UNIVERSITÀ DEGLI STUDI "ROMA TRE"
CORSO DI STUDI IN MATEMATICA
IN450 - INFORMATICA 6
ALGORITMI PER LA CRITTOGRAFIA – A.A. 2012-2013

M. PEDICINI

COLLEZIONE DI ESAMI ED ESERCIZIO

1. Appello del 30/06/2011

Esercizio 2. *Dato il seguente testo cifrato (plaintext in lingua italiana)*

```
0001 oilfvhbsei lvruomohig favvbzimru kvjxrhmikt fvugragpwe oltctedlee rlkvjxttmq cisesf
0081 ueaazyweox otuqtuvelx iiggpwawlz citxalvtxe tiloroimbi algzcigsrm vkjioilivv umfvrb
0161 meptlzcxbr txrucvbghx gwesftindw txfqaivzvv bxttilgpci nvyqxmuvoo rqfmssdxci nxsich
0241 jzhhjkqvui ihewpwpfeg iqfmsgofzd kruvabkipx fvaizmphjw ogewcuvili lvvsdleerd gvbgeo
0321 cenwfvcmne phzkjmgyit cxkieynvft nihmugkwne esvxkmtqjr aorywimpao rtnidlefrd geempt
0401 mgoktwotvr thxccveeib einxpivbuq nityelginp fzelkqvihm awvzckhmdx cxkeoitttp gqfraw
0481 pelmicktfv ozemeempet ctqvgyltgi wvbynifkqu vitttpgrfp ltxwfimgok dmteeyrtki neostm
0561 qeslrqesox agkirmfxax twoiryebtp ggprlxeicj gengrbcytg imfnwsshee gmnehsaerz kzbwio
0641 bpavhcctfv izcqqwbign rbcgpwierv kqpqihtpcr dsryloimwe sbmwnwferx kzqesmmbii tpptal
0721 frogciugjs gbrucmqirl fvczjzaifq elfmphjivs vrphtwkpds riftcwtsrb gzgwjzitgm tpbtit
0801 emsxibcwjg hxcxkigirf fagqqvexii ntjybtjaqi eicvfywetm aetwomogit ilgpfvttlv cpprzt
0881 jirtvxtitx afftvsdlew zxgpnechci vsfvavfdgv ueegfvomtm ptibkeemnt ehkemzoekw crammi
0961 bxagkwkpnm ovruomosca znwmqirkzb qvoerizcxs mxeoftvsui mivzchbppk zvemqmowvt oeuxig
1041 ppmhebczbr sntwpuvile vavimpevym teogogccku venwftcqpv dbmqpsnssl vlktsmmthc gpmich
1121 mpelzkjeci nxjxgvbvmx iieehmogvl kuvilernki seaerociux aivtnimsrt umnxfqphvt chppcx
1201 jsnxdipsow ivymrevvag fvomeislvt czjwtttpgq btptidghvr lxfvguvism zxcvfecavk qruvaf
1281 jwsxtwppbx elkinxbich ezcfcmolrn cqfwivymre siavymnefv egvbtinisl vmfyoelngi elfhim
1361 cvafvagqcm aorkcvdenx ciuybqazim bdbimhcbgk frtbwmimbz iovzivbqej lmuxbqiifz uiuenm
1441 sevxqhcgpr ltgiwvbghn jkkeemsnrd kwuecazwri shebciutfv agqifimelm vhbefuutcm syfmca
1521 prtbvzkedu ubjbcihmuz emnxfqphtp gtfvdxitqj bgevympxvx tbjcqmqinl zmttjenzvm ueuxrb
1601 bpmbwmeime bxjbketens rxcgfghxmm piohofzves oxrhrxqgpe phtwomsmpb xvgzbpawfd gptslm
1681 frtkvkjmss vbeixejrbt jaqppgowzv cramaezweg immbjqhypj fxibqgimpx itwrhssbcm pdjspt
1761 jschhccres vbuqestxub emnksenwza gvusmbjmti sidbdmivjh abrtwmryae tpgxvwibfl qqcvah
1841 dirmfzkwqy olvukrprof fwoshmaylq gpjtakvvvm nmebwctsop ofsithjqag kwcrjtekgi vvjeaf
1921 jravhckwvf incqqeogok tpgjpwsxki thjivbjake ssmtjwvxpp bnfvqevkul kwpimxefgw fimmdx
2001 tmeulokesh iifmvegyix tipxbmdbhc gphmulkwhm hpinftfeog hbjmelfzeg emfmuvobrx qmdlee
2081 sfobcqqrgy chdjwwusmt kcrisghxiq vssritkipx brobrxgvdl egfvuemmie uqniuxolfu qruica
2161 ogiizwggbk ihelkxvxtt xqqmbsrlvb wuvilozzim mmoxhcgpme fhebggiisi rvfmemptit cvtmlt
2241 jymxiqutvs sbftwmdsno vzishrolrn tsoxehumnm bptkzxqium ogfzgimymx miipjembct wrhssm
2321 fpgkrvfibq okvkjinlay rbvsdirvrz nsuyooftwq fxulvtqqjs mtvavvpilf zwcyusrxkc uitslh
```

