Matemática Discreta

Projeto de criptografia RSA

CURSO DE CIÊNCIA DA COMPUTAÇÃO - UFAL

Alunos: Adeilton dos Santos Mesquita, Carlos Henrique Roque Ferreira, Matheus Macário, Maria Vitória Alves Spinello, Mateus dos Santos Ferreira

CONCEITOS MATEMÁTICOS UTILIZADOS

O Algorítmo RSA tem como principal produto duas chaves; a chave pública e a chave privada.

A chave pública é gerada da seguinte forma:

Primeiro escolhe-se dois números primos P e Q, com a posse desses dois números temos N, que é o produto de P por Q.

N = P * Q

Calculamos, então, a função totiente em n: ϕ(n) = (p - 1)(q - 1). (Lê-se "phi de n")

A Função totiente de Euler ou Função Phi irá determinar a quantidade de números naturais menores que N que são co-primos a N, ou seja, sem divisores comuns entrei si com exceção do 1.

$$\varphi(x) = \#\{n \in \mathbb{N} | n \leq x \wedge \mathrm{mdc}(n, x) = 1\}$$

Para todo número primo x a função phi desse número é simplesmente x-1, pois pela definição dos números primos temos que todo primo só pode ser divisível por 1 e por ele mesmo.

A função phi também é multiplicativa, ou seja.

Sendo N = P * Q, o phi de N pode ser expresso por:

N = P * Q

ϕ(N)= ϕ(P) * ϕ(Q)

ϕ(N)= ϕ(P-1) * ϕ(Q-1)

Após isso definimos o número E, o mdc entre E e ϕ(N) tem que ser igual a 1.

mdc(ϕ(N), E) == 1, se não satisfazer a isso, terá que ser escolhido outro número E.

A chave pública é composta por E e N

A chave privada é gerada da seguinte forma:

A chave privada será D, para encontrá-la é preciso satisfazer a igualdade:

mod(D^E, ϕ(N)) == 1.

A chave privada é composta por D e N

Por que o Algorítmo RSA é tão seguro?

O método RSA é seguro pois a chave D é composta por um processo que torna impossível regredir por meio de cálculos ao ponto de se determinar com tempo hábil os números primos P e Q utilizados na formulação das chaves pública e privadas.

Lembrando: Dois números primos P e Q dão origem ao N, que possibilita o cálculo de ϕ(N). Como esse algoritmo utiliza números de 2048 bits, ou seja, um número com 617 dígitos decimais, temos 2^{2048} possibilidades. Para efeito de comparação estimasse que a quantidade de átomos no universo observável está na casa de 10^{80}. Multiplicar dois números primos de 617 dígitos é uma tarefa fácil para nossos computadores atuais, porem fatorar um número N de 4096 bits (2048*2) é inviável pra tecnologia atual. Levaria-se milhares de anos para achar os dois números primos que têm como produto esse tal N, logo, podemos considerar o algorítmo RSA seguro para troca de informações.

FUNCIONAMENTO DO CÓDIGO

```
void criptografar() {
    int *mensagem, *c, n, e;
    int i, j, k, l, m, aux;
    mensagem[100];
    char ch, m;

    printf("\n\n\nCriptografar");
    printf("\n\n\n");
    printf("Digite sua mensagem:");
    gets(mensagem);
    printf("\n");
    printf("Digite o chave publica (n e e):\n\n");
    scanf("%d %d", &n, &e);
    printf("\n");
    printf("Digite a chave publica (n e e):\n\n");
    printf("%d %d\n", n, e);
    printf("\n");
    c[100];
    ch[100], m[100], aux[100];
    int mensagem[100], mensagem2[100];
    int *n = mensagem, *m = mensagem2;

    FILE *w = fopen("mensagem_criptografada.txt", "w");

    for(int i = 0; i < n; i++) {
        scanf_s(w, "%d", &mensagem[i]);
        printf("%d ", mensagem[i]);
        if (mensagem[i] == 0) {
            mensagem[i] = 26;
        }
        m[i] = mensagem[i];
    }
    fclose(w);
    fclose(f);

    printf("Arquivo criptografado e salvo no arquivo 'mensagem_criptografada.txt'\n\n");

    return;
}
```

```
void descriptografar() {
    int *mensagem, *c, n, e;
    int i, j, k, l, m, aux;
    mensagem[100];
    char ch, m;

    printf("\n\n\n Descriptografar");
    printf("\n\n\n");
    printf("Digite o chave publica (n e e):\n\n");
    scanf("%d %d\n", &n, &e);
    printf("\n");
    printf("Digite o chave publica (n e e):\n\n");
    printf("%d %d\n", n, e);
    printf("\n");
    c[100];
    ch[100], m[100], aux[100];
    int mensagem[100], mensagem2[100];
    int *n = mensagem, *m = mensagem2;

    FILE *w = fopen("mensagem_criptografada.txt", "r");

    for(int i = 0; i < n; i++) {
        scanf_s(w, "%d", &mensagem[i]);
        printf("%d ", mensagem[i]);
        if (mensagem[i] == 26) {
            mensagem[i] = 0;
        }
        m[i] = mensagem[i];
    }
    fclose(w);
    fclose(f);

    printf("Arquivo criptografado e salvo no arquivo 'mensagem_criptografada.txt'\n\n");

    return;
}
```

```
    fclose(mensagem[i]);
    fclose(f);
    printf("Mensagem descriptografada salvo\n", mensagem[i]);
    printf("Mensagem descriptografada e salva no arquivo 'mensagem_descriptografada.txt'\n\n");
    system("pause");
    else printf("ERRO: MDC(E^E * ϕ(N), ϕ(N)) == 1\n\n");
else printf("ERRO: P e Q não são coprimos\n\n");

return;
}
```

