tais como números e operações aritméticas, vetores e grafos, entre

¹Esta seção tem como base o apêndice E de [6].

outros. Por exemplo, o grafo $G(V, E)$, onde $V = \{a, b, c, d\}$ e $E = \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{c, d\}\}$, pode ser representado pela palavra

$$(a, b, c, d, \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{c, d\}\}).$$

Um vetor racional c indexado por um conjunto finito E pode ser representado por uma sequência de pares (e, c_e) , onde e é um elemento de E . Por exemplo, a palavra

$$((\{a, b\}, -2), (\{a, c\}, 12), (\{a, d\}, 0), (\{b, c\}, \frac{3}{2}), (\{c, d\}, 7))$$

codifica um vetor indexado pelo conjunto das arestas do grafo acima. O tamanho de uma palavra w , denotado $|w|$, é o número de caracteres de w , contando-se repetições.

Como estaremos lidando com algoritmos numéricos, o tamanho de palavras representando números inteiros e racionais merece especial atenção. O tamanho de um inteiro α é essencialmente $\log \alpha$ e o tamanho de um racional $\frac{\alpha}{\beta}$ é essencialmente $\log \alpha + \log \beta$.

1.2 Algoritmos Eficientes

Um problema computacional é definido como uma tripla $\mathcal{P} = (\mathcal{I}, \mathcal{R}, \mathcal{X})$, onde \mathcal{I} é um conjunto de instâncias, ou casos específicos, com representação finita, \mathcal{R} é um conjunto de respostas, também com representação finita e $\mathcal{X} : \mathcal{I} \rightarrow \mathcal{R}$ é uma função que a cada instância associa uma resposta. Uma solução para o problema computacional \mathcal{P} é um algoritmo \mathcal{A} , que computa \mathcal{X} . Se existir \mathcal{A} , solução de \mathcal{P} , então existe uma máquina de Turing tal que, dada uma instância I adequadamente codificada em palavras compostas por caracteres pertencentes ao alfabeto dessa máquina de Turing, esta devolve, após uma sequência de mudanças de estado correspondentes ao algoritmo \mathcal{A} , a resposta R associada à instância I , também codificada. Certas máquinas de Turing, denominadas *universais* são capazes de simular outras máquinas de Turing. Nesse caso, além de fornecer a instância I do problema computacional é necessário fornecer o algoritmo \mathcal{A} devidamente codificado.

Dois critérios considerados relevantes para a análise da eficiência de um algoritmo estão associados ao número máximo de transições de estado e à quantidade máxima de posições de memória utilizada num dado momento do processo computacional. São eles a complexidade de tempo de computação e a complexidade de espaço de computação. Neste trabalho ficaremos restritos à complexidade de tempo². Seja $T_{\mathcal{A}}(I)$ o número de instruções elementares que o algoritmo \mathcal{A} emprega para devolver $\mathcal{X}(I)$, de acordo com o modelo

²A complexidade de espaço tem como limitante superior a complexidade de tempo.

computacional adotado. A complexidade de tempo, também denominada tempo de pior caso, é a função $T_{\mathcal{A}}(n) = \max(T_{\mathcal{A}}(I) : |I| \leq n)$. Em geral é difícil calcular exatamente a complexidade de tempo. Assim, nos satisfazemos em determinar seu comportamento assintótico, definido com base na seguinte relação binária:

Definição 1.2.1 (Crescimento Assintótico). Seja \preceq a relação binária entre funções $f : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$ e $g : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$ definida como $f \preceq g$ se existem $n_0, c \in \mathbb{N}$ tais que $f(n) \leq cg(n)$ para todo $n > n_0$.

A relação \preceq pode ser usada na definição de limitantes assintóticos:

- Limitante superior: se $T_{\mathcal{A}}(n) \in \{f : f \preceq g\}$ é dito que³ $T_{\mathcal{A}}(n) = O(g(n))$
- Limitante inferior: se $T_{\mathcal{A}}(n) \in \{f : g \preceq f\}$ é dito que $T_{\mathcal{A}}(n) = \Omega(g(n))$
- Limitante justo superior e inferior: Se $T_{\mathcal{A}}(n) = O(g(n))$ e $T_{\mathcal{A}}(n) = \Omega(g(n))$ é dito que $T_{\mathcal{A}}(n) = \Theta(g(n))$ ⁴.

Em geral procura-se estabelecer um limitante assintótico justo $\Theta(g)$ para $T_{\mathcal{A}}(n)$, entretanto na verificação de eficiência com base no tempo de pior caso nos contentamos em estabelecer um limitante superior $O(g)$, que irá determinar uma qualificação para o algoritmo \mathcal{A} . Por exemplo:

$T_{\mathcal{A}}(n) = O(n^k), k \in \mathbb{Z}_{\geq 0}$	\mathcal{A} é dito de complexidade polinomial;
$T_{\mathcal{A}}(n) = O(n^{c \log \log n}), c \in \mathbb{Q}_{\geq 1}$	\mathcal{A} é dito de complexidade superpolinomial;
$T_{\mathcal{A}}(n) = O(a^{\sqrt[n]{n \log n}}), a \in \mathbb{Z}_{>1}, c \in \mathbb{Q}_{>1}$	\mathcal{A} é dito de complexidade subexponencial;
$T_{\mathcal{A}}(n) = O(a^{cn}), a \in \mathbb{Z}_{>1}, c \in \mathbb{Q}_{>1}$	\mathcal{A} é dito de complexidade exponencial.

Um algoritmo \mathcal{A} é considerado eficiente quando tem complexidade polinomial, ou seja, quando $T_{\mathcal{A}}(n) = O(n^k)$, para algum k em $\mathbb{Z}_{\geq 0}$. Conforme mencionado, essa qualificação depende do modelo de computação adotado⁵. Vimos também que se existe um algoritmo que resolve um problema computacional, então existe pelo menos uma máquina de Turing Básica que implementa esse algoritmo. Além disso, se \mathcal{A} tem complexidade polinomial quando considerado como modelo de computação uma máquina de Turing, então essa máquina será igualmente polinomial.

³Com claro abuso de notação nos três casos.

⁴Note que \preceq é transitiva e reflexiva, e que na definição de $\Theta(g(n))$ fica clara a inclusão da propriedade de simetria, portanto $\Theta(g)$ é uma classe de equivalência.

⁵Por exemplo, no modelo probabilístico certos algoritmos são considerados eficientes quando o valor esperado do tempo de execução é $O(n^k)$.

1.3 Algoritmos de Decisão Eficientes

Um problema computacional de decisão é uma tripla $\mathcal{P} = (\mathcal{I}, \mathcal{R}, \mathcal{X})$, onde $\mathcal{R} = \{0, 1\}$. Um problema concreto de decisão é uma quádrupla $\mathcal{P} = (\mathcal{I}, \mathcal{R}, \mathcal{C}, \mathcal{X})$ onde $\mathcal{R} = \{0, 1\}$ e $\mathcal{C} : \mathcal{I} \rightarrow A^*$, onde A^* é a linguagem formada por todas as possíveis palavras de um alfabeto A composto por pelo menos dois caracteres e \mathcal{C} é uma *codificação* que a cada instância associa uma palavra pertencente à linguagem A^* . Se I é uma instância, então $|\mathcal{C}(I)|$ é o tamanho de I na codificação \mathcal{C} .

Um algoritmo \mathcal{A} resolve um problema de decisão concreto \mathcal{P} em tempo $T_{\mathcal{A}}(n)$ se, para todo I tal que $|\mathcal{C}(I)| \leq n$, o tempo do algoritmo para determinar $\mathcal{X}(I)$ é $\leq T_{\mathcal{A}}(n)$.

Os problemas de decisão concretos para os quais existe algoritmo (máquina de Turing básica) de tempo polinomial formam a classe de complexidade **P**. Uma função $\mathcal{X} : A^* \rightarrow A^*$ é computável em tempo polinomial se existe algoritmo \mathcal{A}_0 que computa \mathcal{X} tal que $T_{\mathcal{A}}(n) = O(n^k)$ para algum $k \in \mathbb{Z}_{\geq 0}$.

1.4 Algoritmos Probabilísticos de Decisão

Definição 1.4.1 (problema computacional de decisão). Um problema computacional de decisão é definido como uma tripla $\mathcal{P} = (\mathcal{I}, \mathcal{R}, \mathcal{X})$, onde \mathcal{I} é um conjunto de instâncias I com representação finita, $\mathcal{R} = \{0, 1\}$ é o conjunto de respostas e $\mathcal{X} : \mathcal{I} \rightarrow \mathcal{R}$ é uma função que a cada instância associa uma resposta sim (1), ou não (0). Uma solução para o problema computacional de decisão \mathcal{P} é um algoritmo \mathcal{A} , que computa \mathcal{X} .

Um algoritmo determinístico \mathcal{A} resolve o problema de decisão \mathcal{P} em tempo $T_{\mathcal{A}}(n)$ se, para todo I tal que $|I| \leq n$, o tempo do algoritmo para determinar $\mathcal{X}(I)$ é $\leq T_{\mathcal{A}}(n)$.

Definição 1.4.2 (probabilidade de erro de decisão). Seja $I \in \mathcal{I}$ uma instância de um problema de decisão \mathcal{P} . Seja $R \in \{0, 1\}$ a resposta associada a I por \mathcal{X} . Assumindo que a probabilidade de que o algoritmo \mathcal{A} , que computa \mathcal{X} possa errar a decisão $\mathcal{X}(I)$ seja não nula, ficam definidas as seguintes probabilidades:

- $0 \leq f_s \leq 1$ é a probabilidade de um falso *sim*;
- $0 \leq f_n \leq 1$ é a probabilidade de um falso *não*.

Um algoritmo probabilístico \mathcal{A} decide o problema \mathcal{P} em tempo $T_{\mathcal{A}}(n)$ se, para todo I tal que $|I| \leq n$, o tempo do algoritmo para determinar $\mathcal{X}(I)$, com probabilidades de erro f_s e f_n , é $\leq T_{\mathcal{A}}(n)$.

Definição 1.4.3 (algoritmo de decisão tipo Monte Carlo). Um algoritmo probabilístico que decide \mathcal{P} com probabilidade f_n nula e probabilidade $0 < f_s \leq \frac{1}{2}$ em tempo $T_{\mathcal{A}}(n) = O(n^k)$, para algum $k \in \mathbb{Z}_{\geq 0}$, é dito um algoritmo probabilístico de tempo polinomial tipo *Monte Carlo*⁶. Executando \mathcal{A} k vezes, a probabilidade de falso sim é reduzida para f_s^k , ou seja, pelo menos $\frac{1}{2^k}$.

Definição 1.4.4 (problemas de decisão complementares). Seja $\mathcal{P} = (\mathcal{I}, \mathcal{R}, \mathcal{X})$ um problema de decisão. O problema complementar de \mathcal{P} é definido como $\mathcal{P}_c = (\mathcal{I}, \mathcal{R}, \mathcal{X}_c)$ e é tal que, se $\mathcal{X}(I) = R$, então $\mathcal{X}_c(I) = 1 - R$.

Definição 1.4.5 (tempo esperado de execução polinomial). Seja $t_{\mathcal{A}}(I, n)$ a variável aleatória cujo valor é o tempo em que um algoritmo probabilístico \mathcal{A} decide um problema \mathcal{P} cuja instância I tem tamanho $|I| = n$. O tempo esperado de execução, denotado $E[t_{\mathcal{A}}(I, n)]$, é dito polinomial se $E[t_{\mathcal{A}}(n)] = O(n^k)$, para algum $k \in \mathbb{Z}_{\geq 0}$.

Definição 1.4.6 (algoritmo de decisão tipo Las Vegas). Um algoritmo probabilístico que combina decisões de \mathcal{P} com probabilidade f_n nula e probabilidade $0 < f_s \leq \frac{1}{2}$ e de \mathcal{P}_c com probabilidade f_n^c nula e probabilidade $0 < f_s^c \leq \frac{1}{2}$ fixa, em tempo esperado de execução polinomial, é dito um algoritmo probabilístico de tempo polinomial com probabilidade de erro nula, ou algoritmo de tempo polinomial tipo Las Vegas⁷. Executando \mathcal{A} k vezes, a probabilidade de que a resposta seja correta é de pelo menos $1 - \frac{1}{2^k}$.

