APLICAÇÕES DA ARITMÉTICA MODULAR NA CRIPTOGRAFIA

Alandesson Linhares de Carvalho[1]
Daniel Vivorio Marques Rodrigues[2]
Leonardo Henrique da Rocha Araujo[3]

Ciência da Computação

cadernos de graduação
ciências exatas e tecnológicas

ISSN IMPRESSO 1980-1777
ISSN ELETRÔNICO 2316-3135

RESUMO

A aritmética modular, estudada no curso de Álgebra I, é um recurso baseado na divisibilidade, pode-se usar esse princípio para criar padrões de validação, como no CPF e muitos outros números de identificação com os quais as pessoas se deparam. Por isso esse artigo foi desenvolvido objetivando discutir a criptografia e a aritmética modular e quais são seus usos. Logo, houve a necessidade de falar da história da criptografia e os conceitos básicos de aritmética modular. Além disso, foi discutida uma forma de usar aritmética modular para criar um código que, se interceptado não tem como voltar à mensagem original. E a partir dos conhecimentos adquiridos sobre a aritmética modular foi demonstrado algumas das suas inúmeras aplicações, as quais podem ser comumente encontradas no dia-a-dia.

PALAVRAS-CHAVE

Criptografia. Teoria dos Números. Aritmética Modular. Computação.

ABSTRACT

The modular arithmetic, studied in the course of Algebra I, is a resource based on divisibility, you can use this principle to create validation standards such as the CPF and many other identification numbers with which people encounter. So this article was developed aiming to discuss encryption and modular arithmetic and what are its uses. Therefore, it was necessary to talk about the history of cryptography and the basics of modular arithmetic. Furthermore, a way to use modular arithmetic been discussed to create a code that is intercepted can not get back to the original message. And from the knowledge gained on the modular arithmetic it was shown some of its numerous applications, which can be commonly found in the day-to-day.

KEYWORDS

Encryption. The Theory of Numbers. Modular Arithmetic. Computation.

1 INTRODUÇÃO

1.1 HISTÓRIA DA CRIPTOGRAFIA

Segundo Sá (2007), a criptografia foi inventada e primeiramente aplicada pelo imperador romano Júlio Cesar para se comunicar com seus generais, de forma que qualquer indivíduo que interceptasse suas mensagens não conseguisse decifrá-las. Cesar utilizava uma regra simples para decifrar suas mensagens, ele simplesmente selecionava a palavra desejada e substituía cada letra por uma que ficava três posições antes no alfabeto. Como podemos ver, é o princípio da criptografia, aonde o transmissor codifica sua mensagem com o intuito de que somente o destinatário final entenda. Quando essa mensagem chega ao receptor ele aplicaria a operação inversa e assim decodificaria a mensagem.

Representando matematicamente a criptografia usada por Júlio Cesar: Sendo x uma letra qualquer do alfabeto e y o resultado depois da criptografia teríamos y= x - 3. Imaginando que queríamos criptografar a seguinte mensagem usando a mesma regra usada por Cesar, "Ensino Fundamental", teríamos: "BkpfklCrkaxjbkqxi".

Tabela 1 – Alfabeto codificado com chave diminuir

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

Fonte: KHAN, 2013

Esse é um exemplo como vários outros e serviu durante muitos anos para omitir mensagens, apesar de não serem difíceis de decifrar.

Quando relacionamos esse sistema com a aritmética modular temos o seguinte pensamento:

Tabela 2 – Alfabeto numerado para facilitar o uso de chaves

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

Fonte: KHAN, 2013

Para Sá (2007), designamos um número para cada letra do alfabeto e assim podemos substituir na chave. Quando o resultado for superior a 26 voltaremos ao início do alfabeto como, por exemplo, se tivermos o número 30 corresponderá à letra "D" pois 30 = 26 + 4 ou, como já conhecemos, 30 ≡ 4 (*mod* 26).

Seu principal uso foi na segunda guerra mundial, onde foi investindo em massa em sistemas eletromecânicos na codificação e decodificação das mensagens. Com ajuda de mais de sete mil ingleses trabalhando no *Government Communications Headquarters*os códigos alemães foram quebrados, com aproximadamente quatro mil sinais alemães por dia. Depois da segunda guerra mundial a área da criptografia realmente cresceu, com diversos estudos de complexos algoritmos matemáticos e com esses trabalhos formaram a base para a ciência da computação moderna, conclui Sá (2007).