Tabella delle frequenze del testo:
(a) 0.0252 (b) 0.0388 (c) 0.0408 (d) 0.0168 (e) 0.064 (f) 0.04 (g) 0.0424 (h) 0.03
(i) 0.0784 (j) 0.0264 (k) 0.0328 (l) 0.032 (m) 0.0656 (n) 0.0256 (o) 0.0328 (p) 0.0388
(q) 0.0324 (r) 0.0388 (s) 0.0388 (t) 0.0584 (u) 0.026 (v) 0.0628 (w) 0.0332 (x) 0.0392
(y) 0.0144 (z) 0.0256

Tabella delle frequenze dell'italiano (in ordine decrescente):
(e) 0.1073 (a) 0.0983 (i) 0.0913 (o) 0.0872 (n) 0.0614 (r) 0.0603 (l) 0.0537 (t) 0.0526
(s) 0.0516 (c) 0.0471 (d) 0.0340 (u) 0.0313 (m) 0.0272 (p) 0.0250 (v) 0.0184 (g) 0.0166
(h) 0.0165 (f) 0.0115 (q) 0.0070 (b) 0.0064 (z) 0.0042 (k) 0.00001 (x) 0.00001
(j) 0.
(y) 0. (w) 0.

(1) Applicare la crittoanalisi statistica al cifrato al fine di stabilire se è stato utilizzato un sistema di cifratura monoalfabetico.
(2) In caso il cifrario utilizzato non sia un cifrario monoalfabetico, considerando l'utilizzo di un cifrario polialfabetico (tipo Vigénere), determinare la lunghezza della chiave di cifratura.
(3) Verificare (su un unico sottoblocco monoalfabetico) la correttezza della lunghezza della chiave.
(4) Determinare infine la chiave e decifrare i primi 25 caratteri del messaggio.

Esercizio 3. *Data la seguente funzione di sostituzione*

$$\pi_S : \mathbb{F}_2^m \to \mathbb{F}_2^n$$

dove $m = 4$ *ed* $n = 8$ *è rappresentata in esadecimale dalla sequenza di* 16 *valori* $\pi_S(x)$ *al variare di* x *da* 0 *a* 15*:*

$$(F1, C5, 89, 4D, F2, B6, 7A, 3E, E3, A7, 6B, 2F, D4, 98, 5C, 1F).$$

(1) Illustrare il metodo per calcolare la tabella di distribuzione delle differenze $N_D(\Delta_x, \Delta_y)$*.*
(2) Sapendo che i valori più significativi si ottengono (in ordine decrescente) per $\Delta_x \in \{2, 13, 15, 1, 3, 12, 14\}$ *calcolarne il coefficiente di propagazione (relativo allo xor bit-a-bit).*

Supponiamo che la S-box sia utilizzata in una rete $f : \mathbb{F}_2^4 \to \mathbb{F}_2^{16}$ *in modo che l'output (a 8-bit) della sostituzione al primo round diventi input di due S-box in parallelo che utilizzano la stessa sostituzione* π_S*:*

$$\pi_S(x) = y_1||y_2 \text{ e } f(x) = \pi_S(y_1)||\pi_S(y_2).$$

(3) Trovare un differential-trial per la funzione f*, e calcolarne il coefficiente di propagazione.*

Esercizio 4. *Sia* I *l'ideale generato dai polinomi* $f_1, f_2 \in \mathbb{F}_3[x, y]$*:*

$$f_1 = x^3 - 2xy \qquad f_2 = x^2y - 2y^2 + x$$

con $x < y$ *ai fini della definizione dell'ordine monomiale.*

(1) Mostrare che l'insieme $\{f_1, f_2\}$ *non é una base di Gröbner.*
(2) Calcolare una base di Gröbner per I *rispetto all'ordine lessicografico puro. Calcolare inoltre una base per* $in(I)$*.*

(3) Calcolare la base di Gröbner ridotta.

