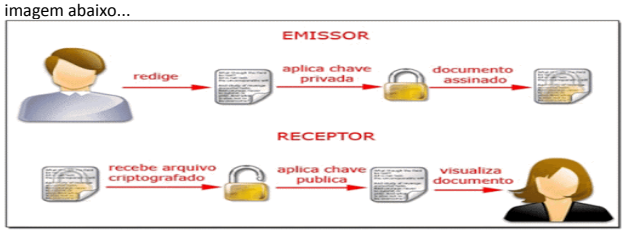
FUNÇÕES CRIPTOGRÁFICAS	FUNÇÕES CRIPTOGRÁFICAS
O que são **processos criptográficos**?	**Processos Criptográficos:** É o nome que se dá aos **processos usados pelos sistemas de criptografia para cifrar e decifrar uma mensagem, esses processos são elaborados por através de um algoritmo responsável por cifrar e/ou decifrar a mensagem.** Para cada estilo de criptografia podem existir um ou mais processos criptográficos.
Ilustre como o **Processo Criptográfico acontece** para a cifragem e a decifragem...	**Cifragem:** Mensagem (Olá tudo bem?) **Processo de Cifragem:** Algoritmo que troca as letras por duas letras superiores em ordem alfabética e retira acentuações do idioma português... **Mensagem Cifrada:** (QNC VXFQ DGO?) ---------- envio de mensagem ---------- **Processo de Decifragem:** Algoritmo que troca as letrar por 2 anteriores em ordem alfabética e adiciona acentuações segundo o idioma português... **Mensagem Decifrada:** (Olá tudo bem?)
O que são **Funções Criptográficas?**	**Funções Criptográficas:** As funções criptográficas são **processos matemáticos que oferecem uma camada á mais de proteção as mensagens criptografadas além da cifragem e das chaves públicas e privadas.** Dentre as Funções Criptográficas, temos 2 mais utilizadas, que são: **- HASH** **- Assinatura Digital** Essas funções, por assim dizer "embaralham aquilo que já estava embaralhado", tornando ainda mais difícil para uma pessoa não autorizada ler uma mensagem que não é sua ou se passar por uma pessoa que não é.
Como se dá o **HASH**?	**HASH:** É uma função criptográfica que **gera um número verificador impossível de inverter, a menos que a pessoa tenha o número original do HASH é impossível decifrar a mensagem.** Isso acontece por que o HASH tem uma função algorítmica que inverte todos os bits de uma mensagem em uma única **expressão numérica**, como por exemplo **298**. Depois de gerar a expressão numérica, HASH precisará determinar um **número verificador** para aquela expressão, digamos que fosse o valor do resultado do **resto da divisão de 298 por 14**, que geraria o valor **4**. Neste caso, o **número verificador** seria o **4**, o HASH envia essa informação junto com a expressão numérica ao destinatário, que irá comparar o verificador informado com o valor na mensagem, caso haja a menor alteração binária, seja no valor da expressão ou o número HASH, elas são detectadas imediatamente, e a mensagem se torna ilegítima. Por isso dizemos que o HASH é uma **função unidirecional**, pois é fácil calcular o número HASH de uma mensagem, mas é impossível descobrir uma mensagem somente sabendo o número verificador. **Envio --> 298-4** **---> Chegada: 298 / 14 quanto é? --> 4** **----------> 4 é igual a 4? --> SIM - MENSAGEM VERDADEIRA** ***Esse é apenas um exemplo simples, a numeração HASH e o código de Verificação geram valores muito mais complexos.**
- reservado para a questão acima -	
Ilustre o uso de HASH	Podemos ilustrar a função HASH com o exemplo do CPF, imagine o CPF Legítimo: **123 456 789 - 88** Os 2 últimos digítos do CPF são usados como número verificador, onde os primeiros 9 digítos irão passar por um algoritmo para gerar esse **número verificador "88"**, caso alguém tente adulterar o CPF digitando... **456 123 789 - 88** Os computadores iriam submeter esse número a função HASH e descobrir que **os números não batem com o dígito de verificação**, descobrindo que o número é falso.
O que é a **colisão de HASH**?	**Colisão de HASH:** é quando temos **duas mensagens que geram o mesmo código HASH**, isso é praticamente impossível de acontecer, pois o HASH é calculado encima do valor binário da mensagem e ainda é submetido ao algoritmo do programa. Mas caso aconteça, lembre-se que isso não gerará problema nenhum, pois a comparação do número HASH é feita no destinatário a partir da mensagem enviada em conjunto ao número HASH, portanto ele não vai comparar um valor HASH com uma mensagem enviada anteriormente ou posteriormente, a comparação é feita instantaneamente sobre a mensagem enviada junto ao número HASH.
Por que podemos dizer que o HASH possuí um **tamanho fixo**?	Por que, **não importa o tamanho físico de uma mensagem, todos os número HASH terão um mesmo tamanho físico.** Por exemplo, digamos que o HASH tenha o tamanho fixo de 20 bytes, não importa de a mensagem tem 30 megabytes ou 40 Gigabytes, o HASH de ambas terá sempre o tamanho de 20 bytes.
Como se dá a **Assinatura Digital**?	**Assinatura Digital:** Como o próprio nome diz, é uma Assinatura que tem o papel de **Autenticar que uma mensagem realmente vêm do remetente que ela afirma vir.** Assim como na Criptografia Assimétrica, a Assinatura Digital também utiliza par de chaves, porém o processo acontece de forma invertida, na criptografia o Destinatário é quem possuía a Chave Privada e enviava a Chave Pública aos remetentes para que eles pudessem lhe enviar mensagens. Mas na Assinatura Digital acontece o contrário, o Remetente possuí a Chave Privada e a usa para autenticar um arquivo, e o destinatário recebe a Chave Pública, que é usada comente para abrir o arquivo enviado pelo remetente, como podemos ver na imagem abaixo...
- reservado para a questão acima -	

FUNÇÕES CRIPTOGRÁFICAS	FUNÇÕES CRIPTOGRÁFICAS
Por que podemos dizer que o **HASH, Assinatura Digital e Criptografia caminham juntos**?	Por que não as usamos separadamente, **uma depende da outra, primeiro criptografamos a mensagem, depois usamos a função HASH para garantir sua integridade durante a transmissão, e por fim assinamos o arquivo por através da Assinatura digital.** E juntas elas garantem alguns princípios e conceitos da segurança da informação: **HASH -** (garante a Integridade dos Dados) **Assinatura Digital -** (Garante a Autenticidade dos Emissores) **Criptografia -** (Garante a Confidencialidade das Informações) **Juntas -** (Garante o Não-Repúdio, os autores não podem negar autoria)
Ilustre o uso na prática de cada uma destas tecnologias