Segundo Sá (2007), atualmente a criptografia não sofreu grandes mudanças na sua ideia inicial, todavia houve grandes melhoras na sua segurança e na complexidade da chave utilizada para criptografar arquivos e mensagens. Esse aperfeiçoamento aconteceu devido à internet, pois a comunicação que existe entre computadores e outros dispositivos conectados a rede precisam passar informações que não devem ser acessado por intermediários.

2 REVISÃO DE LITERATURA

2.1 ARITMÉTICA MODULAR

A aritmética modular trata de conceitos de divisibilidade e congruência que são trabalhados com conjunto dos números inteiros. O estudo da aritmética modular trabalha com módulo (mod) que segundo a comunidade *Khan Academy* é o operador que tem como objetivo conseguir o resto de uma divisão.

Um exemplo da aplicação do operador mod seria sobre a divisão 13/5 = 2 cujo resto é igual a 3, Se preciso pode ser reescrito para se obter como foco o resto da divisão, utilizando o operador mod. Se essa divisão for reescrita usando o mod, o resultado será 13 mod 5 = 3.

Ainda existe outro símbolo de grande importância para a aritmética modular que seria a congruência simbolizada como ≡. Este símbolo está diretamente associado com o operador mod. À congruência é utilizada para reescrever A - B é divisível por C de um uma forma que possa ser melhor estudada. Assim se tem que $A \equiv B \, (mod \, C)$, também se pode afirmar que A = $C.q + B$, sendo que A, B, C, q são inteiros.

Um exemplo da aplicação de congruência seria sobre a divisão $\frac{(26-11)}{5}$. Isso pode ser reescrito, usando mod e congruência. Para se trazer para uma forma mais familiar para a aritmética modular assim se torna possível mudar $\frac{(26-11)}{5}$ para $26 \equiv 11 \, (mod \, 5)$.

2.2 CRIPTOGRAFIA

É, como já mostrado na introdução, uma forma de manter escondida dados por meio de um padrão. Porém ela pode ser facilmente decodificada por computadores se for criada por humanos, então usamos as próprias máquinas para criar essas chaves. Pode ser definidas em dois grupos:

• Criptografia de chave simétrica: "Como o nome sugere, a criptografia simétrica, baseia-se na simetria das chaves, ou seja, a mesma chave usada para criptografar será usada para decifrar a mensagem" (OLIVEIRA; GUIMARÃES; LINS, 2006, p. 36). É basicamente o sistema adotado por Júlio Cesar.

• Criptografia de chave assimétrica: "A criptografia assimétrica, por sua vez, envolve o uso de duas chaves distintas, uma privada e outra pública" (OLIVEIRA; GUIMARÃES; LINS, 2006, p. 38). Esse tipo de criptografia será discutido mais a frente.

Como afirmado por Thiago Castelló e Verônica Vaz (on-line) a chave pública é livremente distribuída para qualquer dispositivo que solicitar uma troca de arquivos. Essa chave é usada para codificar o arquivo que será enviado para o computador que a forneceu. Quando o arquivo for recebido o computador vai precisar da sua chave privada, pois somente sua chave privada conseguirá decodificá-la, já que uma vez que o que o arquivo é codificado com a chave pública, somente o computador com a chave privada é capaz de decodificar esse arquivo.

Gallo e Hancock (2003) mostram o exemplo de aritmética modular na chave pública no caso do algoritmo RSA, esse algoritmo é feito da seguinte forma: primeiro escolhe-se dois números primos e serão chamados de "p" e "q" (17 e 19, por exemplo), em seguida calculamos "n" como produto de p * q = (323), agora o número a ser escolhido é "e", que será um número sem fator comum com (p - 1) * (q - 1) = (288), que poderia ser 35. Por fim deve ser calculado "d", da seguinte forma: e * d *mod* (p - 1) * (q - 1) = 1, também podendo ser escrita de outra forma, e * d ≡ 1 (*mod* (p - 1) *(q - 1)). Calculando:

$35 * d \equiv 1 \ (mod\ 288)$

$35d - 1 = 288q$

Depois de resolver essa equação diofantina, acaba em d = 107, com todas as letras devidamente calculadas:

p = 17
q = 19
n = 323
e = 35
d = 107

A partir daí as chaves são geradas:

A chave pública é (e,n) = (35, 323)

A chave privada é (d, n) = (107, 323)