5. Esercizi dal corso di Adi Shamir (Winter 2011)

Esercizio 6. *(1) Solve the questions in slide 20*
(2) Show how to break the Rabin hash function construction

Esercizio 7. *1) Given a hash function built using the Merkle-Damgard construction, show how to efficiently find a second preimage for a message composed of more than* $2^(n/2)$ *blocks.*
2) Show how to find an expandable message of size between l *and* $2^(l)+l-1$ *blocks in time complexity* $O(l*2^(n/2))$ *(for a hash function using the Merkle-Damgard construction).*
3) Show how to efficiently find a collision in k *Merkle-Damgard hash functions concatenated together* $(h1(M)||h2(M),...,hk(M))$*. What is the complexity of the attack?*

Esercizio 8. *1) Prove that for a random function* f *there is an algorithm that finds* k *preimages in* $O(k*2^n)$ *time and* k*-collisions in* $O(2^(n(k-1)/k))$ *time.*
2) When we go over a message 3 *times in the same order, find the minimal number of blocks in* M *giving rise to* 2^k *colliding messages in* $h(M||M||M)$*.*
3) Assume that in the standard Merkle-Damgard construction each new block M_i *is hashed repeatedly until the top bit of the output is* 0*. Does it effect the Joux attack?*

Esercizio 9. *1) Prove that the extended Floyd algorithm always finds the entrance to the cycle regardless of a and b (the length of the tail and the length of the cycle).*
2) Can Nivasch's algorithm stop at any point other than Min in the cycle? Prove, or give a counter example.
3) Develop the most efficient way to extend the Nivasch algorithm into one which finds a collision point.
4) Generalize the Joux, Lucks 3-way collision algorithm to 4-way collisions by colliding two 2-way collisions. Find the best parameters.

Esercizio 10. *1) Solve the question in slide 24*
2) Given disturbances e(j) in W in steps 1,2 and 3, show how the message difference and the state evolve, and calculate the transition probabilities up to step 9 (in order to get a local collision).

Esercizio 11. *1) Find a 5-round differential characteristic for DES with probability higher than* $1/10486$*.*

Esercizio 12. *1) Find a disturbance vector that requires a difference in round -1.*
2) Show the resultant difference property for the first 20 rounds.
3) Solve the question in slide 6.

Esercizio 13. *1) Show a trivial collision finding attack when all 80 words Wi of SHA-0 are independent. What happens if the first k words are independent for k>16?*
2) Run the LFSR of SHA-0 and SHA-1 both forwards and backwards and evaluate the hamming weight of the difference, starting from a difference of a single bit.
3) Simplify the second preimage attack of Christophe De Canniere and Christian Rechberger in case the was no rotation in the evolution of the Ai's in SHA-0. (rewrite the relevant parts of sections 3.2,3.3,3.4 of the paper).

Esercizio 14. *1) Find (as many as you can) graph properties which are the same for all flavors of f: fi(x)=f(x+i(modN)).*

2) Assume that you are given D ciphertexts generated by the same key for different chosen plaintexts. Find the best approach to invert f on at least one of the D ciphertexts. What is the time/memory/data tradeoff?

Esercizio 15. *1) Find the possible holes in the lower bound proof of the running time of time-memory tradeoff algorithms.*

2) Assume that the random graph consists of 2 components of sizes N1, N2 (N1«N2), and we want to be able to invert only on N1. What is the best approach?

Esercizio 16. *1) Prove that every output bit of an nxn invertible Sbox can be described as a polynomial of degree at most n-1 over GF(2) in the input bits.*

2) Prove that the coefficient of tI is equal to the sum of outputs of the polynomial obtained from all inputs which assign 0 values to all variables that are not contained in I.

Esercizio 17. *1) Describe bad situations (partial attacks) when you are allowed to try more than* $2^(c/2)$ *possible inputs to a hash function based on the sponge construction.*

Esercizio 18. *1) Divide the input variables to a cryptosystem into* $x'_1, ..., x'_n$, $x''_1, ..., x''_n$.

Assume that in a given output polynomial f*, every term can contain at most* k' *variables of type* x' *and at most* $k'' > k'$ *variables of type* x''*. What is the best way to apply the cube attack to reduce the degree to* 1 *of* 0 *?*

Esame del 28/01/2003 – Tempo 3h00

Esercizio 19. *Sia* $p(x) = x^5 + x^2 + 1$ *e* $\mathbb{F}_{2^5} = \mathbb{Z}_2[x]/p(x)$.