Definição 1.4.7 (algoritmo de decisão com probabilidade de erro limitada). Um algoritmo probabilístico que decide \mathcal{P} com probabilidade de falso não $f_n < \frac{1}{3}$ e probabilidade de falso sim $f_s < \frac{1}{3}$ em tempo $T_{\mathcal{A}}(n) = O(n^k)$, $k \in \mathbb{Z}_{\geq 0}$ é dito um algoritmo probabilístico de tempo polinomial com probabilidade de erro limitada⁸. \mathcal{A} pode ser executado várias vezes, sendo considerada a resposta majoritária.

1.5 Máquinas de Turing

Definição 1.5.1 (Máquina de Turing Básica). Uma máquina de Turing básica, determinística, BTM, sobre um alfabeto A é uma sêxtupla $\text{BTM} = (Q, A, \delta, q_0, q_a, q_r)$, onde Q é o conjunto finito de estados de controle interno, $q_0, q_a, q_r \in Q$ são os estados inicial, de aceitação e de rejeição, respectivamente e δ é uma função de transição $\delta : Q \times A \rightarrow Q \times A \times \{-1, 0, 1\}$.

⁶c.f. classe RP - randomized polynomial time

⁷c.f. classe ZPP - zero error probability in polynomial time

⁸c.f. classe BPP - bounded error probability in polynomial time

Definição 1.5.2 (Máquina de Turing Probabilística). Uma máquina de Turing probabilística, PTM, sobre um alfabeto A é uma sêxtupla $\text{PTM} = (Q, A, \delta, q_0, q_a, q_r)$, onde Q é o conjunto finito de estados de controle interno, $q_0, q_a, q_r \in Q$ são os estados inicial, de aceitação e de rejeição, respectivamente e δ é uma distribuição das probabilidades de transição $\delta : Q \times A \times Q \times A \times \{-1, 0, 1\} \rightarrow [0, 1]$.

Definição 1.5.3 (Máquina de Turing Quântica). Uma máquina de Turing quântica, QTM, sobre um alfabeto A é uma sêxtupla $\text{QTM} = (Q, A, \delta, q_0, q_a, q_r)$, onde Q é o conjunto finito de estados de controle interno, $q_0, q_a, q_r \in Q$ são os estados inicial, de aceitação e de rejeição, respectivamente e δ é uma função de amplitudes de transição $\delta : Q \times A \times Q \times A \times \{-1, 0, 1\} \rightarrow \mathbb{C}$.

Capítulo 2

Fatoração de Inteiros

The primes are the material out of which all numbers are build up by multiplication.

G. H. Hardy.

2.1 Fatoração de Inteiros em Primos

Nesta seção será apresentado o problema de fatoração de inteiros em primos e um método de solução. A base teórica desse método e o significado preciso da notação utilizada serão tratados nas seções seguintes.

Considere a seguinte definição formal para o problema de fatoração de inteiros:

Problema FATORAÇÃO-EM-PRIMOS-COMPLETA(m): Dado um inteiro m , encontrar todos os números primos que o dividem, bem como suas respectivas multiplicidades¹.

O teorema 2.1.1 garante que FATORAÇÃO-EM-PRIMOS-COMPLETA(m) tem solução e que essa solução é única.

Teorema 2.1.1 (Teorema Fundamental da Aritmética). *Se m é um inteiro diferente de 0, 1 e -1 , então existem primos positivos $p_1 < p_2 < \dots < p_k$ e inteiros positivos e_1, e_2, \dots, e_k , tais que $m = \alpha p_1^{e_1} \dots p_k^{e_k}$, onde $\alpha = \pm 1$. Além disso, essa decomposição é única².*

¹Atualmente um dos métodos de fatoração mais eficientes no modelo tradicional de computação é o Number Field Sieve (NFS), capaz de obter a fatoração em primos de um inteiro m em $\exp(c(\lg m)^{\frac{1}{3}}(\lg \lg m)^{\frac{2}{3}})$ operações sobre bits [16].

²A demonstração deste, bem como de vários teoremas relevantes na Teoria dos Números, pode ser encontrada em [12].

2.1.1 Identificação de Fator Primo

Uma maneira eficiente de encontrar a decomposição de um inteiro em fatores primos é aplicar recursivamente um algoritmo capaz de identificar fatores não-triviais, primos ou compostos, em tempo polinomial. Para os modelos tradicionais de computação, tal algoritmo não foi encontrado. Para o modelo quântico, o algoritmo de Shor resolve esse problema em tempo polinomial.

A descrição formal do problema específico ao qual se aplica o algoritmo de Shor é a seguinte:

Problema FATOR-PRIMO(n): dado um inteiro n ímpar positivo com pelo menos dois fatores primos distintos, $n = p_1^{e_1} \dots p_k^{e_k}$, $k \geq 2$, encontre um fator primo de n .

As restrições sobre n em FATOR-PRIMO(n) estão associadas a um pré-processamento e decorrem dos seguintes fatos: podemos assumir que n é ímpar, pois as potências de 2 podem ser identificadas e colocadas na decomposição em tempo polinomial; o mesmo ocorre com o sinal, caso n seja negativo; também a restrição de que n tenha dois fatores primos distintos decorre da existência de algoritmos polinomiais capazes de identificar números que são potências de um número primo. Portanto, o problema FATOR-PRIMO(n), tal como definido, corresponde à parte mais difícil da identificação de um fator primo.

O seguinte algoritmo devolve um divisor não-trivial (fator primo, ou composto) de n , ou para e imprime "falha":

Algoritmo 2.1.1 (Algoritmo para Extração de Fatores).

Extrai-Fator(n)	veja as observações:
1 escolha um número $a > 1$ em \mathbb{Z}_n ;	1
2 encontre $d = \text{mdc}(a, n)$;	2
3 se $d > 1$, então devolva d e pare.	3 e 4
4 senão, calcule $r = \text{ord}_n(a)$;	5, 6 e 7
5 se r for ímpar então imprima "falha" e pare.	8
6 senão, calcule $d = \text{mdc}(a^{\frac{r}{2}} - 1, n)$;	9, 10 e 11
7 se $d > 1$, então devolva d e pare.	12
8 senão, calcule $d = \text{mdc}(a^{\frac{r}{2}} + 1, n)$;	13
9 se $d = n$, então imprima "falha" e pare.	14 e 15
10 senão, devolva d e pare.	14

Observações³:

1. A escolha de $a > 1$ deve ser feita com probabilidade uniforme sobre $\mathbb{Z}_n = \{0, 1, 2, \dots, n-1\}$.
2. d é o máximo divisor comum entre a e n .
3. Se $d > 1$, então d é um divisor não-trivial (primo ou composto) de n ; se d for primo, o problema FATOR-PRIMO(n) está resolvido; se d for composto, então os fatores de d devem ser extraídos até que um fator primo seja determinado.
4. Se $d = 1$, a e n não têm divisores comuns (são primos entre si).
5. Nesse caso, calcule $r = \text{ord}_n(a)$, isto é calcule r , a ordem do elemento a em \mathbb{Z}_n , que é igual ao período da função $f : \mathbb{Z} \rightarrow \mathbb{Z}_n$ definida como $f(k) = a^k \bmod n$.
6. Como $a^r \equiv 1 \bmod n$, temos que $a^r - 1 \equiv 0 \bmod n$, portanto n divide $a^r - 1$.
7. Em um modelo tradicional de computação, todas as linhas de **Extrai-Fator**(n) podem ser executadas em tempo polinomial, exceto a linha 4. Veremos ao longo do texto que o cálculo eficiente de r é o elemento crucial do algoritmo de Shor.
8. Se r for ímpar, o conhecimento de n e r não é suficiente para permitir a obtenção de um divisor não-trivial.
9. Se r for par, o inteiro $a^r - 1$ é igual ao produto dos inteiros $a^{\frac{r}{2}} - 1$ e $a^{\frac{r}{2}} + 1$.
10. Como n divide $a^r - 1$ e $a^r - 1 = (a^{\frac{r}{2}} - 1)(a^{\frac{r}{2}} + 1)$, os divisores não-triviais de n dividem ou $a^{\frac{r}{2}} - 1$ ou $a^{\frac{r}{2}} + 1$ ou ambos.
11. Certamente n não divide $a^{\frac{r}{2}} - 1$, pois isso implicaria que $a^{\frac{r}{2}} \equiv 1 \bmod n$, o que é absurdo, dada a definição de ordem. Logo, $\text{mdc}(a^{\frac{r}{2}} - 1, n) \neq n$.
12. Então, $d = \text{mdc}(a^{\frac{r}{2}} - 1, n)$ ou é um fator não-trivial de n , ou é 1, caso n não compartilhe divisores não-triviais com $a^{\frac{r}{2}} - 1$.
13. Caso n não compartilhe divisores não-triviais com $a^{\frac{r}{2}} - 1$, ele certamente será divisor, ou irá compartilhar divisores não-triviais com $a^{\frac{r}{2}} + 1$.

³As afirmações contidas nesta série de observações serão demonstradas ao longo do capítulo.

14. Calculando $d = \text{mdc}(a^{\frac{r}{2}} + 1, n)$, será obtido um fator não-trivial de n , ou então n divide $a^{\frac{r}{2}} + 1$ e nesse caso $d = n$.
15. Se n divide $a^{\frac{r}{2}} + 1$, então $a^{\frac{r}{2}} \equiv -1 \pmod{n}$, ou seja, $a^{\frac{r}{2}}$ é uma raiz quadrada não-trivial de 1 módulo n .

Pelo método descrito, fica claro que um fator não-trivial de n pode ser obtido desde que, escolhido um $a \in \mathbb{Z}_n$, $a > 1$, sua ordem $r = \text{ord}_n(a)$ seja par e além disso, $a^{\frac{r}{2}} \not\equiv -1 \pmod{n}$.

2.1.2 Exemplos de Fatoração

O algoritmo Extraí-Fator(n) será aplicado recursivamente ao número $n = 1514581$ para obter sua fatoração em primos completa.

1. $n_1 = 1514581$. $\mathbb{Z}_{n_1}^* = \{1, 2, \dots, 1514580\}$;
escolhendo $a = 11660$, temos $d = \text{mdc}(a, n_1) = 53$,
um fator não-trivial (primo) de n_1 .
2. $n_2 = \frac{1514581}{53} = 28577$. $\mathbb{Z}_{n_2}^* = \{1, 2, \dots, 28576\}$;
escolhendo $a = 9895$, temos $d = \text{mdc}(a, n_2) = 1$;
 $r = \text{ord}_{n_2}(a)$ é igual ao período de $f(k) = a^k \pmod{n_2}$, que é 328;
calculando $d = \text{mdc}(9895^{\frac{328}{2}} - 1, 28577)$ obtemos $d = 1$,
calculando $d = \text{mdc}(9895^{\frac{328}{2}} + 1, 28577)$ obtemos $d = 41$,
outro fator não-trivial (primo) de n_1 .
3. $n_3 = \frac{28577}{41} = 697$. $\mathbb{Z}_{n_3}^* = \{1, 2, \dots, 696\}$;
escolhendo $a = 120$, temos $d = \text{mdc}(a, n_3) = 1$;
 $r = \text{ord}_{n_3}(a)$ é igual ao período de $f(k) = a^k \pmod{n_3}$, que é 8;
calculando $d = \text{mdc}(120^{\frac{8}{2}} - 1, 697)$ obtemos $d = 17$,
outro fator não-trivial (primo) de n_1 .
4. $n_4 = \frac{697}{17} = 41$. como 41 é primo, a fatoração em primos completa de $n = 1514581$ é $(17) * (41^2) * (53)$.

2.1.3 Exercícios

1. Sendo $a \bmod b = a - \lfloor \frac{a}{b} \rfloor b$, o resto da divisão de a por b , mostre que o algoritmo de Euclides⁴, cuja complexidade de tempo é polinomial, calcula corretamente o máximo divisor comum entre inteiros positivos a e b .

Algoritmo 2.1.2 (Algoritmo de Euclides).Euclides(a, b)

- 1 se $b = 0$
- 2 então devolva a .
- 3 senão, devolva Euclides($b, a \bmod b$).

2. Mostre que a recorrência

$$f(k) = \begin{cases} a^k \bmod n & k = k_0 \\ af(k-1) \bmod n & k > k_0 \end{cases}$$

calcula $f(k) = a^k \bmod n$, $k \geq k_0$. Determine $f(k) = 7^k \bmod 21$ para $1 \leq k \leq 2$. Qual é o período dessa função? Note que $\text{mdc}(7, 21) \neq 1$.

