UNIP - UNIVERSIDADE PAULISTA

DEPARTAMENTO DE CIÊNCIAS EXATAS E TECNOLOGIA – ICET

CIÊNCIA DA COMPUTAÇÃO

APS – ATIVIDADES PRÁTICAS SUPERVISIONADAS

**AS TÉCNICAS CRIPTOGRÁFICAS:**

**CONCEITOS, USOS E APLICAÇÕES**

**São José do Rio Preto**

**2016**

**MATEUS FREITAS MACEDO – B73GGE-8**

**RICARDO TOKUNAGA KAI – N908HF-1**

**RUBENS MENDONÇA MACHADO – C89DFE-9**

**AS TÉCNICAS CRIPTOGRÁFICAS:**

**CONCEITOS, USOS E APLICAÇÕES**

Trabalho apresentado ao Curso de Ciências da Computação, da Universidade Paulista (UNIP) de São José do Rio Preto, para aprovação na APS - Atividades Práticas Supervisionadas.

Orientador: Prof. Geraldo F. D. Zafalon

**São José do Rio Preto**

**2016**

**SUMÁRIO**

|  |  |
| --- | --- |
| 1 Objetivo do Trabalho....................................................................................... | 03 |
| 2 Introdução........................................................................................................  3 Criptografia...................................................................................................... | 04  06 |
| 4 Técnicas criptográficas mais utilizadas........................................................... | 09 |
| 5 Projeto............................................................................................................  5.1 Linhas de código.........................................................................................  7 Referências ...................................................................................................  8 Anexos........................................................................................................... | 13  16  19  21 |
| 9 Anexo fichas.................................................................................................. | 22 |

**1 OBJETIVO DO TRABALHO**

Pesquisar e avaliar de forma clara o conceito de criptografia, sua utilização e suas aplicações. Para que um leigo possa entender e se familiarizar com o tema.

Desenvolver um software na linguagem C. Tentando atingir o objetivo principal, criptografar uma palavra e descriptografalo

  Pesquisar e entender o conceito, utilização e aplicação das técnicas criptográficas. Avaliar as principais técnicas usadas e selecionar uma para a elaboração da dissertação. Buscar os benefícios que a técnica selecionada proporciona. Evidenciar as falhas e as vulnerabilidades da técnica trabalhada. Divulgar aplicações que fazem ou fizeram uso da técnica. Implementar melhorias na técnica. Desenvolver um software em cima da técnica selecionada.

Apresentar o funcionamento do software e suas instruções.

**2 INTRODUÇÃO**

Tanenbaum e Wetherall (2011, p. 481) explicam que a palavra criptografia vem de origem grega que significam ‘escrita secreta’.

Carissimi, Rochol e Granville (2009, p. 346) definem criptografia como um conceito fundamental empregado em segurança de redes de computadores para prover confidencialidade, autenticidade e integridade das informações. Na realidade, a criptografia por si só é uma ciência bastante complexa, fundamenta-se em conhecimentos de matemática, e que tem sido utilizada pela humanidade há muitos séculos.

A segurança é um assunto abrangente e inclui inúmeros tipos de problemas. Em sua forma mais simples, preocupa-se em impedir que pessoas mal-intencionadas leiam ou modifiquem secretamente mensagens enviadas a outros destinatários. Outra preocupação da segurança são as pessoas que tentam ter acesso a serviços remotos que não estão autorizadas a usar. Ela também lida com meios para saber se uma mensagem supostamente verdadeira é um trote. A segurança trata de situações em que mensagens legítimas são capturadas e reproduzidas, além de lidar com pessoas que tentam negar o fato de ter enviados certas mensagens.

Partindo-se da hipótese, que assimilando o tempo e o conteúdo de estudo com as informações obtidas através de pesquisas é possível, de forma estratégica, concretizar as técnicas criptográficas mais conhecidas, buscar falhas e melhorias, portanto, são relevantes as práticas em si.

Com pesquisas, avaliações e benefícios, intenciona evidenciar falhas e vulnerabilidades que as técnicas apresentam, contribuindo na implementação de melhorias em uma técnica específica e destacando suas vantagens em relação às outras técnicas que serão apresentadas, resultando tudo isso em um software de criptografia desenvolvido na linguagem C.

A necessidade de proteger os canais de comunicação entre pessoas de uma mesma comunidade vem desde os primórdios da civilização. Essa preocupação vai muito além da ideia prioritária de só visar proteção aos meios de comunicação, mas também, resguardo do próprio conteúdo da mensagem. Este é um dos motivos que levaram à criação das técnicas criptográficas.

O trabalho intenciona esclarecer conceitos, funções e aplicações das técnicas criptográficas aos leigos. Para tanto, apresentando as técnicas mais usadas e conhecidas e selecionando uma delas, resultando na criação de um software, na linguagem C, que servirá como demonstração.

O trabalho também apresenta citações e fundamentado em obras publicadas por estudiosos especialistas, como Sousa (2002), Stallings (2005, 2007), Carissimi, Rochol e Granville (2009), Tenenbaum e Wetherall (2011), Peixinho, Fonseca e Lima (2013)

**3 CRIPTOGRAFIA**

A criptografia é quase tão antiga quanto à escrita, pois, desde que as pessoas começaram a escrever, passaram a sentir a necessidade de esconder ou de não permitir que qualquer um entendesse o que foi escrito.

