Parcelamento e Troca de Valores de Entrada no Método Binário de Congruência

Mauri J. Klein[1], Gerson Battisti[2], Luander A. Stein[3]
[123] Universidade Regional do Noroeste do Rio Grande do Sul (UNIJUÍ).
Rua São Francisco, 501 - CEP 98700-000 – Ijuí – RS – Brazil

[123] Departamento de Ciências Exatas e Engenharias - DCEENG
{mauri.klein, battisti, luander.stein}@unijui.edu.br

Resumo: *Neste artigo, será apresentado o método binário para verificação de congruência de grandes números, como parte do teste de primalidade. Decompor um número de mais 17 milhões de dígitos ou confirmá-lo como número primo não é trivial. Com o advento da computação e a capacidade de processamento das máquinas, os estudos vêm evoluindo com muita rapidez, porém o custo computacional ainda é muito grande, levando-se em consideração, por exemplo, o teste de primalidade de* $2^{57885161}$*-1. Como resultado serão apresentadas duas alterações no algoritmo: o parcelamento do expoente binário com o intuito de reduzir o número de iterações; e a troca do resto da congruência priorizando sempre o de menor valor, reduzindo o* r_{max} *(resto máximo) em menos da metade em relação a proposta inicial do algoritmo.*

Palavras-Chave: *primos, perfeitos, parcela, binário, congruência*

1. INTRODUÇÃO

Enquanto Dario I rei da Pérsia (522 – 486 A.C) contava os dias através de nós em barbantes e outros povos ainda mais antigos controlavam suas informações contábeis representadas em peças de diferentes formatos e armazenadas em urnas de barro [6], temos hoje acessibilidade a computadores com grande poder computacional capazes de fazer cálculos com grande rapidez.

Neste contexto podemos dizer que os *números primos* desafiam a capacidade do ser humano. A singularidade destes números é fascinante e matemáticos de todas as partes do mundo buscam explicações, novas teorias, conjecturas ou tentam provar algumas já existentes. Este estudo já se estendeu para outras áreas de pesquisa, e neste âmbito, cientistas da computação são potencialmente capazes e de fundamental importância para a mesma, devido à complexidade dos cálculos que precisam ser feitos.

Outra classe de números especiais está diretamente ligada com os números primos: *os números perfeitos*. Estes são gerados através da multiplicação tendo como um dos fatores um número primo gerado pela equação de *mersenne*. Sabendo-se que um número de *mersenne* é primo, pode-se obter um número perfeito [7].

2. A HISTÓRIA DOS NÚMEROS

A história dos números é muito antiga e está contada no livro *A História Universal dos Algarismos* do Autor Georges Ifrah, e vinha sofrendo alterações e melhorias ao longo dos séculos. A evolução ocorreu lentamente e em épocas permanecia estagnada até o surgimento de uma nova linha de pensamento que trazia novas ideias e teorias [6].

Por volta de 300 A.C. na Grécia Antiga surgiram os primeiros conceitos de números primos. Euclides afirma na definição 11 do Livro VII dos Elementos: *"Um número primo é aquele que é medido apenas pela unidade"* [4]. Na definição 13 do Livro dos Elementos, Euclides define também os números compostos. "*Um número composto é aquele que é medido por algum número"*.

Coutinho (2004) traduz estes conceitos para os dias de hoje, e define a expressão 'medido por' como 'divisível por um numero menor do que ele próprio'[3]. Assim podemos definir como primos os números que não são divisíveis por nenhum número menor que ele, exceto o 1.

2.1 Números Primos

Já na metade do século XIX, foi feito um grande progresso nas pesquisas relacionadas à teoria dos números. Bernhard Riemann passou a abordar o problema de uma forma completamente nova. Utilizando uma nova perspectiva, começou a compreender parte do padrão responsável pelo caos dos primos. Havia uma harmonia sutil e inesperada escondida sob o ruído externo dos primos. Apesar deste grande salto adiante, a nova música ainda ocultava muitos de seus segredos.

Riemann foi audacioso e fez uma previsão ousada sobre a melodia misteriosa que havia descoberto. Essa previsão ficou conhecida como a hipótese de Riemann. Quem conseguir provar que a intuição de Riemann sobre a natureza dessa música estava correta terá explicado por que os primos nos transmitem uma impressão tão convincente de aleatoriedade [9].