3. Mostre que o algoritmo abaixo encontra, se houver, um fator não-trivial de um inteiro positivo n . Mostre que o tempo de execução desse algoritmo é proporcional a n , ou seja, é exponencial no número de dígitos de n , que é $O(\log_{10} n)$.

Algoritmo 2.1.3 (Algoritmo Exponencial para Fatoração).Busca-Fator(n)

- 1 para i de 2 até $n-1$
- 2 se $n \equiv 0 \bmod i$
- 3 então devolva i .

4. Mostre que todo inteiro composto tem um fator primo. Sendo essa afirmação correta, mostre que todo inteiro tem uma fatoração em primos completa. O quê o teorema fundamental da aritmética garante acerca dessa fatoração?

⁴Uma análise de complexidade desse algoritmo pode ser encontrada em [4].

5. Seja $\alpha = x^k \bmod n$, onde $x, k, n \in \mathbb{N}$ e $n > 1$. Mostre que o algoritmo $\text{Exp-Modular}(x, k, n)$ encontra α corretamente.

Algoritmo 2.1.4 (Exponenciação Modular Rápida).

$\text{Exp-Modular}(x, k, n)$

- 1 encontre $(k_{\beta-1}, k_{\beta-2}, \dots, k_2, k_1, k_0)$, representação binária de k ;
- 2 $\alpha = 1$;
- 3 para i de $\beta - 1$ até 0
- 4 $\alpha = \alpha^2 \bmod n$;
- 5 se $k_i = 1$
- 6 $\alpha = \alpha x \bmod n$.
- 7 devolva α .

6. Não se conhece algoritmo eficiente para determinação de ordem para os modelos computacionais tradicionais. O algoritmo $\text{Encontra-Ordem}(a, n)$ calcula $r = \text{ord}_n(a)$.

Algoritmo 2.1.5 (Ordem em Tempo Exponencial).

$\text{Encontra-Ordem}(a, n)$

- 1 $r = 1$;
- 2 $p = a$;
- 3 enquanto $p \neq 1 \bmod n$
- 4 $r = r + 1$;
- 5 $p = p * a$;
- 6 devolva r .

Exiba números a e n tais que, para esses valores, $\text{Encontra-Ordem}(a, n)$ consome tempo proporcional a n , ou seja, tais que o tempo de execução seja exponencial no número de dígitos de n .

2.2 O Grupo Aditivo \mathbb{Z}_n

2.2.1 Noções de Teoria dos Grupos

Definição de Grupo e Notação

Sejam

G , um conjunto não-vazio formado por elementos g ;

“ \star ”, uma operação binária em G com propriedades

- de fechamento: $g_1 \star g_2 \in G, \forall g_1 \text{ e } g_2 \text{ em } G$;
- associativa: $(g_1 \star g_2) \star g_3 = g_1 \star (g_2 \star g_3), \forall g_1, g_2 \text{ e } g_3 \text{ em } G$;
- de existência de identidade: existe um único elemento e em G , denominado identidade, tal que $e \star g = g \star e, \forall g \text{ em } G$;

“ \ast ”, uma operação unária em G , denominada inversão, tal que

- $g \star (\ast g) = e, \forall g \text{ em } G$;
- vale a propriedade de fechamento: $(\ast g) \in G, \forall g \text{ em } G$.

Um *grupo* é uma quádrupla $\mathcal{G} = (G, \star, \ast, e)$. Um grupo é dito comutativo, ou abeliano, se $g_1 \star g_2 = g_2 \star g_1, \forall g_1 \text{ e } g_2 \text{ em } G$. Um grupo é dito *aditivo*, ou então *multiplicativo*, de acordo com a natureza de sua operação binária.

Neste texto será adotada a seguinte notação:

grupo aditivo: $\mathcal{G} = (G, +, -g, 0)$;

grupo multiplicativo: $\mathcal{G} = (G, \cdot, g^{-1}, 1)$.

Exemplos:

- $\mathcal{G} = (\mathbb{Z}, +, -z, 0)$ é um grupo aditivo abeliano;
- $\mathcal{G} = (\mathbb{Q}_{>0}, \cdot, q^{-1}, 1)$ é um grupo multiplicativo abeliano;
- $(\mathbb{Z}_{>0}, +)$ não é um grupo aditivo, pois não tem elemento identidade, nem operação de inversão;
- $\mathcal{G} = (\mathbb{C}_{\neq 0}, \cdot, c^{-1}, 1 + 0i)$ é um grupo multiplicativo abeliano.

Subgrupos e Cosets

Definição 2.2.1 (Subgrupo). Se $\mathcal{G} = (G, \star, \ast, e)$, $H \subseteq G$ e $\mathcal{H} = (H, \star, \ast, e)$ é um grupo, então \mathcal{H} é dito um subgrupo de \mathcal{G} . Portanto em um *subgrupo* \mathcal{H} de um grupo \mathcal{G} tem-se $H \subseteq G$; $e \in H$ e as operações binária “ \star ” e unária “ \ast ” têm propriedade de fechamento em \mathcal{H} . A relação subgrupo-grupo é denotada $\mathcal{H} \leq \mathcal{G}$.

Definição 2.2.2 (Coset). Para cada $g \in G$, o *coset* de \mathcal{H} determinado por g é um subconjunto definido como $g \star H = \{g \star h : h \in H\}$. Os *cosets* de um subgrupo $\mathcal{H} \leq \mathcal{G}$ cobrem o grupo \mathcal{G} .

Definição 2.2.3 (Congruência). Se $g_1 \star H = g_2 \star H$ é dito que g_1 é congruente à g_2 módulo H . Isso acontece se e somente se $(\ast g_1) \star g_2 \in H$.

Um subgrupo $\mathcal{H} \leq \mathcal{G}$ é dito *normal* se, para todo $g \in G$ e $h \in H$ tem-se $g \star h \star (\ast g) \in H$. Note que todo subgrupo de um grupo abeliano é normal. Para um *subgrupo normal* $\mathcal{H} \leq \mathcal{G}$ a operação \star entre *cosets* $g_1 \star H$ e $g_2 \star H$ é definida como

$$(g_1 \star H) \star (g_2 \star H) = (g_1 \star g_2) \star H.$$

Definição 2.2.4 (Grupo Fator). Seja $\mathcal{H} \leq \mathcal{G}$ um subgrupo normal de um grupo \mathcal{G} . O grupo fator, também chamado de grupo quociente, \mathcal{G}/\mathcal{H} tem como elementos de grupo os *cosets* de \mathcal{H} , a operação \star entre *cosets* como operação binária de grupo, o elemento identidade é $e \star H = H$ e a operação unária de grupo é $(\ast g) \star H$.

2.2.2 Grupo Aditivo \mathbb{Z}_n

Considere o grupo aditivo abeliano dos números inteiros $\mathcal{G} = (\mathbb{Z}, +, -z, 0)$. Sejam $n \in \mathbb{N}$ e $\mathcal{H} \leq \mathcal{G}$ o grupo aditivo normal formado pelos inteiros divisíveis por n , denotado $\mathcal{H} = (n\mathbb{Z}, +, -nz, 0)$. Os *cosets* de \mathcal{H} são da forma $z + n\mathbb{Z}$, $z \in \mathbb{Z}$. O grupo fator aditivo de \mathcal{G} e \mathcal{H} é denotado $\mathcal{G}/\mathcal{H} = (z + n\mathbb{Z}, +, -z + nz, n\mathbb{Z})$.

Os conjuntos de elementos desses grupos são:

- $G = \mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$;
- $H = n\mathbb{Z} = \{\dots, -3n, -2n, -n, 0, n, 2n, 3n, \dots\}$;
- *coset* $z + n\mathbb{Z} = \{\dots, z - 3n, z - 2n, z - n, z, z + n, z + 2n, z + 3n, \dots\}$;
- $\mathcal{G}/\mathcal{H} = \{\dots, -2 + n\mathbb{Z}, -1 + n\mathbb{Z}, 0 + n\mathbb{Z}, 1 + n\mathbb{Z}, 2 + n\mathbb{Z}, \dots\}$;

Dado um natural n , o grupo fator aditivo \mathcal{G}/\mathcal{H} é denotado \mathbb{Z}_n , ou mesmo $\mathbb{Z}/(n\mathbb{Z})$. A seguir será construído o grupo \mathbb{Z}_3 .

- $G = \mathbb{Z} = \{\dots, -4, -3, -2, -1, 0, 1, 2, 3, 4, \dots\}$;
- $H = 3\mathbb{Z} = \{\dots, -12, -9, -6, -3, 0, 3, 6, 9, 12, \dots\}$;
- $\text{coset } -3 + 3\mathbb{Z} = \{\dots, -15, -12, -9, -6, -3, 0, 3, 6, 9, \dots\}$;
- $\text{coset } -2 + 3\mathbb{Z} = \{\dots, -14, -11, -8, -5, -2, 1, 4, 7, 10, \dots\}$;
- $\text{coset } -1 + 3\mathbb{Z} = \{\dots, -13, -10, -7, -4, -1, 2, 5, 8, 11, \dots\}$;
- $\text{coset } 0 + 3\mathbb{Z} = \{\dots, -12, -9, -6, -3, 0, 3, 6, 9, 12, \dots\}$;
- $\text{coset } 1 + 3\mathbb{Z} = \{\dots, -11, -8, -5, -2, 1, 4, 7, 10, 13, \dots\}$;
- $\text{coset } 2 + 3\mathbb{Z} = \{\dots, -10, -7, -4, -1, 2, 5, 8, 11, 14, \dots\}$;
- $\text{coset } 3 + 3\mathbb{Z} = \{\dots, -9, -6, -3, 0, 3, 6, 9, 12, 15, \dots\}$;

É fácil notar que $-3 + 3\mathbb{Z} = 0 + 3\mathbb{Z} = 3 + 3\mathbb{Z}$, $-2 + 3\mathbb{Z} = 1 + 3\mathbb{Z}$ e $-1 + 3\mathbb{Z} = 2 + 3\mathbb{Z}$. De fato $z_1 + 3\mathbb{Z} = z_2 + 3\mathbb{Z}$ sempre que $z_2 - z_1$ for divisível por 3. Portanto os elementos de grupo de \mathbb{Z}_3 são apenas 3 *cosets*. Tomando como representantes os *cosets* $0 + 3\mathbb{Z}$, $1 + 3\mathbb{Z}$ e $2 + 3\mathbb{Z}$ temos

$$\mathbb{Z}_3 = \{ 0 + 3\mathbb{Z}, 1 + 3\mathbb{Z}, 2 + 3\mathbb{Z} \}.$$

Definição 2.2.5 (Congruência em \mathbb{Z}_n). Um *coset* $z_1 + n\mathbb{Z}$ é igual a outro *coset* $z_2 + n\mathbb{Z}$ quando $z_2 - z_1$ é divisível por n . Nesse caso é dito que z_1 e z_2 são congruentes módulo $n\mathbb{Z}$, ou simplesmente é dito que z_1 e z_2 são congruentes módulo n , relação denotada por $z_1 \equiv z_2 \pmod{n}$.

Definição 2.2.6 (Representação de \mathbb{Z}_n). A representação do grupo fator aditivo $\mathbb{Z}_n = (z + n\mathbb{Z}, +, -z + nz, n\mathbb{Z})$ por seus elementos de grupo é

$$\mathbb{Z}_n = \{ 0 + n\mathbb{Z}, 1 + n\mathbb{Z}, 2 + n\mathbb{Z}, \dots, (n-1) + n\mathbb{Z} \},$$

ou então, quando subentendida a relação de congruência, \mathbb{Z}_n pode ser representado como

$$\mathbb{Z}_n = \{ 0, 1, 2, \dots, (n-1) \}.$$

2.2.3 Grupo Multiplicativo \mathbb{Z}_n^*

O grupo $\mathbb{Z}_n = (z + n\mathbb{Z}, +, -z + nz, n\mathbb{Z})$ tem como elementos de grupo os *cosets* $z + n\mathbb{Z}$. Alguns desses *cosets* serão escolhidos para formar um novo grupo, onde a operação binária será a multiplicação.

Definição 2.2.7 (produto de cosets). Considere a seguinte definição para o produto de *cosets*: $(z_1 + n\mathbb{Z})(z_2 + n\mathbb{Z}) = (z_1 z_2 + n\mathbb{Z})$. Será mostrado que essa definição é consistente, que atende à propriedade de fechamento em \mathbb{Z}_n e que o elemento identidade à ela associado está em \mathbb{Z}_n .