Nas organizações militares as mensagens eram criptografadas e entregues habitualmente a auxiliares mal remunerados, que se encarregavam de criptografá-la e transmiti-las. Uma das restrições era criptografar e transmitir a mensagem no campo de batalha e com pouco equipamento. Outra restrição era a dificuldade de alternar os métodos criptográficos rapidamente, pois isso exigia a repetição do treinamento de um grande número de pessoas. Com o advento dos computadores isso tudo mudou e impulsiono a criptografia.

Os profissionais diferenciam cifras e códigos. Tanenbaum e Wetherall (2011, p. 481) descrevem cifra como uma transformação de caractere por caractere ou de bit por bit, sem levar em conta a estrutura linguística da mensagem e código substitui uma palavra por outra palavra ou símbolo. Os códigos não são mais utilizados, embora tenham uma história gloriosa.

A cifra opera na sintaxe (símbolos) da mensagem, enquanto um código geralmente opera na semântica (significado). Um código é armazenado como um mapeamento num livro de códigos, enquanto cifras transformam símbolos individuais conforme um algoritmo.

Segunda Guerra Mundial no Pacífico. Ela simplesmente tinha índios navajos como transmissor e receptor, que usavam palavras em Navajo específicas para termos militares, como chay-dagahi-nail-tsaidi (literalmente, matador de cágado) para indicar uma arma antitanque. A linguagem navajo é altamente tonal, extremamente complexa, e não tem nenhuma forma escrita. Além disso, nem uma única pessoa no Japão conhecia alguma coisa sobre ela. Dois princípios fundamentais.

Tanenbaum e Wetherall (2011, p. 487) descrevem os dois princípios fundamentais da criptografia e alertam que podemos correr o risco de violá-los.

Vale ressaltar que existem dois tipos de intrusos, ativo e passivo. Intruso ativo, aquele que é capaz de interferi o protocolo, atrasando, eliminando ou alterando a mensagem.

Intruso passivo, aquele que não interfere o protocolo, mas tem acesso às mensagens trocadas.

Redundância: O primeiro princípio é que todas as mensagens criptografadas devem conter alguma redundância, ou seja, informações que não são necessárias para a compreensão da mensagem.

Atualidade: O segundo princípio criptográfico é tomar algumas medidas para assegurar que cada mensagem recebida possa ser confirmada como uma mensagem atual, isto é, enviada muito recentemente. Essa medida é necessária para impedir que intrusos ativos reutilizem mensagens antigas.

Esteganografia: A origem de esteganografia remonta a Grécia antiga e significa “escrita coberta”. Sendo um dos métodos estudados pela criptologia, que basicamente funciona como uma maneira de camuflar a mensagem a ser passada, e não a torna totalmente ininteligível como a criptografia costuma fazer.

Tâmega (2003, p. 25) explica que esteganografia esconde as mensagens por artifícios, como, imagens ou um texto que tenha sentido, mas que sirva apenas para disfarçar o real conteúdo. Funciona da mesma forma que o barn code: mesclar a mensagem numa outra, em que apenas determinadas palavras devem ser lidas para descobrir o texto camuflado. Ao contrário da criptografia, que procura esconder a informação da mensagem, a esteganografia procura esconder a mensagem.

  Logica de programação conhecida como algoritmo, é uma serie de estruturas e instruções que dizem ao computador o que ele deve fazer. No desenvolvimentos de sistemas essas linguagens todas elas são baseadas em logica de programação ou algoritmos.

A linguagem de programação é como uma língua normal, um grupo de palavras com significados. No caso da programação, a maioria das linguagens é escrita em Inglês.

Estas linguagens fazem o computador assimilar cada comando e função de um algoritmo, depois executar cada função.

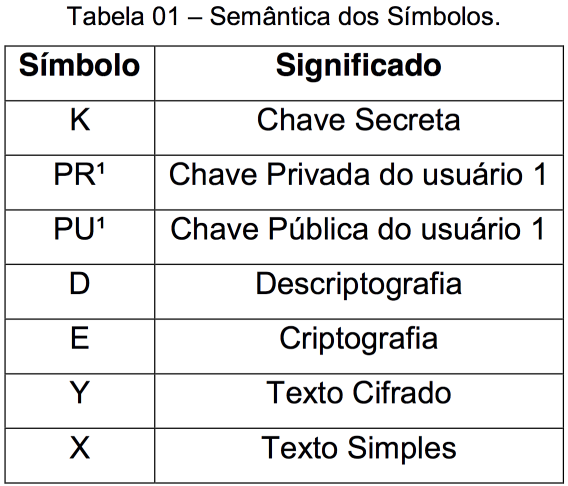
A linguagem de programação é somente como se escreve o algoritmo. O grande problema para muitos é o que “dizer” para o computador fazer o que é desejado.

Na hora de programar alguns passos são indispensáveis, como Declarar Variáveis. Variáveis podem ser escritas por letras ou números, que representam um valor que pode ser mudado a qualquer momento.