Ao calcular as chaves, o próximo passo é criptografar a mensagem, caso a mensagem fosse "Oi", pela tabela ASCII "O" = 79 e "i" = 105. Agora é necessário juntar os dois números (79105) e dividi-los em "blocos" com menos dígitos do que "n" (2), e completando os que faltarem com "0", ficando 79 10 50. Por fim é necessário usar os valores da chave pública para criptografar:

$79^{107} (mod\ 323) = 260$
$10^{107} (mod\ 323) = 116$
$50^{107} (mod\ 323) = 84$

Com isso a mensagem criptografada seria 260 116 84. Já para descriptografar é o mesmo processo, só que com a chave privada:

$260^{107} (mod\ 323) = 79$
$116^{107} (mod\ 323) = 10$
$84^{107} (mod\ 323) = 50$

Com isso volta-se ao código 791050, eliminando o último 0 e separando como antes ficaria 79 105, que seriam traduzidos pela tabela como "Oi". Como se pode perceber é algo bem efetivo, pois o receptor terá a chave privada e pública, enquanto o que enviará só terá a chave pública, então se a mensagem for interceptada, será impossível de ser descriptografada, porque quem interceptar só terá a chave pública, que só serve para criptografar.

2.3 APLICAÇÕES NO DIA A DIA

2.2.1 Cadastro de Pessoa Física - C.P.F.

Segundo Sant'Anna (2013), no nosso cotidiano, temos vários casos de aritmética e acabamos não percebendo, um deles é o Cadastro de Pessoas Físicas (CPF), contendo 11 dígitos com os dois últimos como dígitos de controle, com o intuito de evitar fraudes e enganos, sendo dependente dos 9 primeiros dígitos. Para calcular os 11 dígitos, seguiremos as seguintes regras:

Supondo $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$ os primeiro 9 números do CPF e devemos multiplicá-los respectivamente por {1,2,3,4,5,6,7,8,9} e somar os resultados gerando S_1. Para encontrarmos o 10º dígito (a_{10}) devemos dividir S_1 por 11 sendo o resto da divisão o a_{10}, caso o resto seja 10 devemos utilizar o número 0. Para encontrarmos o ultimo dígito devemos multiplicar respectivamente os dígitos $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}$ por {0,1,2,3,4,5,6,7,8,9} e somar o resultado, gerando agora S2. O décimo primeiro número será o resto da divisão de S_2 por 11, caso tenha resto 10 irá se usar 0 no lugar, conclui Sant'Anna (2013). Simplificando, os dois últimos dígitos são encontrados por meio de duas congruências de modulo 11, $S1$ - $a10 \equiv 0$ (mod 11) e $S2$ - a11 ≡ 0 mod 1, tirando a exceção já citada.

Usando o CPF 054.894.927 como exemplo, teríamos como décimo dígito verificador:

$S1 = 0 * 1 * 5 * 2 * 4 * 3 * 8 * 4 * 9 * 5 * 4 * 6 * 9 * 7 * 2 * 8 * 7 * 9$
$S1 = 0 * 10 * 12 * 32 * 45 * 24 * 63 * 16 * 63 = 265$
Então temos: $265 - a10 \equiv 0$ (mod 11) $\rightarrow a10 = 1$
$S2 = 0 * 0 * 5 * 1 * 4 * 2 * 8 * 3 * 9 * 4 * 4 * 5 * 9 * 6 * 2 * 7 * 7 * 8 * 1 * 9$
$S2 = 0 * 5 * 8 * 24 * 36 * 20 * 54 * 14 * 56 * 9 = 226$
Então temos: $266 - a11 \equiv 0$ (mod 11) $\rightarrow = 6$

Notamos que o CPF completo é 054.894.927-16 então caso os dois últimos fossem diferentes de 16 saberíamos que o CPF era falso.

2.2.2 Cartão de Crédito

Figura 1 – Cartão do banco (Adaptado pelos Autores)

Fonte: Próprio autor.

Ao criar um cartão de crédito, a instituição se preocupa com alguns detalhes, os cartões costumam ter 16 dígitos, sendo o 1º: 1 e 2 para linhas aéreas, 3 para viagens e entretenimento, 4 e 5 para bancos e assim por diante. O segundo dígito é como o primeiro, serve para restringir um pouco mais, em questão de instituição: Visa (4xxxxx), Master (51xxxx - 55xxxx). Depois, do 7º digito até o 15º são o número de identificação do cliente, e por fim o último dígito que é calculado por aritmética modular, da seguinte forma:

Uma soma na qual os números de posição ímpar (1º, 3º, 5º) e somados com os de posição par sem multiplicar, a soma desses números tem que ser divisível por 10 (mod 10).