- sejam $z_1 + n\mathbb{Z} = z'_1 + n\mathbb{Z}$ e $z_2 + n\mathbb{Z} = z'_2 + n\mathbb{Z}$; como $z'_1 - z_1$ e $z'_2 - z_2$ são divisíveis por n , $(z'_1 - z_1)z'_2 + (z'_2 - z_2)z_1$ é também divisível por n , logo $z'_1 z'_2 - z_1 z'_2 + z'_2 z_1 - z_2 z_1 = z'_1 z'_2 - z_2 z_1$ é divisível por n e portanto $z_1 z_2 + n\mathbb{Z} = z'_1 z'_2 + n\mathbb{Z}$, o que mostra a consistência da definição de produto de *cosets*.
- sejam $z_1 + n\mathbb{Z}$ e $z_2 + n\mathbb{Z} \in \mathbb{Z}_n$; se $z_1 z_2 < n - 1$, então $z_1 z_2 + n\mathbb{Z}$ é um dos *cosets* que representam \mathbb{Z}_n ; se $z_1 z_2 = \alpha n + \beta$, $\alpha, \beta \in \mathbb{N}$, então $z_1 z_2 \equiv \beta \pmod{n}$, portanto $(z_1 + n\mathbb{Z})(z_2 + n\mathbb{Z}) \in \mathbb{Z}_n$.
- $1 + n\mathbb{Z} \in \mathbb{Z}_n$ é o elemento identidade.

Definição 2.2.8 (inversão de cosets). A operação unária de inversão é assim definida: $z^{-1} + n\mathbb{Z} = (z + n\mathbb{Z})^{-1} \Leftrightarrow (z + n\mathbb{Z})(z^{-1} + n\mathbb{Z}) = zz^{-1} + n\mathbb{Z} = 1 + n\mathbb{Z}$.

A condição $zz^{-1} + n\mathbb{Z} = 1 + n\mathbb{Z}$ é atendida quando $zz^{-1} - 1$ é divisível por n . Suponha que $\text{mdc}(z, n) = \alpha > 1$. Nesse caso, se $zz^{-1} - 1$ for divisível por n , também será divisível por α , o que é absurdo, pois nesse caso $\frac{1}{\alpha}$, $\alpha > 1$, teria que ser inteiro. Disso se conclui que a operação de inversão só está adequadamente definida se $\text{mdc}(z, n) = 1$, ou seja z e n devem ser primos entre si.

Sendo $\text{mdc}(z, n) = 1$, será demonstrado que $z^{-1} + n\mathbb{Z}$ é único e está em \mathbb{Z}_n .

Lema 2.2.1 (Identidade de Bezout). Para todo par de naturais z, n , existe um par de inteiros α, β , tais que $\alpha z + \beta n = \text{mdc}(z, n)$

Sejam a e b , tais que $az + bn = \text{mdc}(z, n) = 1$. Nesse caso, $az - 1$ é divisível por n , logo $az + n\mathbb{Z} = 1 + n\mathbb{Z}$. Como $(az + n\mathbb{Z}) = (a + n\mathbb{Z})(z + n\mathbb{Z})$, temos $(z + n\mathbb{Z})^{-1} = (az + n\mathbb{Z})$.

Considere $M_n = \{1, z_1, z_2, \dots, z_m\}$ o conjunto de todos os inteiros $1 \leq z \leq n - 1$ tais que $\text{mdc}(z, n) = 1$. Os exemplos a seguir mostram conjuntos M_n e os inversos, para alguns valores de n .

Exemplos:

- $M_8 = \{1, 3, 5, 7\}$ e $1 + 8\mathbb{Z} = \{\dots, -15, -7, 1, 9, 17, 25, 33, 41, 49, \dots\}$, logo $1 \cdot 1 \equiv 1 \pmod{8}$, $3 \cdot 3 \equiv 1 \pmod{8}$, $5 \cdot 5 \equiv 1 \pmod{8}$ e $7 \cdot 7 \equiv 1 \pmod{8}$.

· $M_5 = \{1, 2, 3, 4\}$ e $1 + 5\mathbb{Z} = \{\dots, -9, -4, 1, 6, 11, 16, 21, 26, \dots\}$, logo
 $1 \cdot 1 \equiv 1 \pmod{5}$, $2 \cdot 3 \equiv 1 \pmod{5}$, $3 \cdot 2 \equiv 1 \pmod{5}$ e $4 \cdot 4 \equiv 1 \pmod{5}$.

Definição 2.2.9 (Grupo Multiplicativo \mathbb{Z}_n^*). Seja $M = \{1, z_1, z_2, \dots, z_m\}$ o conjunto de todos os inteiros $1 \leq z \leq n-1$, tais que $\text{mdc}(z, n) = 1$. O grupo multiplicativo \mathbb{Z}_n^* é assim definido:

$$\mathbb{Z}_n^* = (\{z + n\mathbb{Z} : z \in M\}, \cdot, z^{-1} + n\mathbb{Z}, 1 + n\mathbb{Z}), \text{ onde } zz^{-1} = 1 \pmod{n}.$$

Exemplos:

$$\begin{aligned} \cdot \mathbb{Z}_6^* &= (\{1 + 6\mathbb{Z}, 5 + 6\mathbb{Z}\}, \cdot, z^{-1} + 6\mathbb{Z}, 1 + 6\mathbb{Z}), \\ \cdot \mathbb{Z}_{12}^* &= (\{1 + 12\mathbb{Z}, 5 + 12\mathbb{Z}, 7 + 12\mathbb{Z}, 11 + 12\mathbb{Z}\}, \cdot, z^{-1} + 12\mathbb{Z}, 1 + 12\mathbb{Z}). \end{aligned}$$

2.2.4 Grupos Cíclicos

Definição 2.2.10 (Grupo Aditivo Cíclico de Tamanho Finito). Um grupo aditivo \mathcal{G} de tamanho m é dito cíclico se existe pelo menos um elemento de grupo $g \in G$, tal que o conjunto G pode ser escrito como $G = \{0g, g, 2g, \dots, (m-1)g\}$, onde $0g = e$, por definição.

Exemplo: $\mathbb{Z}_n = \{0 + n\mathbb{Z}, 1 + n\mathbb{Z}, \dots, n-1 + n\mathbb{Z}\}$ é cíclico, e pode ser gerado por todo $z \in \mathbb{Z}_n, z \neq 0$, tal que $\text{mdc}(z, n) = 1$. Para $n = 5$ temos:

$$\begin{array}{llll} 0(1+5\mathbb{Z}) = 0+5\mathbb{Z} & 0(2+5\mathbb{Z}) = 0+5\mathbb{Z} & 0(3+5\mathbb{Z}) = 0+5\mathbb{Z} & 0(4+5\mathbb{Z}) = 0+5\mathbb{Z} \\ 1(1+5\mathbb{Z}) = 1+5\mathbb{Z} & 1(2+5\mathbb{Z}) = 2+5\mathbb{Z} & 1(3+5\mathbb{Z}) = 3+5\mathbb{Z} & 1(4+5\mathbb{Z}) = 4+5\mathbb{Z} \\ 2(1+5\mathbb{Z}) = 2+5\mathbb{Z} & 2(2+5\mathbb{Z}) = 4+5\mathbb{Z} & 2(3+5\mathbb{Z}) = 1+5\mathbb{Z} & 2(4+5\mathbb{Z}) = 3+5\mathbb{Z} \\ 3(1+5\mathbb{Z}) = 3+5\mathbb{Z} & 3(2+5\mathbb{Z}) = 1+5\mathbb{Z} & 3(3+5\mathbb{Z}) = 4+5\mathbb{Z} & 3(4+5\mathbb{Z}) = 2+5\mathbb{Z} \\ 4(1+5\mathbb{Z}) = 4+5\mathbb{Z} & 4(2+5\mathbb{Z}) = 3+5\mathbb{Z} & 4(3+5\mathbb{Z}) = 2+5\mathbb{Z} & 4(4+5\mathbb{Z}) = 1+5\mathbb{Z} \end{array}$$

note que

$$5(1+5\mathbb{Z}) = 0+5\mathbb{Z} \quad 5(2+5\mathbb{Z}) = 0+5\mathbb{Z} \quad 5(3+5\mathbb{Z}) = 0+5\mathbb{Z} \quad 5(4+5\mathbb{Z}) = 0+5\mathbb{Z}$$

Definição 2.2.11 (Grupo Multiplicativo Cíclico de Tamanho Finito). Um grupo multiplicativo \mathcal{G} de tamanho m é dito cíclico se existe pelo menos um elemento de grupo $g \in G$, tal que o conjunto G pode ser escrito como $G = \{g^0, g^1, g^2, \dots, g^{m-1}\}$, onde $g^0 = e$, por definição.

Exemplos de grupos multiplicativos cíclicos:

$$\begin{aligned} \cdot \mathbb{Z}_6^* &= \{1 + 6\mathbb{Z}, 5 + 6\mathbb{Z}\} \text{ é cíclico, pois:} \\ &\quad (5 + 6\mathbb{Z})^0 = 1 + 6\mathbb{Z} \text{ (por definição)} \\ &\quad (5 + 6\mathbb{Z})^1 = 5 + 6\mathbb{Z} \end{aligned}$$

- $\mathbb{Z}_7^* = \{1 + 7\mathbb{Z}, 2 + 7\mathbb{Z}, 3 + 7\mathbb{Z}, 4 + 7\mathbb{Z}, 5 + 7\mathbb{Z}, 6 + 7\mathbb{Z}\}$ é cíclico, pois:

$$\begin{aligned} (3 + 7\mathbb{Z})^0 &= 1 + 7\mathbb{Z} \text{ (por definição)} \\ (3 + 7\mathbb{Z})^1 &= 3 + 7\mathbb{Z} \\ (3 + 7\mathbb{Z})^2 &= 9 + 7\mathbb{Z} = 2 + 7\mathbb{Z} \\ (3 + 7\mathbb{Z})^3 &= 27 + 7\mathbb{Z} = 6 + 7\mathbb{Z} \\ (3 + 7\mathbb{Z})^4 &= 81 + 7\mathbb{Z} = 4 + 7\mathbb{Z} \\ (3 + 7\mathbb{Z})^5 &= 243 + 7\mathbb{Z} = 5 + 7\mathbb{Z} \end{aligned}$$

Usando a partir de agora a noção de congruência implícita em \mathbb{Z}_n^* .

- $\mathbb{Z}_8^* = \{1, 3, 5, 7\}$ não é cíclico.
- $\mathbb{Z}_9^* = \{1, 2, 4, 5, 7, 8\}$ é cíclico, pois:

$$\begin{aligned} 2^0 &\equiv 1 \pmod{9} \\ 2^1 &\equiv 2 \pmod{9} \\ 2^2 &\equiv 4 \pmod{9} \\ 2^3 &\equiv 8 \pmod{9} \\ 2^4 &\equiv 7 \pmod{9} \\ 2^5 &\equiv 5 \pmod{9} \end{aligned}$$
- $\mathbb{Z}_{10}^* = \{1, 3, 7, 9\}$ é cíclico, pois:

$$\begin{aligned} 7^0 &\equiv 1 \pmod{10} \\ 7^1 &\equiv 7 \pmod{10} \\ 7^2 &\equiv 9 \pmod{10} \\ 7^3 &\equiv 3 \pmod{10} \end{aligned}$$
- $\mathbb{Z}_{15}^* = \{1, 2, 4, 7, 8, 11, 13, 14\}$ não é cíclico.
- $\mathbb{Z}_{16}^* = \{1, 3, 5, 7, 9, 11, 13, 15\}$ não é cíclico.
- $\mathbb{Z}_{17}^* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$ é cíclico.
- $\mathbb{Z}_{21}^* = \{1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20\}$ não é cíclico.

São cíclicos os grupos \mathbb{Z}_n^* onde n é primo, ou potência de número primo, caso de \mathbb{Z}_7^* , \mathbb{Z}_9^* , \mathbb{Z}_{17}^* , ou quando n é par, mas não uma potência de 2, caso de \mathbb{Z}_6^* e \mathbb{Z}_{10}^* .

Não são cíclicos os grupos \mathbb{Z}_n^* onde n é impar, mas não potência de número primo, caso de \mathbb{Z}_{15}^* e \mathbb{Z}_{21}^* , ou quando n é uma potência de 2, caso de \mathbb{Z}_8^* e \mathbb{Z}_{16}^* .

2.2.5 Conceito de Ordem em \mathbb{Z}_n^*

Definição 2.2.12 (Ordem em um Grupo Multiplicativo). Em um grupo multiplicativo \mathcal{G} , a *ordem* de um elemento g é o menor inteiro não negativo z tal que $g^z = e$.