**4 DISSERTAÇÃO TÉCNICAS CRIPTOGRÁFICAS MAIS UTILIZADAS**

Uma das cifras mais antigas é a cifra de César, atribuída a Júlio César que foi a nossa escolha para criar o software em C

Os métodos de criptografia são classificados em duas categorias: chave simétrica e chave assimétrica. Para coopreender melhor é preciso entender a semântica dos símbolos.



Chave simétrica ou criptografia simétrica é uma forma de criptossistema em que a criptografia e a descriptografia são realizadas usando a mesma chave. Ela é conhecida também como criptografia convencional.

A criptografia simétrica transforma o texto claro em texto cifrado, usando uma chave secreta e um algoritmo de criptografia. Usando a mesma chave e um algoritmo de descriptografia, o texto claro é recuperado a partir do texto cifrado. Ela usa cifra de substituição ou de transposição.

Cifras de Substituição, cada letra ou grupo de letras é substituído por outra letra ou grupo de letras, de modo a criar um “disfarce”, mas mantendo a ordem dos símbolos.

A vantagem da criptografia de chave simétrica é sua simplicidade, o que reduz sua complexidade computacional para as operações de cifragem e decifração, embora ela apresente duas desvantagens importantes.

A primeira é que o remetente e o destinatário devem conhecer a chave secreta, o que implica um problema: como divulgar o valor da chave entre os interessados sem comprometer o seu sigilo. [...]. A segunda desvantagem é praticamente uma consequência da primeira: a maneira de gerenciar e compartilhar as chaves para cada par de usuários garantindo seu segredo dos demais, considerando-se ainda, potencialmente, uma quantidade importante de chaves. [...]. (CARISSIMI, ROCHOL e GRANVILLE, 2009, p. 391)

Chave Assimétrica Carissimi, Rochol e Granville (2009, p. 348) descrevem que as principais características da chave assimétrica é empregar para a cifragem e a decifração duas chaves distintas: uma chave pública e uma chave privada (secreta).

Chave assimétrica é também, segundo Peixinho, Fonseca e Lima (2013, p. 125), conhecida como criptografia de chave pública.

O algoritmo de chave pública normalmente empregado é o de 1997, definido pelos pesquisadores Ron Rivest, Adi Shamir e Leonard Adleman, mais conhecido como RSA.

Essa característica resulta em duas vantagens em relação à criptografia simétrica. A primeira vantagem é a redução da quantidade de chaves necessárias para n partes comunicarem entre si, já que cada parte utiliza seu par de chaves independentemente de quem é seu correspondente.

A segunda vantagem é em relação à divulgação e o conhecimento das chaves entre as partes envolvidas. Por definição, a chave privada não é divulgada nem conhecida pelos demais; já a chave pública, justamente por se pública, é livremente distribuída. [...] (CARISSIMI, ROCHOL e GRANVILLE, 2009, p. 349).

Existem diversas formas de representação de algoritmos, mas não há um consenso com relação à melhor delas. O critério usado para classificar hierarquicamente estas formas está diretamente ligado ao nível de detalhe ou, inversamente, ao grau de abstração oferecido.

Algumas formas de representação de algoritmos tratam os problemas apenas em nível lógico, abstraindo-se de detalhes de implementação muitas vezes relacionados com alguma linguagem de programação específica.

Por outro lado existem formas de representação de algoritmos que possuem uma maior riqueza de detalhes e muitas vezes acabam por obscurecer as ideias principais do algoritmo, dificultando seu entendimento.

Dentre as formas de representação de algoritmos mais conhecidas podemos citar:

-Descrição Narrativa;

-Fluxograma Convencional;

-Pseudocódigo, também conhecido como Linguagem Estruturada ou Portugol.

Descrição Narrativa

Nesta forma de representação os algoritmos são expressos diretamente em linguagem natural. Como exemplo, têm-se os algoritmos seguintes:

Exemplo:

Troca de um pneu furado:

Afrouxar ligeiramente as porcas ,

Suspender o carro,

Retirar as porcas e o pneu,

Colocar o pneu reserva,

Apertar as porcas ,

Abaixar o carro,

Dar o aperto final nas porcas.

Esta representação é pouco usada na prática porque o uso da linguagem natural muitas vezes dá oportunidade a más interpretações, ambiguidades e imprecisões.

Fluxograma. É uma representação gráfica de algoritmos onde formas geométricas diferentes implicam ações (instruções, comandos) distintos. Tal propriedade facilita o entendimento das ideias contidas nos algoritmos e justifica sua popularidade.

Esta forma é aproximadamente intermediária à descrição narrativa e ao pseudocódigo (subitem seguinte), pois é menos imprecisa que a primeira e, no entanto, não se preocupa com detalhes de implementação do programa, como o tipo das variáveis usadas.

Nota-se que os fluxogramas convencionais preocupam-se com detalhes de nível físico da implementação do algoritmo. Por exemplo, figuras geométricas diferentes são adotadas para representar operações de saída de dados realizadas em dispositivos distintos, como uma fita magnética ou um monitor de vídeo.

De modo geral, um fluxograma se resume a um único símbolo inicial por onde a execução do algoritmo começa, e um ou mais símbolos finais, que são pontos onde a execução do algoritmo se encerra. Partindo do símbolo inicial, há sempre um único caminho orientado a ser seguido, representando a existência de uma única sequência de execução das instruções. Isto pode ser melhor visualizado pelo fato de que, apesar de vários caminhos poderem convergir para uma mesma figura do diagrama, há sempre um único caminho saindo desta. Exceções a esta regra são os símbolos finais, dos quais não há nenhum fluxo saindo, e os símbolos de decisão, de onde pode haver mais de um caminho de saída.