Por exemplo: Descobrir qual o último dígito do cartão 4417 1234 5678 911X para que ele seja válido.

(4 * 2) + 4 + (1 * 2) + 7 + (1 * 2) + 2 + (3 * 2) + 4 + (5 * 2) + 6 + (7 * 2) + 8 + (9 * 2) + 1 + (1 * 2) + x é divisível por 10.

Todos os números que são duplos (14) têm os dígitos somados também, ficando: 67 + x é divisível por 10.

Pelos conceitos já vistos antes, podemos escrever como:
$67 \equiv -X \ (mod\ 10)$

Logo X é quanto falta para 67 chegar a um número divisível por 10 (nesse caso, 70), o número escolhido seria 3:

$67 \equiv -3 \ (mod\ 10)$ seria verdade, pois:
$67 + 3 = 10q$
$70 = 10q$

2.2.3 Código de Barras

Para Esquica (2013), a principal função do código de barras é a praticidade, apenas com uma leitura por meio de uma máquina se obtém a captação de dados, uma transação mais rápida e é possível atualização dos dados em tempo real, gerando um maior controle, praticidade, manipulação remota e ainda a diminuição de custos.

De acordo com Esquica (2013), o código de barra é usado universalmente e serve para fazer identificações em diversas áreas como na indústria, comércio, bancos, bibliotecas entre outras áreas. Os primeiros a realizar estudos sobre códigos para melhorar o processo de comercialização foram Joseph Woodland e Bernard Silver em 1952, criando circunferências concêntricas de espessura variável.

Segundo Esquica (2013), na década de 1970 George J. Laurer elaborou o sistema de código de barras usado até hoje, barras verticais alternadas entre pretas e brancas e com números embaixo. Esse código possuía 12 números abaixo das listras e foi aceito em 1973, recebendo o nome de *Universal Product Code* (UPC), sendo Estados Unidos e Canadá os primeiros países a adotarem o UPC.

Figura 2 – Código de Barra modelo UPC

Fonte: ESQUINCA, 2013

Como escrito por Esquica (2013), em 1976 esse código foi melhorado, acrescentando mais um 1 número na sequência para se identificar o país de origem do produto. Ao receber o 13º número esse sistema foi renomeado para *European Article Numbering system* ou EAN-13.

Figura 3 – Código de barra modelo EAN-13

Fonte: ESQUINCA, 2013

O código de barras é uma representação de números em formato de listras, produzindo uma leitura óptica para interpretar qual o código representa. São formadas por sequências de barras verticais alternadas entre preto e branco e com a largura variando entre fina, média, grossa e muito grossa, identificando os seus números respectivos. A Tabela 3 abaixo mostra a interpretação de cada barra, conclui Esquinca (2013).

Tabela 3 – Classificação das listras do código de barras

Listras	Fina	Média	Grossa	Muito Grossa
Branca	0	00	000	0000
Preta	1	11	111	1111

Fonte: Retirado de Aritmética: Códigos de barras e outras aplicações de congruências.

Os códigos EAN-13 possuem 3 blocos de barras um pouco maiores que as outras, cada bloco contendo 3 barras para servirem de delimitadores e não são lidos pelo leitor óptico.

Já o código UPC utiliza o mesmo princípio do EAN-13, sendo a única diferença que o primeiro e último número estão decodificados com barras do mesmo comprimento dos delimitadores.

Segundo Esquica (2013), para se ler um código de barra UPC terá de ser feita uma leitura sobre a espessura e cor da barra com auxílio da tabela acima e a cada 4 barras terá associado uma sequência de 7 dígitos binários. Cada dígito de 0 a 9 possui uma representação de zeros e uns. O número será representado por uma determinada sequência lida pelo leitor óptico, respeitando sua posição, sendo a direito ou esquerda do número.

Tabela 4 – Número referente a cada leitura

Dígito	Lado Esquerdo	Lado Direito
0	0001101	1110010
1	0011001	1100110
2	0010011	1101100
3	0111101	1000010
4	0100011	1011100
5	0110001	1001110
6	0101111	1010000
7	0111011	1000100
8	0110111	1001000
9	0001011	1110100

Fonte: Retirado de Aritmética: Códigos de barras e outras aplicações de congruências.