Riemann desenvolveu sua idéia original após descobrir um espelho matemático através do qual era possível observar os primos. Hoje, o problema proposto em 1859 por Bernhard Riemann faz parte de um seleto grupo de problemas não resolvidos, e para quem solucionar e apresentar a prova da Hipótese de Riemann será pago um prêmio de um milhão de Dólares oferecido em 2000 por Clay Mathematics Institute [2].

Outras pesquisas e as novas descobertas foram feitas com o passar dos anos e com isso, os números primos vieram a ter uma relevante importância para a Ciência da Computação. Alguns algoritmos e estruturas de dados se baseiam nos números primos como é o caso das tabelas Hash [8].

A partir de 1970, devido ao conceito de criptografia de chave-pública, passaram a formar a base dos primeiros algoritmos de criptografia, como exemplo, o algoritmo cryptosystem da RSA [10].

2.2 Números perfeitos

Paralelamente aos números primos, apresentam-se números ainda mais peculiares: *os números perfeitos.* Os números perfeitos foram caracterizados 300 A.C. por Euclides, que definiu um número como perfeito quando a soma de todos os seus divisores é igual ao próprio número. Por volta do ano 600, o PE. Frances Marin *Mersenne* juntamente com Pierre de Fermat, definiram a classe dos números de *mersenne*, como sendo da forma 2^n -1. Para todo n=primo, que gere um número de *mersenne* também primo, obtêm-se um número perfeito relacionado, ou seja, 2^n -1*2^{n-1} = número perfeito [5];

Em artigo publicado em 2010, os pesquisadores chineses Sibao Zhang, Xiaocheng Ma e Lihang Zhou desenvolveram um estudo sobre a distribuição dos números primos de *mersenne*, apresentando uma tabela de probabilidade de incidência em um determinado intervalo de números [11].

n	$x = log_2(n) - 1$	$exp = 2^{2^x}$	*Quantidade de nº primos de *mersenne* = $n - log_2(n)$
2	0	2	1
4	1	4	2
8	2	16	5
16	3	256	12
32	4	65.536	27
64	5	4.294.967.296	58

*Quantidade de números primos de *mersenne* menores que 2^{exp}.

Tabela 1: Dedução da distribuição dos números primos de *mersenne* baseada na conjectura de Zhang, Ma e Zhou.

Fonte: Próprio autor

Observando esta distribuição, fica evidente a infinitude dos números primos de *mersenne* e por conseqüência os números perfeitos. No entanto, considerando que a incidência de números primos de mersenne apresentada pelos pesquisadores é apenas uma conjectura, esta infinitude também não pode ser provada.

DOI: 10.5540/03.2014.002.01.0020

O maior número primo encontrado até hoje, que é também um número primo de *mersenne*, tem mais de 17 milhões de dígitos ($2^{57885161}$-1). Isto está diretamente relacionado com o algoritmo de teste de primalidade utilizado para estes números. O teste de primalidade de Lucas-Lehmer é de longe o mais eficiente e com menor custo computacional, mas apenas se aplica aos números de *mersenne* [7].

Este número "especial" foi encontrado em 2012 por uma rede voluntária através de um sistema de computação distribuída denominada: Great Internet *Mersenne* Prime Search [5]. Ainda não é conhecido nenhum número perfeito impar e conjectura-se que eles não existam, porém a prova não é trivial.

No entanto, outros algoritmos são utilizados para testes preliminares com a finalidade de eliminar a grande maioria dos números compostos com características fortes de números primos. Um destes algoritmos está descrito na seção seguinte.

3. MÉTODO BINÁRIO DE CONGRUÊNCIAS

Percebe-se, que os números em questão são de grandes magnitudes, de difícil manipulação e custosos computacionalmente. O GIMPs utiliza alguns métodos para o garimpo de números primos de *mersenne*. Um destes métodos é a congruência através da transformação de p=(expoente de base 2) em número binário, e posteriormente a verificação da sua divisibilidade por números primos pequenos, descrito por ATKIN e BERNSTEIN em 2004 [1].