Definição 2.2.13 (Ordem no Grupo Multiplicativo \mathbb{Z}_n^*). Em \mathbb{Z}_n^* a *ordem* de um elemento $z + n\mathbb{Z}$ é o menor natural r tal que $(z + n\mathbb{Z})^r = 1 + n\mathbb{Z}$, ou, considerando a noção de congruência, $z^r \equiv 1 \pmod{n}$. A *ordem* é denotada $\text{ord}_n(a)$.

Exemplos:

- em \mathbb{Z}_6^* , a ordem de $5 + 6\mathbb{Z}$ é 2, pois $(5 + 6\mathbb{Z})^2 = 25 + 6\mathbb{Z} = 1 + 6\mathbb{Z}$ e 2 é o menor natural para o qual isso ocorre.
- em \mathbb{Z}_7^* , a ordem de $3 + 7\mathbb{Z}$ é 6, pois $(3 + 7\mathbb{Z})^6 = 729 + 7\mathbb{Z} = 1 + 7\mathbb{Z}$ e 6 é o menor natural para o qual isso ocorre.

Usando a noção de congruência e a notação de ordem...

- $6 = \text{ord}_9(2)$, pois $2^6 = 64$ e $64 \equiv 1 \pmod{9}$.
- $4 = \text{ord}_{10}(7)$, pois $7^4 = 2401$ e $2401 \equiv 1 \pmod{10}$.
- no grupo \mathbb{Z}_{15}^* temos:
 - $1 = \text{ord}_{15}(1)$
 - $4 = \text{ord}_{15}(2)$, pois $2^4 = 16 \equiv 1 \pmod{15}$
 - $2 = \text{ord}_{15}(4)$, pois $4^2 = 16 \equiv 1 \pmod{15}$
 - $4 = \text{ord}_{15}(7)$, pois $7^4 = 2401 = 160 \cdot 15 + 1 \equiv 1 \pmod{15}$
 - $4 = \text{ord}_{15}(8)$, pois $8^4 = 4096 = 273 \cdot 15 + 1 \equiv 1 \pmod{15}$
 - $2 = \text{ord}_{15}(11)$, pois $11^2 = 121 \equiv 1 \pmod{15}$
 - $4 = \text{ord}_{15}(13)$, pois $13^4 = 28561 = 1904 \cdot 15 + 1 \equiv 1 \pmod{15}$
 - $2 = \text{ord}_{15}(14)$, pois $14^2 = 196 \equiv 1 \pmod{15}$
- no grupo \mathbb{Z}_{21}^* temos:
 - $1 = \text{ord}_{21}(1)$
 - $6 = \text{ord}_{21}(2)$, pois $2^6 = 64 = 3 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $3 = \text{ord}_{21}(4)$, pois $4^3 = 64 = 3 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $6 = \text{ord}_{21}(5)$, pois $5^6 = 15625 = 744 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $2 = \text{ord}_{21}(8)$, pois $8^2 = 64 = 3 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $6 = \text{ord}_{21}(10)$, pois $10^6 = 1000000 = 47619 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $6 = \text{ord}_{21}(11)$, pois $11^6 = 1771561 = 84360 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $2 = \text{ord}_{21}(13)$, pois $13^2 = 169 = 8 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $3 = \text{ord}_{21}(16)$, pois $16^3 = 4096 = 195 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $6 = \text{ord}_{21}(17)$, pois $17^6 = 24137569 = 1149408 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $6 = \text{ord}_{21}(19)$, pois $19^6 = 47045881 = 2240280 \cdot 21 + 1 \equiv 1 \pmod{21}$
 - $2 = \text{ord}_{21}(20)$, pois $20^2 = 400 = 19 \cdot 21 + 1 \equiv 1 \pmod{21}$

2.2.6 Decomposição de \mathbb{Z}_n^* em Subgrupos Cíclicos

Considere a fatoração em primos completa $n = p_1^{e_1} \dots p_r^{e_r}$ para n ímpar ($p_i \neq 2, i = 1, \dots, r$). Colocando a fatoração completa na forma $n = n_1 \dots n_r$, onde $n_i = p_i^{e_i}, i = 1, \dots, r$, se obtém fatores que são primos ou potência de primos. Será mostrado que o grupo \mathbb{Z}_n^* , n ímpar, pode ser expresso como produto cartesiano de r subgrupos cíclicos $\mathbb{Z}_{n_i}^* \leq \mathbb{Z}_n^*$.

Para que o produto cartesiano $\mathcal{G}' \times \mathcal{G}''$ de grupos multiplicativos tenha estrutura de grupo basta que se defina $(g'_1, g''_1)(g'_2, g''_2) = (g'_1 g'_2, g''_1 g''_2)$. Nesse caso (e', e'') é a identidade e $(g', g'')^{-1} = (g'^{-1}, g''^{-1})$.

Exemplo:

$$\mathbb{Z}_3^* = \{1, 2\} \text{ e } \mathbb{Z}_5^* = \{1, 2, 3, 4\}.$$

$$\begin{aligned} \mathbb{Z}_3^* \times \mathbb{Z}_5^* = & \{ (1, 1), (1, 2), (1, 3), (1, 4), (2, 1), (2, 2), (2, 3), (2, 4) \}, \\ & (a, b) \cdot (c, d) = (ac, bd), \\ & (a, b)^{-1} = (c, d) : ac \equiv 1 \pmod{3}, bd \equiv 1 \pmod{5}, \\ & (1, 1). \end{aligned}$$

Note que \mathbb{Z}_3^* e \mathbb{Z}_5^* são cíclicos (3 e 5 são primos) e podem ser escritos, por exemplo, como $\mathbb{Z}_3^* = \{2^0, 2^1\}$ e $\mathbb{Z}_5^* = \{3^0, 3^1, 3^2, 3^3\}$.

Para mostrar que $\mathbb{Z}_n^* = \mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_2}^* \times \dots \times \mathbb{Z}_{n_r}^*$ é necessário mostrar que cada r -upla $(g', g'', g''', \dots, g^{r'})$ do produto $\mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_r}^*$ está associada a um único elemento de grupo $z + n\mathbb{Z}$ de \mathbb{Z}_n^* . Para tanto será utilizado o teorema chinês do resto.

Teorema Chinês do Resto

Teorema 2.2.2 (Teorema Chinês do Resto). *Seja $n = n_1 n_2 \dots n_r$, onde $\text{mdc}(n_i, n_j) = 1$ quando $i \neq j$. Então, para um dado grupo de r cosets $z_i + n_i \mathbb{Z}, i \in \{1, 2, \dots, r\}$, existe um único coset $k + n\mathbb{Z}$ tal que $k + n_i \mathbb{Z} = z_i + n_i \mathbb{Z}$.*

Demonstração: seja $m_i = \frac{n}{n_i}$, então $\text{mdc}(n_i, m_i) = 1$ e pela identidade de Bezout $a_i m_i + b_i n_i = 1$ para inteiros a_i e b_i . Seja $k = a_1 m_1 z_1 + \dots + a_r m_r z_r$, então $k - z_i = a_1 m_1 z_1 + \dots + (a_i m_i - 1) z_i + \dots + a_r m_r z_r$. Como $a_i m_i + b_i n_i = 1$, $a_i m_i - 1 = -b_i n_i$, portanto $a_i m_i - 1$ é divisível por n_i , assim como cada m_j onde $j \neq i$. Logo, $k - z_i$ é divisível por n_i . Segue que $k + n_i \mathbb{Z} = z_i + n_i \mathbb{Z}$. Suponha que exista k' tal que $k' + n_i \mathbb{Z} = z_i + n_i \mathbb{Z}$ para todo i . Nesse caso $k' - k$ seria divisível por cada um dos n_i e, como os n_i são primos entre si, $k' - k$ seria divisível por $n = n_1 n_2 \dots n_r$. Portanto $k' + n\mathbb{Z} = k + n\mathbb{Z}$.

□

Suponha que cada grupo de r *cosets* a que se refere o teorema chinês do resto seja tomado de $\mathbb{Z}_{n_i}^*$, $i = 1, \dots, r$ (isso equivale a escolher uma fatoração de n em primos e potências de primos). Nesse caso, cada r -upla $(z_1 + n_1\mathbb{Z}, z_2 + n_2\mathbb{Z}, \dots, z_r + n_r\mathbb{Z})$ em $\mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_2}^* \times \dots \times \mathbb{Z}_{n_r}^*$ está associada a um único *coset* $k + n\mathbb{Z}$. Resta mostrar que tal $k + n\mathbb{Z}$ está em \mathbb{Z}_n^* e que o número de elementos de \mathbb{Z}_n^* é igual ao número de r -uplas definidas em $\mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_2}^* \times \dots \times \mathbb{Z}_{n_r}^*$.

Para mostrar que $k + n\mathbb{Z}$ está em \mathbb{Z}_n^* basta mostrar que $\text{mdc}(k, n) = 1$. Suponha, por absurdo, que $\text{mdc}(k, n) > 1$, então para algum $i \in \{1, 2, \dots, r\}$ teríamos $\text{mdc}(k, n_i) > 1$, mas $k + n_i\mathbb{Z} = z_i + n_i\mathbb{Z}$ e, por escolha, $z_i + n_i\mathbb{Z} \in \mathbb{Z}_{n_i}^*$ então, necessariamente $\text{mdc}(z_i, n_i) = \text{mdc}(k, n_i) = 1$.

Resta mostrar que o número de elementos de \mathbb{Z}_n^* é igual ao número de r -uplas definidas em $\mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_2}^* \times \dots \times \mathbb{Z}_{n_r}^*$. Certamente o número de elementos de \mathbb{Z}_n^* não é menor que o número de r -uplas, já que cada r -upla está associada a um *coset* distinto $k + n\mathbb{Z}$ de \mathbb{Z}_n^* . Suponha que o número de elementos de \mathbb{Z}_n^* seja maior que o número de r -uplas. Nesse caso existiria $k + n\mathbb{Z}$ em \mathbb{Z}_n^* tal que $\text{mdc}(k, n) = 1$ e $\text{mdc}(k, n_i) > 1$ para algum $i \in \{1, 2, \dots, r\}$. Mas $n = n_1 n_2 \dots n_r$ onde cada n_i é primo, ou potência de primo, então k seria necessariamente um divisor de n e portanto $\text{mdc}(k, n) > 1$.

A conclusão é que, se $n = n_1 n_2 \dots n_r$ é uma fatoração de n em primos ou potências de primos, então $\mathbb{Z}_n^* = \mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_2}^* \times \dots \times \mathbb{Z}_{n_r}^*$. Mais precisamente $\mathbb{Z}_n^* \cong \mathbb{Z}_{n_1}^* \times \mathbb{Z}_{n_2}^* \times \dots \times \mathbb{Z}_{n_r}^*$, onde \cong denota um isomorfismo de grupo⁵.

Exemplo:

$$\mathbb{Z}_3^* = \{1, 2\}, \quad \mathbb{Z}_5^* = \{1, 2, 3, 4\} \text{ e } \mathbb{Z}_{15}^* = \{1, 2, 4, 7, 8, 11, 13, 14\}.$$

$$\mathbb{Z}_3^* \times \mathbb{Z}_5^* = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 1), (2, 2), (2, 3), (2, 4)\},$$

$$\begin{aligned} (1, 1) &= (1 \bmod 3, 1 \bmod 5); \\ (1, 2) &= (7 \bmod 3, 7 \bmod 5); \\ (1, 3) &= (13 \bmod 3, 13 \bmod 5); \\ (1, 4) &= (4 \bmod 3, 4 \bmod 5); \\ (2, 1) &= (11 \bmod 3, 11 \bmod 5); \\ (2, 2) &= (2 \bmod 3, 2 \bmod 5); \\ (2, 3) &= (8 \bmod 3, 8 \bmod 5); \\ (2, 4) &= (14 \bmod 3, 14 \bmod 5); \end{aligned}$$

Como cada um dos $\mathbb{Z}_{n_i}^*$ é cíclico, \mathbb{Z}_n^* pode ser expresso como produto cartesiano de subgrupos cíclicos.