Como por exemplo, o número 036000 - 291452 será escrito como: 0001101 - 0111101 - 0101111 - 0001101 - 0001101 - 0001101 - 1101100 - 1110100 - 1110010 - 1011100 - 1001110 - 1101100 e representado ilustradamente como:

Figura 4 – Exemplo de código de barras UPC

Fonte: ESQUINCA, 2013

Para se ler um código de barra EAN-13 deve-se fazer a seguinte interpretação: separamos os 3 primeiros números, eles servem para identificar o país de origem, no caso do Brasil é usado 789 para designar produtos fabricados no Brasil. Os próximos 4 a 5 números se referem ao código da empresa, os 5 números seguintes o código do produto e o último número é o dígito verificador, de acordo com Esquica (2013).

Figura 5 – Explicação do código de barras EAN-13

Fonte: ESQUINCA, 2013

Segundo Esquica (2013), com a existência de dois sistema de código de barra (UPC e EAN-13) teoricamente as lojas deveriam ter 2 leitores ópticos diferentes, mas por um motivo de praticidade foi criado um sistema de leitura que possa ler tanto UPC quanto EAN-13. Para poder ler os dois sistemas em uma mesma máquina o primeiro dígito no sistema EAN-13 é determinado pelos próximos 6 dígitos e para isso ocorrer foi acrescentada mais uma representação para cada dígito esquerdo.

Tabela 4 – Leitura óptica para UPC e EAN-13

Dígito	Lado Esquerdo (Ímpar)	Lado Esquerdo (Par)	Lado Direito
0	0001101	0100111	1110010
1	0011001	0110011	1100110
2	0010011	0011011	1101100
3	0111101	0100001	1000010
4	0100011	0011101	1011100
5	0110001	0111001	1001110
6	0101111	0000101	1010000
7	0111011	0010001	1000100
8	0110111	0001001	1001000
9	0001011	0010111	1110100

Fonte: Retirado de Aritmética: Códigos de barras e outras aplicações de congruências

Como dito por Esquica (2013), se inicia a leitura do código de barras da esquerda para direita, usaremos a tabela 3 para ler o código, se a sequência de 7 números encontrada tenha quantidade ímpar de uns então procura-se na tabela 4 o algarismo correspondente na coluna "Lado Esquerdo (Ímpar)" e caso contrário se procura na coluna "Lado Esquerdo (Par)". O resto da leitura feita no lado direito é igual à da forma UPC.

Se usarmos a figura 5 como exemplo, teremos o número 4-891668-326689.

Tabela 5 – Exemplo de leitura EAN-13

Lado Esquerdo		Lado Direito	
1º	8↔0110111 (quant. ímpar de uns)	7º	3 ↔ 1000010
2º	9↔0010111 (quant. par de uns)	8º	2 ↔ 1101100
3º	1↔0011001 (quant. ímpar de uns)	9º	6 ↔ 1010000
4º	6↔0101111 (quant. ímpar de uns)	10º	6 ↔ 1010000
5º	6↔0000101 (quant. par de uns)	11º	8 ↔ 1001000
6º	8↔0001001 (quant. par de uns)	12º	9 ↔ 1110100

Fonte: Retirado de Aritmética: Códigos de barras e outras aplicações de congruências.

Para Esquica (2013), a partir dos 6 primeiros números verificados vemos que eles são classificado em quantidade ímpar ou par de números uns, e com essa classificação geramos o 1º número.

Tabela 6 – Ordem de codificação EAN-13

Dígito Inicial	1º	2º	3º	4º	5º	6º
0	Ímpar	Ímpar	Ímpar	Ímpar	Ímpar	Ímpar
1	Ímpar	Ímpar	Par	Ímpar	Par	Par
2	Ímpar	Ímpar	Par	Par	Ímpar	Par
3	Ímpar	Ímpar	Par	Par	Par	Ímpar
4	Ímpar	Par	Ímpar	Ímpar	Par	Par
5	Ímpar	Par	Par	Ímpar	Ímpar	Par
6	Ímpar	Par	Par	Par	Ímpar	Ímpar
7	Ímpar	Par	Ímpar	Par	Ímpar	Par
8	Ímpar	Par	Ímpar	Par	Par	Ímpar
9	Ímpar	Par	Par	Ímpar	Par	Ímpar

Fonte: Retirado de Aritmética: Códigos de barras e outras aplicações de congruências.