Pseudocódigo ou portugol.

Esta forma de representação de algoritmos é rica em detalhes, como a definição dos tipos das variáveis usadas no algoritmo. Por assemelhar-se bastante à forma em que os programas são escritos, encontra muita aceitação.

Na verdade, esta representação é suficientemente geral para permitir a tradução de um algoritmo nela representado para uma linguagem de programação específica seja praticamente direta.

Todo o trabalho realizado por um computador é baseado na manipulação das informações contidas em sua memória. Grosso modo, estas informações podem ser classificadas em dois tipos:

-As instruções, que comandam o funcionamento da máquina e determinam a maneira como devem ser tratados os dados. As instruções são específicas para cada modelo de computador, pois são funções do tipo particular de processador utilizado em sua implementação.

-Os dados propriamente ditos, que correspondem à porção das informações a serem processadas pelo computador.

**6 PROJETO (ESTRUTURA DO PROGRAMA)**

O sistema possui duas funções, mais a principal função principal, uma para criptografar e a outra para descriptografar mensagem.

Criptografar: a função criptografar cria ponteiro que chama um arquivo de texto, logo em seguida tenho uma condição para saber se o ponteiro conseguiu abrir o arquivo de texto, caso ele não consiga deve retornar uma mensagem “Houve um erro ao abrir o arquivo”.

Em seguida uso o “fflush” para limpar o buffer do teclado e peço ao usuário que digite a mensagem que deseja criptografar, capturo a mensagem e gravo na variável “arquivo” (do tipo char).

Para criptografar a mensagem eu uso um “for”, e para cada caractere incrementa + 50, após criptografar, gravo a mensagem dentro do arquivo de texto e mostro a mensagem criptografada na tela, e ao digitar qualquer teclado limpa a tela e volta ao menu principal.

Descriptografar: Ao começar a função já limpo o buffer do teclado usando o fflush, e chamo o arquivo de texto, para ler a mensagem contida nele, em seguida tem uma condição que verifica se conseguiu ler o arquivo de texto, caso não consiga ele deve mostrar a mensagem “Não foi possível abrir o arquivo!!”.

Para descriptografar a mensagem eu uso um “while” que busca a mensagem no arquivo de texto e grava na variável “arquivo”, e dentro do mesmo uso um for para ir descriptografando cada letra do texto contido no arquivo, no “for” ele subtrai 50 (-50) para cada caractere digitado depois mostra a mensagem descriptografada na tela.

Função (main): Na função principal eu tenho um while, e peço ao usuário que digite 1 para criptografar, 2 para descriptografar e 3 para sair, gravo o valor digitado em uma variável escolha, em seguida tenho um switch case para cada caso, Caso 1, eu chamo a função criptografar, Caso 2, eu chamo a função descriptografar, caso 3, sair.

Apesar de a criptologia estar bastante avançada na época, em 50 a.C. César usava um sistema bastante simples de substituição. Suetônio, escritor romano que viveu no início da era cristã (69 d.C.), em Vida dos Césares, escreveu a biografia dos imperadores romanos, de Júlio César a Domiciano.  
 Conta que Júlio César usava na sua correspondência particular um código de substituição no qual cada letra da mensagem original era substituída pela letra que a seguia em três posições no alfabeto: a letra A era substituída por D, a B por E, e assim sucessivamente.  
 Hoje em dia, porém, a denominação de Código de César é utilizada para qualquer cifra na qual cada letra da mensagem clara seja substituída por outra deslocada um número fixo de posições, não necessariamente três.

Um exemplo é o código que, ainda segundo Suetônio, era usado por Augusto, onde a letra A era substituída por B, a B por C e assim sucessivamente.  
 Como o alfabeto romano possui 26 letras, é possível obter 26 alfabetos cifrantes diferentes, dos quais um (o do deslocamento zero) não altera a mensagem original.

Cada um destes alfabetos cifrantes é conhecido como Alfabeto de César.  
Características:  
\* Origem: Usada pelo imperador romano Júlio César em 50 a.C.  
\* Classe: Substituição Simples.  
\* Tipo: Monoalfabética (porque usa apenas UM alfabeto cifrante) Monogrâmica (porque trata cada UM dos caracteres individualmente).  
\* Segurança: Baixíssima  
\* Uso: Aplicável apenas em textos muito curtos.  
\* Criptoanálise: Uma simples criptoanálise baseada na característica estatística da língua é suficiente para decifrar o texto.  
A substituição original do código de César encontra-se na tabela abaixo:  
  
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  
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C  
  
O Código de César original tem um deslocamento de 3 posições, porém é possível usar qualquer deslocamento. O deslocamento zero mantém a mensagem original.

Entramos com um sistema interativo da cifra de César onde o usuário digita o valor de casa que ele irá deslocar. A partir deste temos também uma adição de um caractere especial entre cada caractere do arquivo original, e esse vária caso o numero do caractere do arquivo for par ou ímpar.