Exemplo:

⁵A definição de isomorfismo de grupo pode ser vista na pág. 30.

$$\mathbb{Z}_3^* = \{2^0, 2^1\}, \quad \mathbb{Z}_5^* = \{2^0, 2^1, 2^2, 2^3\} \text{ e } \mathbb{Z}_{15}^* = \{1, 2, 4, 7, 8, 11, 13, 14\}.$$

$$\mathbb{Z}_3^* \times \mathbb{Z}_5^* = \{(2^0, 2^0), (2^0, 2^1), (2^0, 2^3), (2^0, 2^2), (2^1, 2^0), (2^1, 2^1), (2^1, 2^3), (2^1, 2^2)\},$$

$$(2^0, 2^0) = (1 \bmod 3, 1 \bmod 5);$$

$$(2^0, 2^1) = (7 \bmod 3, 7 \bmod 5);$$

$$(2^0, 2^3) = (13 \bmod 3, 13 \bmod 5);$$

$$(2^0, 2^2) = (4 \bmod 3, 4 \bmod 5);$$

$$(2^1, 2^0) = (11 \bmod 3, 11 \bmod 5);$$

$$(2^1, 2^1) = (2 \bmod 3, 2 \bmod 5);$$

$$(2^1, 2^3) = (8 \bmod 3, 8 \bmod 5);$$

$$(2^1, 2^2) = (14 \bmod 3, 14 \bmod 5);$$

Capítulo 3

Modelo Computacional Quântico

There must be a history of the universe in which Belize won every gold medal at the Olympic Games, though maybe the probability is low.

S. W. Hawking.

3.1 Introdução

O objetivo deste capítulo é apresentar modelos de portas quânticas e sua utilização em circuitos computacionais quânticos. Inicialmente é discutido o modelo matemático de sistemas mecânicos quânticos em espaços de Hilbert e com base na formalização obtida são discutidos o modelo teórico de qubits e de registradores quânticos. A seguir são definidas as portas lógicas quânticas e apresentados fundamentos de circuitos computacionais quânticos.

3.2 Visão geral

Neste capítulo, o formalismo matemático é gradualmente simplificado à medida que os objetos a serem definidos ficam restritos àqueles que compõem sistemas computacionais quânticos.

- um exemplo de sistema mecânico quântico empregado na descrição do comportamento esperado de qubits e registradores quânticos são as partículas com spin $\frac{1}{2}$;
- espaços de Hilbert de dimensão finita podem ser utilizados na formalização matemática de sistemas computacionais quânticos;

- um espaço de Hilbert de dimensão n , denotado H_n , corresponde ao *espaço de estados* de um sistema mecânico quântico com n níveis;
- o estado de um sistema quântico com n níveis corresponde a um *operador de densidade* em H_n ;
- um *operador de densidade* é um operador auto-adjunto, positivo e com traço unitário;
- mudanças de estado ao longo do tempo correspondem à atuação de *operadores unitários*;
- um estado quântico é medido indiretamente por meio de um operador auto-adjunto invariante no tempo, denominado *observador*;
- a atuação de um *observador* afeta o estado quântico;
- o formalismo aqui apresentado, segundo o qual estados quânticos variam no tempo por meio da ação de operadores unitários, enquanto os observadores são representados por operadores auto-adjuntos fixos no tempo é denominado *Schroedinger picture*;
- a projeção unidimensional do estado um sistema quântico é denominada *estado puro*;
- o estado quântico de um sistema com n níveis pode ser definido como um vetor ψ em H_n .
- um qubit é um sistema quântico com dois níveis e com uma base fixa, denominada base computacional;
- uma porta lógica quântica unária é um mapeamento unitário $U : H_2 \rightarrow H_2$;
- uma porta lógica quântica binária é um mapeamento unitário $U : H_4 \rightarrow H_4$;

3.3 Mecânica Quântica em espaços de Hilbert

Vimos no capítulo 2 que a determinação $r = \text{ord}_n(a)$ é a parte mais difícil do algoritmo 2.1.1 para obtenção de fatores não triviais. Nesta seção vamos definir esse problema.

Capítulo 4

Algoritmo de Shor

Despite the great difficulty of constructing a truly general-purpose quantum computer, it might be relatively easy to construct a special-purpose quantum factoring machine which could be used for code-breaking. History does have a tendency to repeat itself; were not the first digital computers used for code-breaking?

S. Y. Yan.

4.1 Introdução

O objetivo deste capítulo é apresentar detalhadamente a rotina quântica que é a base do algoritmo de Shor. Inicialmente será formulado o problema específico tratado por essa rotina. Em seguida será mostrada a aplicação da transformada de Fourier, que é seu elemento principal, e discutida sua probabilidade de acerto.

4.2 Visão geral

Uma das estratégias usadas em algoritmos quânticos consiste na aplicação de transformações em estados superpostos, aproveitando o paralelismo que pode ser induzido nos registradores quânticos. Mecanismos como a transformada de Fourier são utilizados para obter informações específicas, ou mesmo para induzir o aumento da probabilidade de obtenção de medições em que se está interessado. No caso do algoritmo de Shor, limitantes para as probabilidades envolvidas podem ser calculados com boa aproximação.

- uma rotina quântica para encontrar o período de funções exponenciais modulares é o núcleo do algoritmo de Shor;

- o período de funções $f(k) = a^k \bmod n$ está relacionado com a ordem $\text{ord}_n(a)$ de elementos a de \mathbb{Z}_n^* (elementos de \mathbb{Z}_n não nulos, tais que $\text{mdc}(a, n) = 1$);
- foi visto no capítulo 2, que um divisor não trivial de n pode ser obtido se $r = \text{ord}_n(a)$ for par e $a^{\frac{r}{2}} \not\equiv 1 \bmod n$;
- será mostrado que a probabilidade de que essas duas condições ocorram simultaneamente para um número $a \in \mathbb{Z}_n^*$ escolhido aleatoriamente em uma distribuição de probabilidades uniforme é maior que $\frac{9}{16}$;
- a rotina quântica para encontrar o período de $f(k) = a^k \bmod n$ emprega um par de registradores quânticos $|x_1\rangle |x_2\rangle$;
- o primeiro registrador, $|x_1\rangle$, é usado para armazenar o domínio de $f(k) = a^k \bmod n$ e, posteriormente, para armazenar a transformada discreta inversa de Fourier;
- o segundo registrador, $|x_2\rangle$, é usado para armazenar a imagem de $f(k) = a^k \bmod n$;
- os registradores ficam inicialmente no estado $|0\rangle |0\rangle$;
- no primeiro registrador é induzida, pela aplicação de uma transformada Hadamard-Walsh, uma superposição de estados contendo elementos de um conjunto \mathbb{Z}_{2^l} , $n^2 \leq 2^l < 2n^2$, expressos em codificação binária (etapa denominada randomização do registrador);
- os registradores vão para o estado $|k\rangle |0\rangle$, onde $|k\rangle$ contém, superpostos em seus l qubits, todos os valores de k em $[0, 2^l - 1]$;
- um circuito quântico (concatenação de portas quânticas que implementam uma transformação unitária) em l qubits lê o primeiro registrador, processa a função $f(k) = a^k \bmod n$ e armazena a superposição resultante nos j qubits do segundo registrador, sendo $l \geq j \geq \lceil \lg n \rceil$;
- os registradores vão para o estado $|k\rangle |f(k)\rangle$;
- um circuito quântico em j qubits processa a transformada discreta inversa de Fourier, armazenando o resultado no primeiro registrador;
- será mostrado o procedimento para aplicar a transformada discreta de Fourier como transformação unitária;
- os registradores vão para o estado $| \text{tdif} \rangle |f(k)\rangle$;
- é feita uma observação no dois registradores, que colapsam em algum estado $|p\rangle |a^k \bmod n\rangle$, onde $p \in \mathbb{Z}_{2^l}$;

- usando o algoritmo de Euclides são obtidos os convergentes $\frac{p_i}{q_i}$ de $\frac{p}{2^l}$;
- a ordem r é o menor q_i tal que $a^{q_i} \equiv 1 \pmod n$, se tal q_i existir;
- será mostrado que esse método é correto.

4.3 Determinação da Ordem

Vimos no capítulo 2 que a determinação de $r = \text{ord}_n(a)$ é a parte mais difícil do algoritmo 2.1.1. Nesta seção vamos definir esse problema, mostrar um limitante inferior para a probabilidade de obtenção de uma ordem adequada ao procedimento do algoritmo 2.1.1 e finalmente relacionar a ordem $r = \text{ord}_n(a)$ ao período da função $f(k) = a^k \pmod n$.

4.3.1 Definição do Problema de Determinação da Ordem

Problema DETERMINAÇÃO-DA-ORDEM(n, a): dado um inteiro n ímpar positivo com pelo menos dois fatores primos distintos e um natural $a < n$ tal que $\text{mdc}(a, n) = 1$, encontre o menor natural k tal que $a^k \equiv 1 \pmod n$. Tal k é denominado ordem de a módulo n , ou ordem de a em \mathbb{Z}_n^* e denotado $\text{ord}_n(a)$.

4.3.2 Probabilidade de Obtenção de uma Ordem Adequada

O algoritmo 2.1.1 obtém um divisor não trivial de n se $r = \text{ord}_n(a)$ for par e $a^{\frac{r}{2}} \not\equiv 1 \pmod n$. Será demonstrado que a probabilidade de que isso ocorra para $a \in \mathbb{Z}_n^*$ escolhido de maneira aleatória numa distribuição de probabilidade uniforme (isto é, todo $z \in \mathbb{Z}_n^*$ tem igual probabilidade de ser escolhido), tem limitante inferior $\frac{9}{16}$.

Cardinalidade de \mathbb{Z}_n^*

Definição 4.3.1 (Função $\phi(n)$ de Euler). Para um inteiro positivo n , $\phi(n)$ é igual ao número de inteiros positivos z menores que n tais que z e n são primos entre si:

$$\phi(n) = \sum_{\substack{1 \leq z < n \\ \text{mdc}(z, n) = 1}} 1.$$

Exemplos:

n	1	2	3	4	5	6	7	8	9	10	15	21
$\phi(n)$	1	1	2	2	4	2	6	4	6	4	8	12

Os elementos de grupo de \mathbb{Z}_n^* são *cosets* $z + n\mathbb{Z}$, $z \in \{1, \dots, n-1\}$, do subgrupo $n\mathbb{Z}$, tais que $\text{mdc}(z, n) = 1$. O número de elementos de grupo de \mathbb{Z}_n^* , denominado cardinalidade de \mathbb{Z}_n^* e denotado $|\mathbb{Z}_n^*|$ é $\phi(n)$.

Se n é ímpar, composto e sua fatoração em primos é $n = p_1^{e_1} \dots p_k^{e_k}$, sabemos que

$$\mathbb{Z}_n^* \cong \mathbb{Z}_{p_1^{e_1}}^* \times \mathbb{Z}_{p_2^{e_2}}^* \times \dots \times \mathbb{Z}_{p_k^{e_k}}^*,$$

então

$$|\mathbb{Z}_n^*| = \left| \mathbb{Z}_{p_1^{e_1}}^* \right| \left| \mathbb{Z}_{p_2^{e_2}}^* \right| \dots \left| \mathbb{Z}_{p_k^{e_k}}^* \right|,$$

que pode ser escrito como

$$\phi(n) = \phi(p_1^{e_1}) \phi(p_2^{e_2}) \dots \phi(p_k^{e_k}),$$

que é igual a

$$\phi(n) = p_1^{e_1} \left(1 - \frac{1}{p_1}\right) \dots p_k^{e_k} \left(1 - \frac{1}{p_k}\right) = n \left(1 - \frac{1}{p_1}\right) \dots \left(1 - \frac{1}{p_k}\right).$$

portanto

$$|\mathbb{Z}_n^*| = n \left(1 - \frac{1}{p_1}\right) \dots \left(1 - \frac{1}{p_k}\right).$$

Ordem dos Elementos de \mathbb{Z}_n^*

Seja $\mathbb{Z}_{p_i^{e_i}}^* = \left\{ z_{p_i}^0, z_{p_i}^1, \dots, z_{p_i}^{\phi(p_i^{e_i})-1} \right\}$, onde z_{p_i} é um elemento gerador do grupo cíclico $\mathbb{Z}_{p_i^{e_i}}^*$. Sabemos que $\left| \mathbb{Z}_{p_i^{e_i}}^* \right| = \phi(p_i^{e_i}) = p_i^{e_i} (p_i - 1)$ é par, que a ordem de $z_{p_i}^0$ é 1, que a ordem de $z_{p_i}^1$ é $\phi(p_i^{e_i})$, que a ordem de $z_{p_i}^j$ é $\frac{\phi(p_i^{e_i})}{\text{mdc}(j, \phi(p_i^{e_i}))}$ e que a ordem de $z_{p_i}^{\phi(p_i^{e_i})-1}$ é $\phi(p_i^{e_i})$, (pois $\text{mdc}(a, a-1) = 1$).

