MAC 0336/5723 - Criptografia e Segurança de Dados - Lista de exercícios 1 (USP) - 2019

- prazo de entrega: veja no paca.ime.usp.br
- resolver *individualmente*, duas soluções idênticas receberão nota zero na lista toda
- entregue as suas soluções no sistema PACA, digitado, em formato PDF
- manuscritos **não** são corrigidos
- escreva no cabeçalho o seu NUSP e nome completo

Notações usadas nesta lista:

1. Os valores inteiros de 10 a 15 são representados na base 16 pelos símbolos A, B, C, D, E, F, respectivamente. Denotamos um byte de 8 bits, na base 16, da seguinte maneira: para X, Y de 4 bits, $XY = (XY)_{16} = X \times 2^4 + Y$. Exemplos: $(3A)_{16} = 3 \times 2^4 + 10 = 58$ e $(2A)_{16} = 2 \times 2^4 + 10 = 42$.
2. $\lceil x \rceil$ é *teto* de x. Exemplos: $\lceil 2.59 \rceil$ vale 3, e $\lceil 2.01 \rceil$ vale 3.

Exercício 1 (40%) São dadas n informações $X = \{x_1, x_2, ...x_n\}$ ocorrendo respectivamente com as respectivas probabilidades de ocorrerem: $p(x_1), p(x_2), ...p(x_n)$.
Este exercício é para:

1. Calcular a entropia de X para $p(x_1) = 1/16, p(x_2) = 1/4, p(x_3) = 1/16, p(x_4) = 1/4, p(x_5) = 1/4, p(x_6) = 1/16, p(x_7) = 1/16$.
2. Demonstrar (i.e., provar matematicamente) que, para $j = 1, 2, ...n$, se $\max_j \lceil \log_2 [\frac{1}{p(x_j)}] \rceil$ representa o comprimento **suficiente** de bits para codificar cada um dos $x_j : j = 1, 2, ...n$.
3. Demonstrar que $\log_2 n$ é a entropia **máxima** de qualquer X.Sugestão: Supor dado o Lema: "A função $\log_2()$ é estritamente côncava". Aplicar o Teorema de Jensen: "Se $f : \mathbb{R} \to \mathbb{R}$ é uma função contínua estritamente côncava no intervalo I, então $\sum_{i=1}^{n} a_i f(x_i) \leq f(\sum_{i=1}^{n} a_i x_i)$, onde, para $1 \leq i \leq n : a_i \in \mathbb{R}$, $a_i > 0$ e $\sum_{i=1}^{n} a_i = 1$."
4. Para qual conjunto X essa entropia **máxima** ocorre? Demonstrar esse fato.

Exercício 2 (20%) Este exercício é sobre *multiplicação* de um vetor de 8 bits, um byte, por outro de 8 bits, sobre o Corpo de Galois $GF(2^8)$, conforme as páginas 93, 94, e 272 a 273 do livro-texto, que é usada na definição do AES (Advanced Encryption Standard). Denotaremos a multiplicação por $\otimes$. Por exemplo, $(45)_{16} \otimes (0A)_{16} = (94)_{16}$. Por definição $m(x) = x^8 + x^4 + x^3 + x + 1 = (11B)_{16} = (100011011)_2$.
Este exercício é para:

1. Escrever TODOS os passos dos seguintes cálculos: $(B2)_{16} \otimes (15)_{16}$, ou seja, escrever:
    1. os dois polinômios $s(x), t(x)$ correspondentes a esses dois operandos, e
    2. o polinômio produto $u(x) = s(x) \times t(x)$, e

3. os polinômios quociente $q(x)$ e resto $r(x)$ resultantes da divisão $u(x)/m(x)$.

2. Escrever TODOS os passos do cálculo da inversa $r^{-1}(x) \bmod m(x)$ utilizando o Algoritmo de Euclides estendido, ou seja, escrever TODOS os polinômios intermediários que são quociente e resto resultantes de cada divisão efetuada por esse algoritmo.

3. Escrever TODOS os passos da verificação que $r^{-1}(x) \otimes r(x) = 1 \bmod m(x)$ ou seja, escrever:

   1. o polinômio produto $U(x) = r^{-1}(x) \times r(x)$, e
   2. os polinômios quociente $Q(x)$ e resto $R(x)$ resultantes da divisão $U(x)/m(x)$.

Exercício 3 (20%) Este exercício é sobre *multiplicação* de um vetor de 4 bytes por outro de 4 bytes, sobre o Corpo de Galois $GF(2^{32})$, conforme as páginas 99, 100, e 272 a 273 do livro-texto, que é usada na definição do AES.
Denotamos um vetor de 32 bits por um vetor de 4 bytes sobre $GF(2^8)$, $[a_3, a_2, a_1, a_0]$. Tal vetor é representado polinomialmente por $A(x) = a_3x^3 + a_2x^2 + a_1x + a_0$.
É usada a *soma* de um byte por outro byte, denotada por $\oplus$; por definição a soma é o ou-exclusivo (xor) bit por bit. Por exemplo: $(45)_{16} \oplus (78)_{16} = (3D)_{16}$. A multiplicação de um byte por outro byte, a_ib_j, é calculada conforme o exercício anterior.
A *multiplicação de dois vetores*, $A(x)$ e $B(x)$, cada um de 4 bytes, é descrita no livro-texto. $M(x) = x^4+1$ é fixo, é tal que $x^j \bmod M(x) = x^{j \bmod 4}$.
Este exercício é para escrever TODOS os passos para calcular: $(B255873D) \otimes (127BC466)$, ou seja, escrever:

1. os dois polinômios $A(x), B(x)$ correspondentes a esses dois operandos, e

2. o produto, polinômio de grau 6, $C(x) = A(x) \times B(x)$, e

3. os polinômios quociente $Q(x)$ e resto $R(x)$ resultantes da divisão $C(x)/M(x)$.

Exercício 4 (20%) Este exercício é sobre $MixColumns()$. A multiplicação de 4 bytes por 4 bytes, $T(x) \otimes U(x)$, é calculada conforme o exercício anterior. É usado o vetor fixo $c(x) = (03)_{16}x^3 + (01)_{16}x^2 + (01)_{16}x + (02)_{16}$. Como $c(x)$ e $M(x) = x^4 + 1$ são co-primos (i.e., relativamente primos), $c(x)$ possui inversa $c^{-1}(x) \bmod M(x)$: $c^{-1}(x) = (0B)_{16}x^3 + (0D)_{16}x^2 + (09)_{16}x + (0E)_{16}$ e $c(x) \otimes c^{-1}(x) = 1$.

- A operação $MixColumns(A(x))$, utilizada no AES, consiste em calcular $B(x) = A(x) \otimes c(x)$. O resultado é de 4 bytes (32 bits).

- A inversa da transformação $MixColumns(B(x))$ opera também sobre um vetor de 4 bytes (32 bits), e consiste em calcular $A(x) = B(x) \otimes c^{-1}(x)$.

Este exercício é para escrever TODOS os passos para calcular: $MixColumns(B255873D)$, ou seja, escrever:

1. o polinômio $A(x)$ correspondente a esse operando $B255873D$, e

2. o produto, polinômio de grau 6, $C(x) = A(x) \times c(x)$, e

3. os polinômios quociente $Q(x)$ e resto $B(x)$ resultante da divisão $C(x)/M(x)$, para obter o resultado $B(x)$.

A seguir escrever os passos (1), (2) e (3) para calcular o inverso de $MixColumns(B)$, e verificar se obtém $A(x)$ original, como deveria ocorrer.