Funções Utilizadas

primo() - Determina se um número é ou não primo.

euclides() - Calcula o máximo divisor comum entre dois números dados.

buscar_st() e **buscar_st2()** - Calcula s e t da congruência.

congruencias() - Verifica a condição da congruencia em D onde mod(D^E , ϕ(N)) == 1.

letra_numero() - Converte as letras para a cifra ((a=2),(b=3),(c=4)....(z=27),(espaço=28))

potencia() - Calcula de potência com "llint long long int" no retorno.

criptografar() - Criptografa uma mensagem com base nas chaves N e E fornecidas e salva o código no arquivo "mensagem_criptografada.txt"

Funções Utilizadas

ler_mensagem() - Extrai o código criptografado do arquivo "'mensagem_criptografada.txt" e armazena num array llint long long int char mensagem[][].

numero_letra() - Converte a cifra ((a=2),(b=3),(c=4)....(z=27),(espaço=28)) para o padrão ASCII (alfabeto de A-Z).

descriptografar() - Solicita P, Q e E, gera as chaves D, N e descriptografa uma mensagem com salvando-a no arquivo "mensagem_descriptografada.txt".

menu()- Chama todas as funções principais (gerar_chave, criptografar, descriptografar, encerrar).

Imagens de execução do código

```
.......................
############ MENU ############
.......................
|1| Gerar chaves
|2| Criptografar
|3| Descriptografar
|4| Encerrar
.......................

Opcão: 1

GERAR CHAVES
------------
Digite os números primos (p) e (q) e o expoente (e):
7
11
13
```

```
Chaves criadas e salva nos arquivos 'chave_publica.txt' e 'chave_privada.txt'.

Pressione qualquer tecla para continuar. . .
```

```
.......................
############ MENU ############
.......................
|1| Gerar chaves
|2| Criptografar
|3| Descriptografar
|4| Encerrar
.......................

Opcão: 2

CRIPTOGRAFAR
------------
Digite sua mensagem:
ola
Digite a chave publica (n) e (e):
77
13

Mensagem criptografada e salva no arquivo 'mensagem_criptografada.txt'.

Pressione qualquer tecla para continuar. . .
```

```
.......................
############ MENU ############
.......................
|1| Gerar chaves
|2| Criptografar
|3| Descriptografar
|4| Encerrar
.......................

Opcão: 3

Digite os números primos (p) e (q) e o expoente (e):
7
11
13

Mensagem descriptografada:
Ola!

Mensagem descriptografada e salva no arquivo 'mensagem_descriptografada.txt'.

Pressione qualquer tecla para continuar. . .
```


REFERÊNCIAS:

https://sites.google.com/ic.ufal.br/matematica-discreta/documentos?authuser=0

https://github.com/tarcisio-marinho/RSA

https://www.ime.unicamp.br/~ftorres/ENSINO/MONOGRAFIAS/Fernando_TN17M2.pdf

https://medium.com/@tarcisiome/algoritmo-de-criptografia-assim%C3%A9trica-rsa-ec0254a3c704a2