Un crittosistema è ottenuto utilizzando la funzione SubBytes *di AES: ogni byte* $a_4a_3a_2a_1a_0$ *è trasformato in*

$$SubBytes(a_4a_3a_2a_1a_0) = b_4b_3b_2b_1b_0.$$

Codificare il messaggio $m =$ `0xDEADBEEF` *usando la modalità CBC.*

Esercizio 20. *Il seguente crittogramma si suppone sia stato ottenuto mediante Vigenére con una chiave* corta*:*

AONLO DUOFA ONLTT HTUZT TLQGL SFPOZ DHQAE DAIEU ONSED PNFOE YPNPA OEBFR LRFOE ESIZN SMUDM YRUVW EVFAB TDHGQ BSRVR FBNQV RFVTM REMYM EHGMP NEAAE LAAMT DVBXL SMUDN FOBWO EPNSL NPAHQ TTAKI QAOES EQWNA TODLA ZKBKH SXLEB AOEHY ILEZ.

Eseguire il Kasiski test, e calcolare l'indice di coincidenza per determinare la lunghezza della chiave e la chiave stessa.

Infine, decifrare il testo.

Esercizio 21. *Sia* e_k *un cifrario a 4-bit definito mediante:*

$$e_k(m) = (b_1 + b_3, b_2 + b_4, b_2 + b_3, b_1 + b_4)$$

dove $B = b_1b_2b_3b_4 = k \oplus m$.

Codificare il messaggio dato da 11010110111001110010010001001000*, usando la chiave* $k = 1011$ *in modalità:*

(1) ECB;

(2) CBC con vettore iniziale 1001*;*

(3) CFB con vettore iniziale 1001 *e dimensione dei blocchi* $r = 1$.

Soluzioni Esame del 18/02/2003

Esercizio 22. *Sia* $p(x) = x^4 + x^3 + 1$ *e* $\mathbb{F}_{2^4} = \mathbb{Z}_2[x]/p(x)$.

Un crittosistema è ottenuto utilizzando la funzione SubBytes *di AES: ogni byte* $a_3a_2a_1a_0$ *è trasformato in*

$$SubBytes(a_3a_2a_1a_0) = b_3b_2b_1b_0.$$

Codificare il messaggio, qui rappresentato in notazione esadecimale, $m =$ `0xDEADBEEF` *usando la modalità CBC.*

Soluzione. Il campo $\mathbb{F}_{2^4}$ contiene polinomi del tipo $\sum_{i=0}^{3} a_i x^i$ con $a_i \in \{0, 1\}$, dunque la corrispondenza tra la rappresentazione binaria di m ed elementi di $\mathbb{F}_{2^4}$ sarà fatta considerando sequenze di 4 bits, inoltre poichè una sequenza di 4 bits rappresenta esattamente una cifra esadecimale codificheremo una cifra esadecimale alla volta.

Il polinomio che corrisponde alla cifra esadecimale `0xD` è $z(x) = x^3 + x^2 + 1$ (ottenuto dalla rappresentazione binaria 1101).

Cerchiamo il polinomio inverso di $z(x)$ in $\mathbb{F}_{2^4}$: applicando l'algoritmo di euclide generalizzato: dunque si tratta di trovare $q_1(x)$ e $r_1(x)$ tali che $p(x) = z(x)q_1(x) + r_1(x)$ in $\mathbb{F}_{2^4}$, mediante divisione si ottiene $p(x) = z(x)x + (1 + x)$.

Il procedimento viene iterato considerando l'ultimo resto trovato come divisore fino da ottenere resto 1; cerchiamo allora $q_2(x)$ e $r_2(x)$ tali che $p(x) = r_1(x)q_2(x) + r_2(x)$ e in questo caso otteniamo $p(x) = r_1(x)x^3 + 1$.

Dunque abbiamo che

$$p(x) = z(x)q_1(x) + r_1(x) = z(x)q_1(x) + \frac{p(x) - r_2(x)}{q_2(x)}$$

quindi

$$p(x)q_2(x) = z(x)q_1(x)q_2(x) + p(x) - r_2(x)$$

e

$$r_2(x) = z(x)q_1(x)q_2(x)$$

essendo $p(x)s(x) = 0$ per ogni $s(x)$ come elementi di $\mathbb{F}_{2^4}$;

Dunque poichè $r_2(x) = 1$ abbiamo che $z^-1(x) = q_1(x)q_2(x) = x^4 \mod \mathbb{F}_{2^4} = x^3 + 1$.

Dunque $SubBytes($`0xD`$) =$ `0x9`.