**6.1 RELATÓRIO (LINHAS DO CODIGO)**

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

//Função Criptografar

int criptografar()

{

int i;

char arquivo [10000];

FILE \*fp;

fp = fopen ("Arquivo.txt", "w");

if (fp == NULL)

{

printf ("Houve um erro ao abrir o arquivo.\n");

return 1;

}

fflush(stdin);

printf("Digite o que deseja gravar:\n");

gets(arquivo);

for( i = 0; i < strlen(arquivo); i++ )

{

arquivo[i] += 50;

}

printf(">> Arquivo Gerado: %s\n\n ", arquivo);

for(i=0; arquivo[i]; i++)

{

fputc(arquivo[i], fp);

}

printf ("Arquivo criado com sucesso.\n");

fclose(fp);

printf("Digite qualquer tecla para voltar ao menu principal!");

getchar();

system("cls");

}

//Função Descriptografar

int descriptografar()

{

fflush(stdin);

int i;

char arquivo [10000];

FILE \*file;

file = fopen("Arquivo.txt", "r");

if(file == NULL)

{

printf("Não foi possivel abrir o arquivo!!\n");

getchar();

}

system("cls");

while(fgets(arquivo, 100000, file) != NULL)

{

for( i = 0; i < strlen(arquivo); i++ )

{

arquivo[i] -= 50;

}

printf("%s", arquivo);

}

printf("\n\nDigite qualquer tecla para voltar ao menu principal!");

getchar();

system("cls");

}

int main()

{

int escolha=0;

while(escolha < 3)

{

printf("Bem Vindo ao sistema de criptografia!\n\n\n");

printf("Digite 1:\nPara criptografar um arquivo.\n\nDigite 2:\nPara descriptografar um arquivo.\n\nDigite3:\nPara Sair.\n");

scanf("%d",& escolha);

switch(escolha)

{

case 1:

criptografar();

break;

case 2:

descriptografar();

break;

case 3:

exit(0);

break;

}

}

**7 REFERÊNCIAS**

**GUIA DE NORMATIZAÇÃO PARA TRABALHOS ACADÊMICOS DA UNIVERSIDADE PAULISTA**, revisado e atualizado pelas bibliotecárias Alice Horiuchi e Bruna Orgler Shiavi / Biblioteca Universidade Paulista, 2016.

ASCENIO, Ana Fernanda Gomes. **Fundamentos da Programação de Computadores: Algoritmos, Pascal, C/C++ e JAVA.** 2.ed. São Paulo: Pearson Prentice Hall, 2007

FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico**. Lógica de Programação:** A construção de Algoritmos e Estruturas de Dados. São Paulo: Pearson, 2005.

FARRER, Harry [et al]. **Algoritmos Estruturados.** 2.ed. Rio de Janeiro: Guanabara Koogan, 1989. 252p.

ZIVIANE, Nivio.**Projetos de Algoritmos com Implementações em Pascal e C.** São Paulo: Pioneira, 2004.

ANZANO, J. A. N. G; OLIVEIRA, J. F. O. Algoritmos: Lógica para Desenvolvimento de Programação de Computadores. 27. ed. São Paulo: Érica, 2014. 328 p.

FORMOLO, D. Criptografia com Caos. São Leopoldo: Universidade do Vale do Rio dos Sinos, 2009. Disponível em: <http://livros01.livrosgratis.com.br/cp106771.pdf>.

SEMENTE, R. S. Otimização de Algoritmos Criptográficos para Redes de Sensores e Atuadores Sem-fio para Poços do Tipo Plunger Lift. Rio Grande do Norte: Universidade Federal do Rio Grande do Norte, 2011. Disponível em: <http://repositorio.ufrn.br/jspui/bitstream/123456789/15359/1/DISSERTACAO%20RO DRIGO%20SOARES%20SEMENTE.pdf>.

PALMA, S. M; PERREIRA, A. A. S. Análise Crítica da Implementação da Cifra RC4 no Protocolo WEP. São Paulo: Universidade Estadual de Campinas. Disponível em <http://www.lasca.ic.unicamp.br/home/media/publications/1WEP.pdf>.

LINHARES, A. G; GONÇALVES, P. A. S. Uma Análise dos Mecanismos de Segurança de Redes IEEE 802.11: WEP, WPA, WPA2 e IEEE 802.11w\*. Pernambuco: Universidade Federal de Pernambuco. Disponível em: <http://www.cin.ufpe.br/~pasg/gpublications/LiGo06.pdf>.

TRINTA, F. A. M; MACÊDO, R. C. Um Estudo sobre Criptografia e Assinatura Digital. Disponível em: <http://www.di.ufpe.br/~flash/ais98/cripto/criptografia.htm>.

INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO. Criptografia. Disponível em: <http://www.dei.isep.ipp.pt/~andre/documentos/criptografia.html>.

CANALTECH. Google, Microsoft e Mozilla irão abandonar criptografia RC4 em seus navegadores. Disponível em: <http://canaltech.com.br/noticia/seguranca/google-microsoft-e-mozilla-irao- abandonar-criptografia-rc4-em-seus-navegadores-48481/>.