Todos os possíveis divisores de números de *mersenne* são da forma $p*k+1$. Onde k é sempre um número par. Por exemplo, para chegar-se ao resto da divisão de 2^p por $p*k+1$, temos o seguinte pseudo-algoritmo:
Considerando:
p=23=10111(binário);
k=*2*;
r=1 (inicial);

1º PASSO: $r = r^2$
2º PASSO: Remover o primeiro bit de *p.*
3º PASSO: Caso o bit removido seja "1", multiplica-se o valor do r^2 por 2 ($r = r * 2$).
Caso o bit removido seja "0" mantêm o valor do r^2.
4º PASSO: Efetua-se a congruência do r por $p * 2 + 1$, atribuindo este valor a r;

Este procedimento deve ser efetuado até que não tenha mais bits a serem removidos, isto é, a quantidade de bits define a quantidade de iterações.
Caso a congruência seja 1, após a realização completa das iterações, o número em questão é um número COMPOSTO, pois ele tem divisão exata.

r^2	Bits Restantes	Remover bit	Multiplicação por 2	$r = Mod.47$
1 x 1=1 *	10111	~~1~~0111	1 x 2 = 2	2
2 x 2=4	0111	~~10~~111	-	4
4 x 4=16	111	~~101~~11	16 x 2 = 32	32
32 x 32=1024	11	~~1011~~1	1024 x 2 = 2048	27
27 x 27=729	1	~~10111~~	729 2 = 1458	1

*Obs.: Inicia-se com resto 1.

Tabela 2: Congruência pelo método binário

Se observarmos no quadro, podemos ver que temos números de tamanho reduzido, ou seja, como é efetuada a congruência após cada iteração, podemos concluir que este número jamais será maior que $((p * k)2) * 2$. Ao passo que uma exponenciação simples geraria um valor com grandes dimensões.

Considerando o exemplo acima, com $\boldsymbol{p}$ **= 23** e $\boldsymbol{k}$**=2**: *Valor máximo possível pelo método Binário:* ***4232****. Valor máximo pela exponenciação simples:* ***8388608****;*

Esta diferença se torna ainda mais visível considerando $\boldsymbol{p}$ **= 607** e $\boldsymbol{k}$**=2**: Valor máximo possível pelo método Binário: **2937888**. Valor máximo pela exponenciação simples:

DOI: 10.5540/03.2014.002.01.0020

5311379928167670986895882065524686273295931177270319231994441382004035598608522427391625022652292856688893294862465010153465793376527072394095199787665873519438312708353932190317281 27(183 dígitos);

Para o maior número primo teríamos mais de 17 milhões de dígitos envolvidos em cada cálculo. Por este método, considerando ***k*=2**, teríamos nó máximo um valor igual a **6701383727991842**.

Este método, porém, pode ser aprimorado, e este é o objetivo deste trabalho. Apresentar algumas alterações para tornar o mesmo ainda mais eficiente.

4. RESULTADOS

As iterações necessárias para afirmar que um número de *mersenne* é divisível por qualquer número da forma $p * k + 1$ pelo método binário apresentado, é igual à quantidade de dígitos de p transformado em número de base 2 e o tamanho máximo do seu divisor pode ser $(p * k)^2 * 2$.

Porém, esta conclusão pode eventualmente ser obtida com menos iterações com o parcelamento do expoente binário, que reduz proporcionalmente o número de iterações, dada pela seguinte equação:

$$i = \frac{d}{np}$$

Onde,
i=número de iterações;
d=dígitos do expoente binário e
np=número de parcelas.

Este parcelamento é possível quando tenho um expoente binário com parcelas iguais cujo resto desta parcela é igual a 1 quando dividido por $p * k + 1$.

Consideremos *p*=63 e *k=2*;

Teremos:

$$p = 63 = 111111$$

Podemos dividir o número binário em duas partes iguais:

$$p = \ 63 = \ 111 \ \ 111$$

Aplicando o método binário, observa-se que o resto da divisão é igual a 1 após a terceira iteração, ou seja, ao fim da primeira parcela. Deste modo pode-se concluir que $2^{63} \equiv 1 \bmod 127$.

r^2	Bits Restantes	Remover bit	Multiplicação por 2	$r = Mod.127$
1 x 1=1	111 111	~~1~~11 111	1 x 2 = 2	2
2 x 2=4	111 111	~~11~~1 111	4 x 2 = 8	8
8 x 8=64	111 111	~~111~~ 111	64 x 2 = 128	1

Tabela 3: Expoente binário por parcela

Outro fator interessante pode ser considerado para a eficiência do algoritmo buscando limitar o tamanho máximo dos valores. Este fator está diretamente relacionado com a nova entrada do algoritmo, ou seja, o resto *r* de cada iteração.