No exemplo anterior chegamos a sequência: ímpar, par, ímpar, ímpar, par, par. Usando a tabela 6 chegaremos ao número 4, então o código de barras está correto.

Para Esquica (2013), quando existe algum erro na leitura das barras o operador terá que digitar a sequência de números e podem ocorrer erros. O código de verificação é um recurso para detectar alguns desses erros.

Segundo Esquica (2013), consideremos a1 até a13 a sequência de dígitos de um código de barras qualquer de modelo EAN-13 o último dígito é chamado de dígito de verificação, no modelo UPC também. No sistema EAN-13 o dígito de verificação é encontrado a partir dos 12 primeiros números e no UPC são usados os 11 primeiros números.

Considerando o sistema EAN-13 nomearemos o 13º número de X e escrevendo em forma de vetor teremos: α = (a1, a2,..., a12, X) e no sistema EAN-13 teremos o vetor fixo ω = (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1). Se calcularmos o produto escalar de $\alpha . \omega$ teremos o dígito de verificação $X \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, é tal que $\alpha . \omega$ mod 10.

Finalizado por Esquica (2013), se formos calcular o dígito de verificação no sistema UPC terá somente a pequena modificação no vetor por se usar somente 12 números, ω = (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1).

3 CONSIDERAÇÕES FINAIS

Então, pode-se concluir, com uma análise da pesquisa realizada, que existem muitas aplicações em que a aritmética modular age de forma essencial, como mostrado durante a pesquisa. Além disso, muitos desses usos podem ser encontrados durante o percorrer do dia e o escopo dessas aplicações podem variar, tanto para identificação ou até para ajudar no processo de criptografia, ou seja, melhorar a proteção de informações.

Como já visto em um dos tópicos, o processo de aritmética modular é algo perfeito para a criptografia, pois é um processo que gera um número que não pode ser revertido, por exemplo, o método de Júlio César era uma fórmula y = x - 3 e era facilmente revertido para x = y + 3, mas com módulo isso já é mais complexo, o valor de y pode ser conhecido, mas nem por isso implica dizer que o valor de x será por consequência.

REFERÊNCIAS

CASTELLÓ, Thiago; VAZ, Verônica. **Assinatura digital.** 24/05/2007. 10p. Disponível em: <http://www.gta.ufrj.br/grad/07_1/ass-dig/Introduo.html>. Acesso em: 14 out. 2013.

ESQUINCA, Josiane Colombo Pedrini. **Aritmética:** Códigos de barras e outras aplicações de congruências. Disponível em: <http://bit.profmat-sbm.org.br/xmlui/bitstream/handle/123456789/371/2011_00238_JOSIANE_COLOMBO_PEDRINI_ESQUINCA.pdf?sequence=1>. Acesso em: 14 nov. 2013.

GALLO, Michael A.; HANCOCK, William M. **Comunicação entre computadores e tecnologias de rede.** São Paulo: Thomson, 2003. 673p.

KHAN, Salman. Khan Academy. **Cryptography**; aproximadamente 3 telas. Disponível em:<https://www.khanacademy.org/math/applied-math/cryptography/modar ithmetic/a/what-is-modular-arithmetic>. Acesso em: 14 out. 2013.

OLIVEIRA, Raimundo Corrêa de; GUIMARÃES, Alexandre Guedes; LINS, Rafael Dueire. **Segurança em Redes Privadas Virtuais - Vpns.** São Paulo: Brasport, 2006. 232p.

SANT'ANNA, Iury Kersnowsky de. **A aritmética modular como ferramenta para as séries finais do ensino fundamental.** Disponível em: <http://bit.profmat-sbm.org.br/xmlui/bitstream/handle/123456789/285/2011_00137_IURY_KERSNOWSKY_DE_SANTANNA.pdf?sequence=1>. Acesso em: 14 nov. 2013.

Data do recebimento: 11 de março de 2015
Data da avaliação: 16 de maio de 2015
Data de aceite: 18 de maio de 2015

1. Curso de Ciência da Computação da Universidade Tiradentes. E-mail: alandessonlc@hotmail.com
2. Curso de Ciência da Computação da Universidade Tiradentes. E-mail: danielvivoriomr@hotmail.com
3. Curso de Ciência da Computação da Universidade Tiradentes. E-mail: leonardohra@gmail.com