DEVMEDIA. Utilizando Criptografia Simétrica em Java. Disponível em: <http://www.devmedia.com.br/utilizando-criptografia-simetrica-em-java/31170>. Acesso em

<http://pt.wikipedia.org/wiki/Criptografia>

<http://cartilha.cert.br/criptografia/>

<http://www.techtudo.com.br/artigos/noticia/2012/06/o-que-e-criptografia.html>

<http://informatica.hsw.uol.com.br/criptografia.htm>

<http://olhardigital.uol.com.br/video/o-que-e-criptografia/32637>

[http://pt.wikipedia.org/wiki/Cifra\_de\_C%C3%A9sar](http://pt.wikipedia.org/wiki/Cifra_de_César)

<http://cmup.fc.up.pt/cmup/ajmachia/html/slide_2.html>

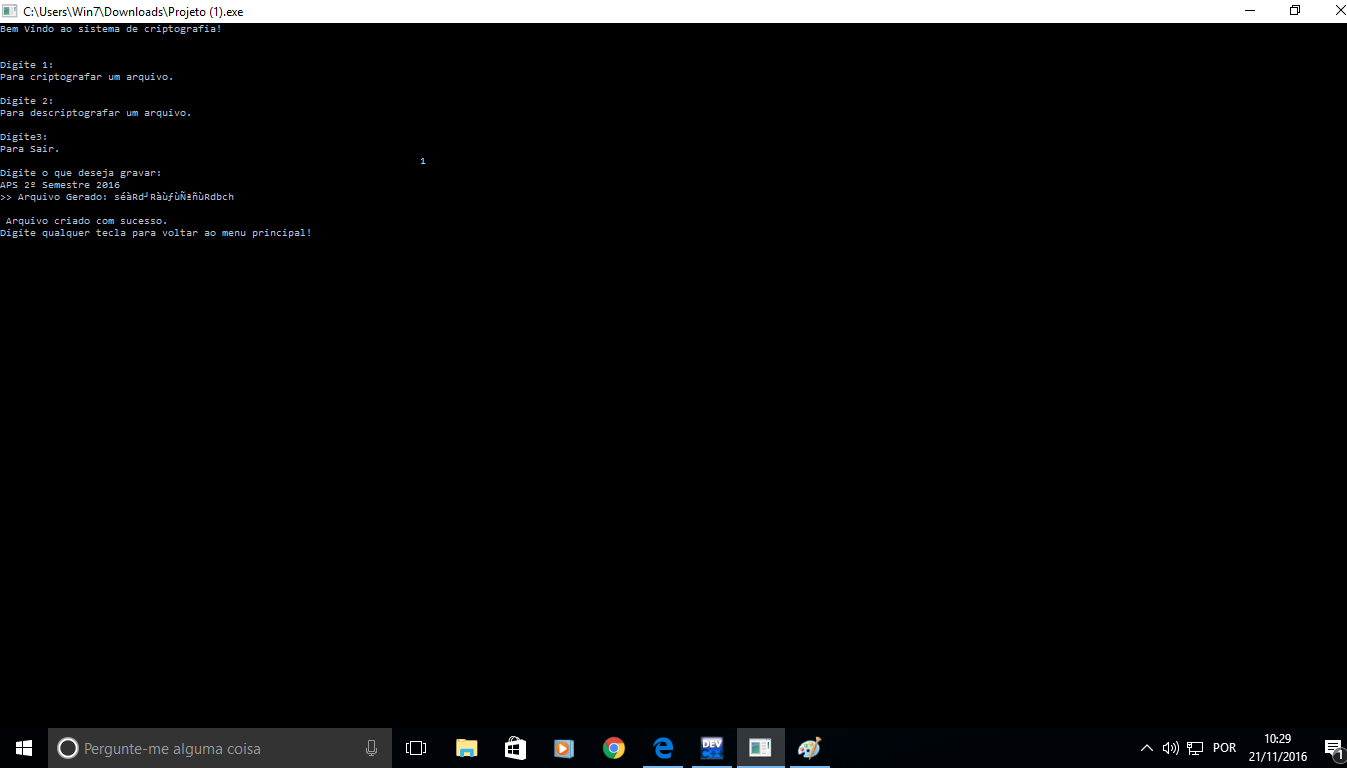
<http://setesys.com.br/blog/como-produzir-senhas-criativas-utilizando-a-cifra-de-cesar/>

[http://www.mcsesolution.com/Seguran%C3%A7a/a-matematica-da-cifra-de-cesar.html](http://www.mcsesolution.com/Segurança/a-matematica-da-cifra-de-cesar.html)