Exemplo: $\mathbb{Z}_7^* = \{3^0, 3^1, 3^2, 3^3, 3^4, 3^5\}$ e $\phi(7) = 6$,

$$3^{0^1} = 1 \equiv 1 \pmod{7}$$

$$3^{1^6} = 729 = 104 \cdot 7 + 1 \equiv 1 \pmod{7}$$

$$3^{2^{\frac{6}{2}}} = 729 = 104 \cdot 7 + 1 \equiv 1 \pmod{7}$$

$$3^{3^{\frac{6}{3}}} = 729 = 104 \cdot 7 + 1 \equiv 1 \pmod{7}$$

$$3^{4^{\frac{6}{2}}} = 531441 = 75920 \cdot 7 + 1 \equiv 1 \pmod{7}$$

$$3^{5^{\frac{6}{1}}} = 2.058911320946490e14 = 2.941301887066400e13 \cdot 7 + 1 \equiv 1 \pmod{7}$$

Dado um elemento de $\left\{z_{p_i}^0, z_{p_i}^1, \dots, z_{p_i}^{\phi(p_i^{e_i})-1}\right\}$, a probabilidade de que sua ordem seja ímpar é $\frac{1}{2}$.

Vimos no capítulo 2, que o conjunto de elementos de grupo de \mathbb{Z}_n^* pode ser escrito na forma:

$$\left\{z_{p_1}^0, z_{p_1}^1, \dots, z_{p_1}^{\phi(p_1^{e_1})-1}\right\} \times \dots \times \left\{z_{p_i}^0, z_{p_i}^1, \dots, z_{p_i}^{\phi(p_i^{e_i})-1}\right\} \times \dots \times \left\{z_{p_k}^0, z_{p_k}^1, \dots, z_{p_k}^{\phi(p_k^{e_k})-1}\right\}$$

Cada elemento z de \mathbb{Z}_n^* pode ser decomposto numa k -upla de elementos de subgrupos cíclicos: $z = (z_{p_1}^{\alpha_1}, \dots, z_{p_i}^{\alpha_i}, \dots, z_{p_k}^{\alpha_k})$.

Seja $r_i = \text{ord}_{p_i^{e_i}}(z_{p_i}^{\alpha_i})$. A ordem de $z \in \mathbb{Z}_n^*$ é $\text{mmc}(r_1, \dots, r_i, \dots, r_k)$.

Para que $\text{mmc}(r_1, \dots, r_i, \dots, r_k)$ seja ímpar é necessário que cada r_i seja ímpar. Portanto a probabilidade de que $r = \text{ord}_n(z)$ seja par é $(1 - \frac{1}{2^k})$.

Além de par, a ordem r adequada deve ser tal que $z^{\frac{r}{2}} \not\equiv 1 \pmod{n}$.

Sabemos que $r = \text{mmc}(r_1, \dots, r_i, \dots, r_k)$, logo todo r_i divide r . Suponha que exista r_i que divida $\frac{r}{2}$. Nesse caso teríamos $z^{\frac{r}{2}} \equiv 1 \pmod{p_i^{e_i}}$. No entanto, $z^{\frac{r}{2}} \equiv -1 \pmod{n}$ implica que $z^{\frac{r}{2}} \equiv -1 \pmod{p_i^{e_i}}$. Essas duas condições só podem ser atendidas se $p_i = 2$. Como n é ímpar não existe r_i que divida $\frac{r}{2}$. Portanto, sendo $r = 2^\beta t$, todo $r_i = 2^\beta t_i$, onde t e t_i são ímpares. A probabilidade que isso ocorra para cada i é menor ou igual a $\frac{1}{2}$, ou seja, a probabilidade de que $z^{\frac{r}{2}} \equiv 1 \pmod{n}$ é menor ou igual a $\frac{1}{2^k}$.

Portanto a probabilidade de que se obtenha uma ordem adequada para um elemento $a \in \mathbb{Z}_n^*$ escolhido de maneira aleatória numa distribuição de probabilidade uniforme é $P \geq (1 - \frac{1}{2^k})^2$, que é maior ou igual a $\frac{9}{16}$, pois $k \geq 2$.

4.3.3 Ordem como Período da Função $f(k) = a^k \pmod{n}$

Escolhendo aleatoriamente $a \in \mathbb{Z}_n^*$ sabemos qual a probabilidade de que $r = \text{ord}_n(a)$ seja adequada para a determinação de fatores não-triviais de n . Para a determinação de r temos o algoritmo 2.1.5. Uma outra maneira de determinar r é calcular o período da função $f(k) = a^k \pmod{n}$.

4.4 Transformada de Fourier Discreta

Nesta seção será vista a transformada de Fourier clássica e sua utilização na determinação do período de funções; a transformada de Fourier quântica e o modo de aplicá-la sobre registradores quânticos.

4.4.1 Caracteres

Numa descrição informal, caracter é um funcional multiplicativo de um grupo para os números complexos. Nesta seção serão vistos caracteres de grupos abelianos e sua relação com a transformada de Fourier discreta.

Definição 4.4.1 (Morfismo de Grupos). Sejam \mathcal{G} e \mathcal{H} grupos multiplicativos. Um morfismo de grupo entre \mathcal{G} e \mathcal{H} é um mapeamento $f : \mathcal{G} \rightarrow \mathcal{H}$ tal que $f(g_1 g_2) = f(g_1) f(g_2)$ para todo $g_1, g_2 \in \mathcal{G}$.

Como $f(1) = f(1 \cdot 1) = f(1) f(1)$ então $f(1) = 1$. Por outro lado, $1 = f(1) = f(g g^{-1}) = f(g) f(g^{-1})$ implica que $f(g^{-1}) = f(g)^{-1}$ e por indução $f(g^k) = f(g)^k$ para todo inteiro k .

Definição 4.4.2 (Isomorfismo de Grupos). Um isomorfismo de grupo entre \mathcal{G} e \mathcal{H} é um morfismo de grupo tal que $g_1 \neq g_2 \Rightarrow f(g_1) \neq f(g_2)$ (é injetor) e $f(G) = H$ (é sobrejetor). Portanto um isomorfismo é um mapeamento bijetor.

Dois grupos \mathcal{G} e \mathcal{H} são isomórficos se existe um isomorfismo $f : \mathcal{G} \rightarrow \mathcal{H}$. Nesse caso os dois grupos diferem somente na representação, sendo $g \in G$ correspondente a $f(g) \in H$. A notação utilizada é $\mathcal{G} \cong \mathcal{H}$.

Caracteres de Grupos Abelianos

O caracter de um grupo aditivo abeliano¹ finito G é um morfismo $\chi : G \rightarrow \mathbb{C}_{\neq 0}$, portanto²:

- $\chi(g_1 + g_2) = \chi(g_1) \chi(g_2)$ para todo $g_1, g_2 \in G$;
 - $\chi(0) = 1 + 0i$;
 - $\chi(g)^k = \chi(kg)$;
 - se $n = |G|$, então $\chi(g)^n = \chi(ng) = \chi(0) = 1 + 0i$. Portanto
- $$\chi(g) = \sqrt[n]{1 + 0i}.$$

Definição 4.4.3 (Grupo Caracter ou Grupo Dual). Dados dois caracteres χ_1 e χ_2 , seja $\chi_1 \chi_2 : G \rightarrow \mathbb{C}_{\neq 0}$ o caracter produto, definido por $\chi_1 \chi_2(g) = \chi_1(g) \chi_2(g)$. O grupo $\widehat{G} = (\chi(G), \chi_1(g) \chi_2(g), \chi(g)^{-1}, \chi_0)$ é um grupo abeliano denominado grupo caracter de G ou grupo dual de G . O elemento neutro χ_0 é denominado caracter principal, ou caracter trivial e $\chi_0(g) = 1 + 0i$ para todo $g \in G$.

¹A definição de grupo abeliano, ou comutativo, pode ser vista na pág. 13.

²Lembrando que $\mathbb{C}_{\neq 0}$ é um grupo multiplicativo abeliano onde $e = 1$ e $(*c) = c^{-1}$.

Exemplo: Considere $\mathbb{Z}_5 = \{0 + 5\mathbb{Z}, 1 + 5\mathbb{Z}, 2 + 5\mathbb{Z}, 3 + 5\mathbb{Z}, 4 + 5\mathbb{Z}\}$. \mathbb{Z}_5 é um grupo aditivo abeliano cíclico. Seja $y + 5\mathbb{Z}$ um dado elemento de grupo de \mathbb{Z}_5 . Definindo o morfismo $\chi_y : \mathbb{Z}_5 \rightarrow \mathbb{C}_{\neq 0}$ como $\chi_y(z) = e^{\frac{2\pi i z y}{5}}$ temos:

$$\begin{array}{lll} \chi_0(0) = 1 + 0i & \chi_1(0) = 1 + 0i & \chi_2(0) = 1 + 0i \\ \chi_0(1) = 1 + 0i & \chi_1(1) = 0.3090 + 0.9511i & \chi_2(1) = -0.8090 + 0.5878i \\ \chi_0(2) = 1 + 0i & \chi_1(2) = -0.8090 + 0.5878i & \chi_2(2) = 0.3090 - 0.9511i \\ \chi_0(3) = 1 + 0i & \chi_1(3) = -0.8090 - 0.5878i & \chi_2(3) = 0.3090 + 0.9511i \\ \chi_0(4) = 1 + 0i & \chi_1(4) = 0.3090 - 0.9511i & \chi_2(4) = -0.8090 - 0.5878i \\ \chi_0(5) = 1 + 0i & \chi_1(5) = 1 + 0i & \chi_2(5) = 1 + 0i \end{array}$$

$$\begin{array}{lll} \chi_3(0) = 1 + 0i & \chi_4(0) = 1 + 0i & \chi_5(0) = 1 + 0i \\ \chi_3(1) = -0.8090 - 0.5878i & \chi_4(1) = 0.3090 - 0.9511i & \chi_5(1) = 1 + 0i \\ \chi_3(2) = 0.3090 + 0.9511i & \chi_4(2) = -0.8090 - 0.5878i & \chi_5(2) = 1 + 0i \\ \chi_3(3) = 0.3090 - 0.9511i & \chi_4(3) = -0.8090 + 0.5878i & \chi_5(3) = 1 + 0i \\ \chi_3(4) = -0.8090 + 0.5878i & \chi_4(4) = 0.3090 + 0.9511i & \chi_5(4) = 1 + 0i \\ \chi_3(5) = 1 + 0i & \chi_4(5) = 1 + 0i & \chi_5(5) = 1 + 0i \end{array}$$

Sejam $y, z, z_1, z_2 \in \mathbb{Z}_n = \{0, 1, \dots, n-1\}$. Pode ser verificado que:

- $\chi_y, y \neq 0$ tem período n ;
- $\chi_y(z) = \chi_z(y)$;
- $\chi_y(z_1 + z_2) = \chi_y(z_1)\chi_y(z_2)$;
- $\chi_{z_1}\chi_{z_2} = \chi_{(z_1 + z_2)}$;
- $y \neq z \neq 0 \Rightarrow \chi_y \neq \chi_z$;

Portanto para qualquer $\chi_y, y \neq 0$, o grupo multiplicativo $\widehat{\mathbb{Z}}_n = (\{\chi_y(0), \chi_y(1), \dots, \chi_y(n-1)\}, \chi_y(z)\chi_y(z), \chi_y(z)^{-1}, \chi_0)$ é o grupo dual de \mathbb{Z}_n , sendo também um grupo cíclico e isomorfo a \mathbb{Z}_n .