Caso *r* seja um valor maior que p, pode-se optar por utilizar o resto da subtração, conforme equação:

$$r_{ent} = (p * k + 1) - r$$

Com isto, limita-se o valor máximo do cálculo, para $p^2 * 2$, equivalendo a menos da metade proposto pelo algoritmo original.

Consideremos novamente *p*=23 e *k=2*, teremos:

$$r_{max} = 23^2 * 2 = 1058$$

r^2	Bits Restantes	Remover bit	Multiplicação por 2	$r = Mod.47$
1 x 1=1	10111	~~1~~0111	1 x 2 = 2	2
2 x 2=4	0111	~~10~~111	-	4
4 x 4=16	111	~~101~~11	16 x 2 = 32	(32>23)=47-32=15
15 x 15=225	11	~~1011~~1	225 x 2 = 450	(27>23)=47-27=20
20 x 20=400	1	~~10111~~	400 x 2 = 800	1

Tabela 4: Algoritmo com troca pelo menor resto *r*.

Neste exemplo percebeu-se duas trocas por valores menores. No entanto, existem casos em que esta troca gera resultados mais interessantes. Isto ocorre no pior caso para o algoritmo original, ou seja, quando: $r = p * k$;

Com a troca proposta,

$$r_{ent} = \big((p * k) + 1\big) - r$$

Substituindo *r;*

$$r_{ent} = \big((p * k) + 1\big) - (p * k)$$

Logo: $r = 1$;

5. CONCLUSÕES

Levando-se em consideração a importância da pesquisa em relação à teoria dos números, principalmente aos números primos e perfeitos, pequenos ajustes em algoritmos existentes se tornam muito úteis, devido à complexidade do problema.

A incorporação do parcelamento do expoente, pode ser útil quanto ao tempo de execução do algoritmo, pois cada iteração se torna muito custosa computacionalmente quando falamos em milhares de dígitos.

De igual forma, a redução do valor máximo do resto também se mostra de forma atraente, considerando que o mesmo serve de entrada para cada nova iteração. Com o algoritmo apresentado, a redução pode representar uma considerável redução em relação ao valor a ser utilizado pelo algoritmo original.

Para trabalhos futuros, pode ser abordada a relação entre os restos de cada divisão para k={2,4,6,8,...}, gerando possíveis saídas antecipadas do algoritmo.

6. REFERÊNCIAS

[1] ATKIN, A.; BERNSTEIN, D. Prime Sieves using Binary Quadratic *Forms*.-Mathematics of Computation, Providence - Rhode Island - USA, v. 73, p. 1023-1030, 2004.

[2] CLAY MATHEMATICS INSTITUTE. Riemann Hypothesis. Disponível em: <http://www.claymath.org/millennium/Riemann_Hypothesis/> Acesso em: 02 de outubro de 2013.

[3] COUTINHO, S. C. Números inteiros e criptografia RSA. Segunda edição, IMPA, 2000. 213 págs.

[4] EUCLIDES. Elementos de Geometria. Versão Latina de Frederico Commandino. 1944.

[5] GIMPS - Great Internet *Mersenne* Prime Search. Disponível em: <http://www.mersenne.org> Acesso em: 02 de outubro de 2013.

[6] IFRAH, Georges. História Universal dos Algarismos: a inteligência dos homens contada pelos números e pelo cálculo. Traduzido por Alberto Muñoz e Ana Beatriz Katinsky. Rio de Janeiro: Nova Fronteira, 1997 – 2V. Tradução de: Histoire universalle dês chiffres.

[7] MERSENNE PRIMES: History, Theorems and Lists. Disponível em: <http://primes.utm.edu/*mersenne*/index.html> Acesso em: 02 de outubro de 2013.

[8] ROBERT, Sedgewick. Algorithms in C. Addison-Wesley, 1990.

[9] SAUTOY, du Marcus. A musica dos números primos. A história de um problema não resolvido na matemática. Traduzido por Diego Alfaro. Rio de Janeiro: Jorge Zahar Ed., 2007.

[10] SINGH, Simon. O Livro dos Códigos. A Ciência do sigilo – do antigo Egito à criptografia quântica. Tradução de Jorge Calife. – Rio de Janeiro: Record, 2001.

[11] ZHANG, Sibao; MA, Xiaocheng; ZHOU, Lihang. Some Notes on the Distribution of Mersenne Primes. Applied Mathematics, 2010, 1, 312-315. doi:10.4236/am.2010.14041 Published Online October 2010 (http://www.SciRP.org/journal/am)