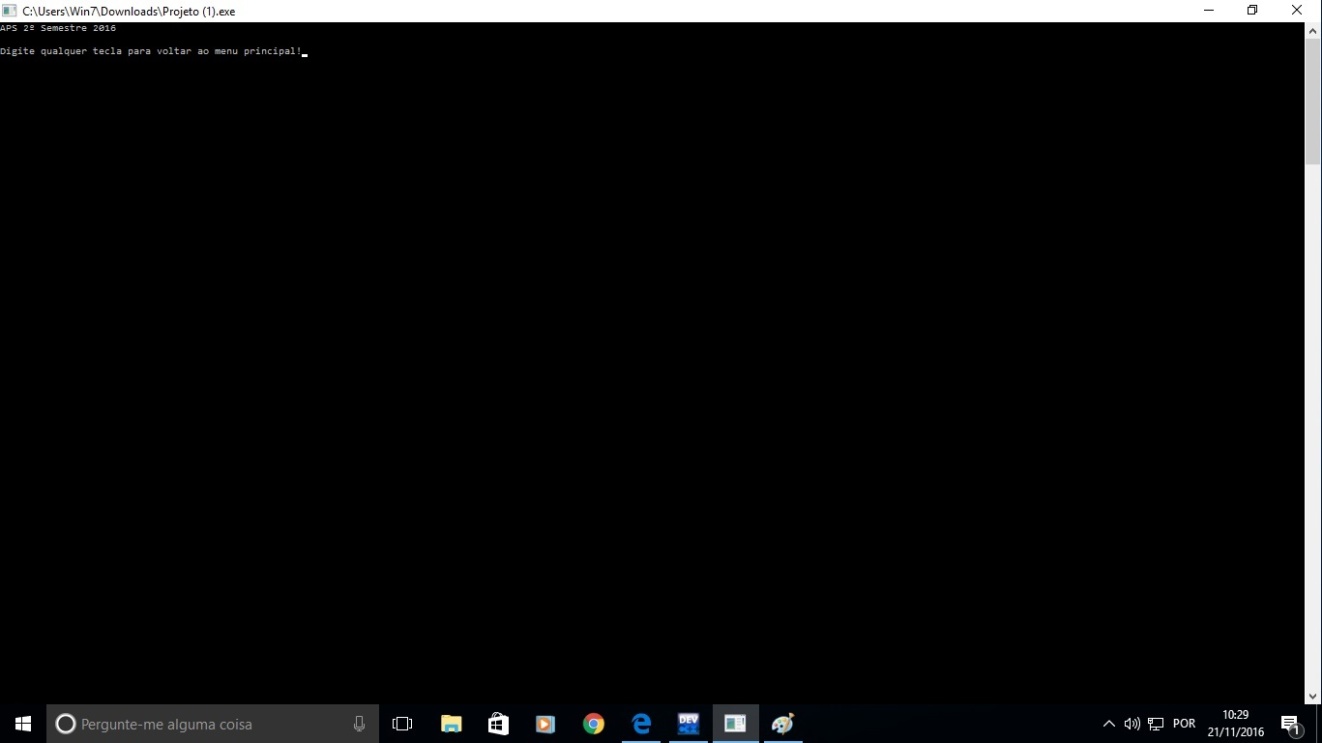
[http://pt.wikipedia.org/wiki/Cifra\_de\_substitui%C3%A7%C3%A3o](http://pt.wikipedia.org/wiki/Cifra_de_substituição)