Exemplo: sejam $\mathbb{Z}_5 = \{0, 1, 2, 3, 4\}$ e $y = 2$, temos

$$\widehat{\mathbb{Z}}_5 = \{\chi_2(0), \chi_2(1), \chi_2(2), \chi_2(3), \chi_2(4)\}$$

ou

$$\widehat{\mathbb{Z}}_5 = \{\chi_0, \chi_2(3), \chi_2(3)^2, \chi_2(3)^3, \chi_2(3)^4\}$$

Onde

$$\begin{array}{ll} \chi_2(0) = 1 + 0i & \chi_0(0) = 1 + 0i \\ \chi_2(1) = -0.8090 + 0.5878i & \chi_2(3) = 0.3090 + 0.9511i \\ \chi_2(2) = 0.3090 - 0.9511i & \chi_2(3)^2 = -0.8090 + 0.5878i \\ \chi_2(3) = 0.3090 + 0.9511i & \chi_2(3)^3 = -0.8090 - 0.5878i \\ \chi_2(4) = -0.8090 - 0.5878i & \chi_2(3)^4 = 0.3090 - 0.9511i \end{array}$$

Base Ortogonal de Caracteres

Seja \mathcal{G} um grupo aditivo abeliano finito com $G = \{g_1, g_2, \dots, g_n\}$. Seja V o espaço vetorial sobre \mathbb{C} formado por funções $f : \mathcal{G} \rightarrow \mathbb{C}$ onde

$\mathbf{f} = (f(g_1), f(g_2), \dots, f(g_n))$ é um vetor em V ,

$\langle \mathbf{f} | \mathbf{h} \rangle = \sum_{i=1}^n f^*(g_i) h(g_i)$ é o produto interno em V ,

$\|\mathbf{f}\| = \sqrt{\langle \mathbf{f} | \mathbf{f} \rangle}$ é a norma de \mathbf{f} em V .

Uma base ortonormal de V pode ser obtida através dos caracteres de \mathcal{G} .

Proposição 4.4.1 (Base Ortonormal de Caracteres).

Seja $\mathcal{B} = \{B_1, B_2, \dots, B_n\}$, onde $B_i = \frac{1}{\sqrt{n}} \chi_i$. Então \mathcal{B} é uma base ortonormal de V .

Demonstração:

$$\langle \chi_i | \chi_j \rangle = \sum_{k=1}^n \chi_i^*(g_k) \chi_j(g_k).$$

Como $\chi^*(g) \chi(g) = |\chi(g)|^2 = 1$ temos que $\chi^*(g) = \chi(g)^{-1}$ para todo g em G .

$$\text{Então } \langle \chi_i | \chi_j \rangle = \sum_{k=1}^n \chi_i(g_k)^{-1} \chi_j(g_k).$$

Se $i = j$, $\chi = \chi_i^{-1} \chi_j$ é um caracter trivial de \mathcal{G} . Então $\langle \chi_i | \chi_j \rangle = n$.

Se $i \neq j$, $\chi = \chi_i^{-1} \chi_j$ é um caracter não-trivial de \mathcal{G} .

Nesse caso, considerando a permutação $g \rightarrow g + g_i$ de G , temos

$$\sum_{k=1}^n \chi(g_k) = \sum_{k=1}^n \chi(g + g_k) = \chi(g) \sum_{k=1}^n \chi(g_k).$$

$$\text{Portanto } \sum_{k=1}^n \chi(g_k) (1 - \chi(g)) = 0.$$

Como χ é não-trivial, existe g tal que $\chi(g) \neq 1$.

$$\text{Então } \sum_{k=1}^n \chi(g_k) = 0. \text{ Logo, se } i \neq j, \langle \chi_i | \chi_j \rangle = 0$$

Exemplo: sejam $\mathbb{Z}_5 = \{0, 1, 2, 3, 4\}$, $\chi_y(z) = e^{\frac{2\pi i z y}{5}}$, e

$\mathcal{B} = \{B_0, B_1, B_2, B_3, B_4\}$ onde $B_i = (\chi_i(0), \chi_i(1), \chi_i(2), \chi_i(3), \chi_i(4)) :$

$$\begin{aligned}
B_0 &= \left(\frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}} \right) \\
B_1 &= \left(\frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}} e^{\frac{2\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{4\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{6\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{8\pi i}{5}} \right) \\
B_2 &= \left(\frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}} e^{\frac{4\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{8\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{12\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{16\pi i}{5}} \right) \\
B_3 &= \left(\frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}} e^{\frac{6\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{12\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{18\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{24\pi i}{5}} \right) \\
B_4 &= \left(\frac{1}{\sqrt{5}}, \frac{1}{\sqrt{5}} e^{\frac{8\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{16\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{24\pi i}{5}}, \frac{1}{\sqrt{5}} e^{\frac{32\pi i}{5}} \right)
\end{aligned}$$

4.4.2 Transformada de Fourier Discreta Clássica

Seja

$$\mathbf{f} = f(g_1)\mathbf{e}_1 + f(g_2)\mathbf{e}_2 + \cdots + f(g_n)\mathbf{e}_n$$

a representação de f na base ortonormal

$$\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n\}$$

onde

$$\mathbf{e}_1 = (1, 0, \dots, 0), \mathbf{e}_2 = (0, 1, \dots, 0), \dots, \mathbf{e}_n = (0, 0, \dots, 1).$$

Seja

$$\mathbf{f} = \widehat{f}(g_1)B_1 + \widehat{f}(g_2)B_2 + \cdots + \widehat{f}(g_n)B_n$$

a representação de f na base ortonormal de caracteres \mathcal{B} .

A n -upla $(\widehat{f}(g_1), \widehat{f}(g_2), \dots, \widehat{f}(g_n))$ é a transformada de Fourier discreta - tdf - de $(f(g_1), f(g_2), \dots, f(g_n))$.

Quando aplicada ao vetor \mathbf{f} , a matriz M de mudança da base $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n\}$ para a base $\{B_1, B_2, \dots, B_n\}$ produz a tdf de f :

$$\begin{pmatrix} \widehat{f}(g_1) \\ \widehat{f}(g_2) \\ \vdots \\ \widehat{f}(g_n) \end{pmatrix} = \frac{1}{\sqrt{n}} \begin{pmatrix} \chi_1^*(g_1) & \chi_1^*(g_2) & \cdots & \chi_1^*(g_n) \\ \chi_2^*(g_1) & \chi_2^*(g_2) & \cdots & \chi_2^*(g_n) \\ \vdots & \vdots & & \vdots \\ \chi_n^*(g_1) & \chi_n^*(g_2) & \cdots & \chi_n^*(g_n) \end{pmatrix} \begin{pmatrix} f(g_1) \\ f(g_2) \\ \vdots \\ f(g_n) \end{pmatrix}$$

Exemplo: seja $\mathbf{f} = (1, i, -1, -i, 1)'$.

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & e^{-\frac{2\pi i}{5}} & e^{-\frac{4\pi i}{5}} & e^{-\frac{6\pi i}{5}} & e^{-\frac{8\pi i}{5}} \\ 1 & e^{-\frac{4\pi i}{5}} & e^{-\frac{8\pi i}{5}} & e^{-\frac{12\pi i}{5}} & e^{-\frac{16\pi i}{5}} \\ 1 & e^{-\frac{6\pi i}{5}} & e^{-\frac{12\pi i}{5}} & e^{-\frac{18\pi i}{5}} & e^{-\frac{24\pi i}{5}} \\ 1 & e^{-\frac{8\pi i}{5}} & e^{-\frac{16\pi i}{5}} & e^{-\frac{24\pi i}{5}} & e^{-\frac{32\pi i}{5}} \end{pmatrix} \begin{pmatrix} 1 \\ i \\ -1 \\ -i \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0.5792 + 0.4208i \\ 0.2452 + 0.7548i \\ -0.4813 + 1.4813i \\ 3.6569 - 2.6569i \end{pmatrix}$$

Dado um vetor $v = [f(0) \ f(1) \ \dots \ f(2^l - 1)]$ com 2^l amostras de uma função $f(\cdot), f: \mathbb{Z} \rightarrow \mathbb{C}$, a transformada de Fourier discreta clássica é o vetor $\hat{v} = [\hat{f}(0) \ \hat{f}(1) \ \dots \ \hat{f}(2^l - 1)]$ onde

$$\hat{f}(x) = \frac{1}{\sqrt{2^l}} \sum_{y=0}^{2^l-1} e^{-\frac{2\pi i xy}{2^l}} f(y)$$

4.4.3 Transformada de Fourier e Periodicidade

Se $f(\cdot), f: \mathbb{Z} \rightarrow \mathbb{C}$ é periódica de período p

Bibliografia

- [1] N. I. Akhiezer and I. M. Glazman, *Theory of Linear Operators in Hilbert Space*, Dover Books on Mathematics, Dover Publications Inc., New York, 1961.
- [2] M. A. Aikvis and V. V. Goldberg, *An introduction to Linear Algebra and Tensors*, Dover Books on Mathematics, Dover Publications Inc., New York, 1972.
- [3] G. Brassard and P. Hoyer, *An exact quantum polynomial-time algorithm for Simon's problem*, Proceedings of the 1997 Israeli Symposium on Theory of Computing and Systems - ISTCS'97 (1997), 12–23, <http://xxx.lanl.gov/abs/quant-ph/9704027>.
- [4] T.H. Cormen, C.E. Leiserson, and R.L. Rivest, *Introduction to algorithms*, The MIT Electrical Engineering and Computer Science Series, The MIT Press, Cambridge, Massachusetts, 1990.
- [5] D. Deutsch and R. Jozsa, *Rapid solutions of problems by quantum computation*, Proceedings of the Royal Society of London **A-439** (1992), 553–558.
- [6] C. G. Fernandes, F. K. Miyazawa, M. Cerioli, and P. Feofiloff (eds.), *Uma introdução sucinta a Algoritmos de Aproximação*, Publicações Matemáticas - 23 Colóquio Brasileiro de Matemática, IMPA - Instituto de matemática Pura e Aplicada, Rio de Janeiro, Brasil, 2001.
- [7] R. P. Feynman, *Feynman Lectures on Computation*, The Advanced Book Program, Perseus Publishing, Boston, Massachusetts, 2000.
- [8] J. Gruska, *Quantum Computing*, Advanced Topics in Computer Science Series, McGraw-Hill, Berkshire, England, 1999.
- [9] M. Hirvensalo, *Quantum Computing*, Natural Computing Series, Springer-Verlag, Berlin, 2001.
- [10] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages and Computation*, second ed., Addison-Wesley, Boston, Massachusetts, 2001.

-
- [11] I. M. Isaacs, *Character theory of Finite Groups*, Dover Books on Mathematics, Dover Publications Inc., New York, 1961.
 - [12] F. C. P. Milies and S. P. Coelho, *Números, uma introdução à Matemática*, Série Acadêmica, Edusp - Editora da Universidade de São Paulo, São Paulo, Brasil, 1998.
 - [13] A. O. Pittenger, *An introduction to Quantum Computing Algorithms*, Progress in Computer Science and Applied Logic, Birkhäuser, Boston, Massachusetts, 1999.
 - [14] P. W. Shor, *Algorithms for quantum computation: discrete log and factoring*, Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science (1994), 20–22.
 - [15] ———, *Polynomial-time Algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Journal on Computing **26** (1997), no. 5, 1484–1509.
 - [16] S. Y. Yan, *Number Theory for Computing*, Springer-Verlag, Berlin, Germany, 2000.
 - [17] N. Young, *An introduction to Hilbert Space*, Cambridge Mathematical Textbooks, Cambridge University Press, Cambridge, United Kingdom, 1988.

Índice

- \mathbb{Z}_n
 - congruência em, 15
 - representação de, 15
- \mathbb{Z}_n^*
 - cardinalidade de, 27
 - notação de ordem, 19
 - ordem dos elementos de, 28
- álgebra linear, iv
- cosets*, 14
 - inversão de, 16
 - operação entre, 14
 - produto de, 16
- algoritmo
 - exponencial para fatoração, 11
 - de Euclides, 11
 - exponenciação modular, 12
 - exponencial para ordem, 12
 - quântico - estratégias, 25
- Caracteres
 - base ortogonal de, 32
 - de grupos abelianos, 30
- caracteres, 30
- congruência, 14
- fatoração em primos completa, 7
- grupo, 13
 - aditivo
 - \mathbb{Z}_n , 14
 - cíclico, 17
 - notação de, 13
 - character, 30
 - definição, 13
 - dual, 30
 - elemento identidade, 13
 - fator, 14
 - multiplicativo
 - \mathbb{Z}_n^* , 17
 - cíclico, 17
 - notação de, 13
 - notação de, 13
 - operação binária de, 13
 - operação unária de, 13
- identificação de fator primo, 8
- isomorfismo de grupos, 21, 30
- isomorfo, 21
- máximo divisor comum, 9
- morfismo de grupos, 30
- ordem
 - como período, 29
- probabilidade de ordem adequada, 27
- problema computacional, 2
- problema concreto de decisão, 4
- subgrupo, 14
 - normal, 14
- tdf, 29
 - clássica, 33
 - e periodicidade, 34
- teorema fund. da aritmética, 7
- teoria dos números, iv, 7