**8 ANEXOS**

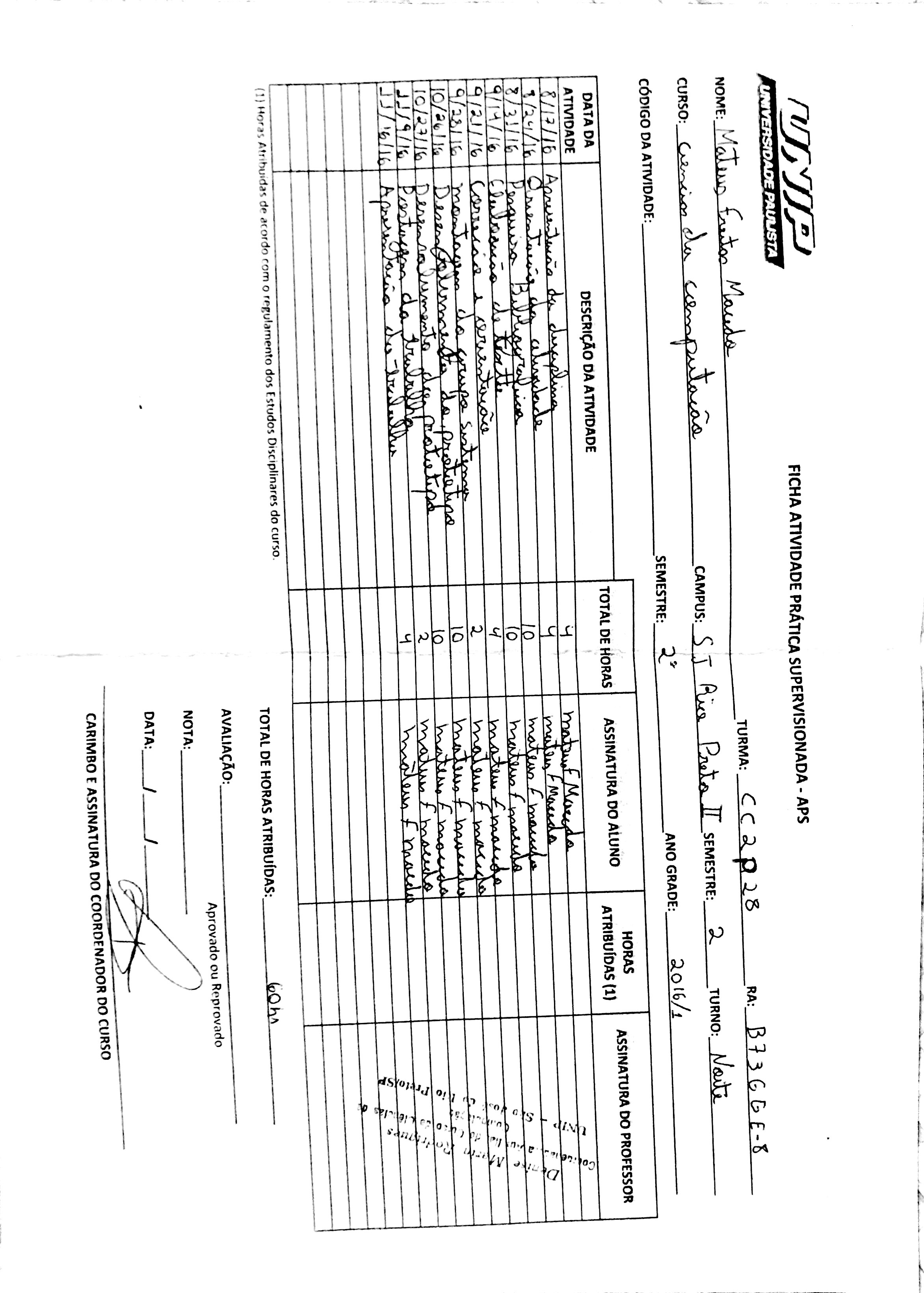
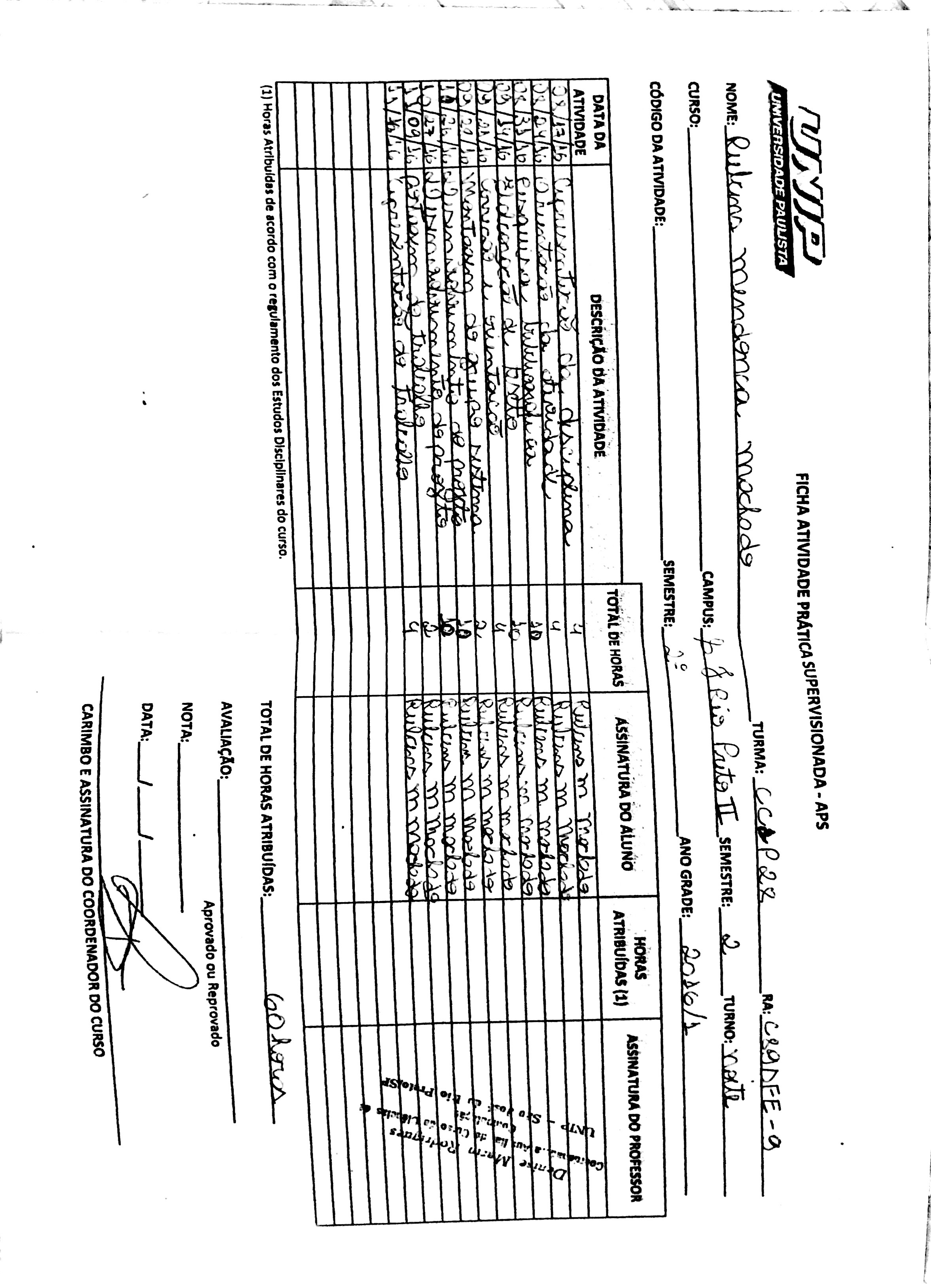
Terminal com o programa rodando

****

Descriptografando

****

**9 ANEXOS FICHAS**



RICARDO%20-%20UNIP