In modalità CBC (cipher block chaining mode) il prossimo blocco di 4bits `0xE` viene prima posto in X-OR con il codice ottenuto al passo precedente e poi cifrato con la funzione di cifratura, dunque nel nostro caso si tratta di calcolare `0x9` $\oplus$ `0xE` = `0x7` il polinomio corrispondente è $x^2 + x + 1$, applichiamo ora un altro procedimento per ricavare il polinomio inverso. Il polinomio inverso $z^{-1}(x) = a_3x^3 + a_2x^2 + a_1x + a_0$ deve soddisfare $z(x)z^{-1}(x) = 1 \mod \mathbb{F}_{2^4}$ dunque

(1)
$$(a_3x^3+a_2x^2+a_1x+a_0)(x^2+x+1) = a_3x^3(x^2+x)+a_3x^3+a_2x^4+a_2x^3+a_2x^2+a_1x^3+a_1x^2+a_1x+a_0x^2+a_0x+a_0$$

poichè $xp(x) = 0$ abbiamo che $x^5 + x^4 = x$ e $x^4 = x^3 + 1$ dunque semplificando e raccogliendo i termini

$$z(x)z^{-1}(x) = (a_1 + a_3)x^3 + (a_2 + a_1 + a_0)x^2 + (a_3 + a_1 + a_0)x + (a_2 + a_0)$$

per il principio di identità dei polinomi se questo deve essere uguale ad 1 abbiamo che il seguente sistema tra i coefficenti di $z^{-1}(x)$ deve essere soddisfatto:

$$\begin{aligned} a_1 + a_3 &= 0 \\ a_2 + a_1 + a_0 &= 0 \\ a_3 + a_1 + a_0 &= 0 \\ a_2 + a_0 &= 1 \end{aligned}$$

dalla cui soluzione

$$\begin{aligned} a_1 = a_2 = a_3 &= 1 \\ a_0 &= 0 \end{aligned}$$

si ottine il polinomio inverso

$$z^{-1}(x) = x^3 + x^2 + x$$

corrispondente alla cifra esadecimale `0xE`.

La cifratura prosegue poi iterando questo stesso schema: dunque `0xE` $\oplus$ `0xA` = `0x4`; il polinomio di cui computare l'inverso è quindi $z(x) = x^2$ il cui inverso è banalmente $(x^3 + x)$ essendo $p(x) = z(x)(x^3 + x) + 1$.

Dunque la codifica è `0xA`.

Il passo successivo è dato da `0xA` $\oplus$ `0xD` = `0x7` la cui codifica `0xE` è stata già computata più sopra.

Il passo successivo è dato da `0xE` $\oplus$ `0xB` = `0x5` il polinomio corrispondente questa volta è $z(x) = x^2 + 1$ il cui inverso

□

Esercizio 23. *Il seguente crittogramma si suppone sia stato ottenuto mediante Vigenére con una chiave* corta*:*

```
HLXOU YHNNL IZNSM YENVR ZMRQL AZKEC BAXPN PPGZP TKHNP YISOT
EAHQF ADLEZ KOILD IPTTY EMZOZ HNPJO ZZCUL NOLAZ KSTVU XKAOA
TADAD KSAUE MUOFO EDPNM ZPUYI FVFNY OFOED OOAKE HLRKV NQPSQ
UTUAL QKTAH LXAHQ YISOT EHNPM RQLDA TSELT RVRFO IZAHU ZDQJL
MYAFP OZDIF OOGAD UZTUU CFPOZ VFMUY WPNPZ UOOAE YAOLC ASOGY
SQELM UGGHG QYEXP GUVNB VLUAI OHLAY OFOED VPUUI AUNMA IAUAX
VREVC UHLAY ISPNB YOBLR FFBUY TTVRA AHQYS FHTGZ FGYTT LRYVR
QUOPP SFPNO AIAUS THLXI EYHDQ VNFOE NHSUZ ORAHQ WOXPT UJAXQ
UDPSP PCFPO ZHLAY IZAED UAFPO ZHLEA AFBSA MTTLC ABNFY YAYTQ
YRUAO DFTAD HUJHM WEDZO ZIEXV NSZWT LTTLR UABQP NPLPQ UDQUT
FYUEA NAUSQ SFSVV QYNUU GAYUZ KEDHN KVTTL RXPMU AAFPO ZVFEV
                          VQYEU NNF
```

Eseguire il Kasiski test, e calcolare l'indice di coincidenza per determinare la lunghezza della chiave e la chiave stessa.

Infine, decifrare il testo.

Esercizio 24. *a) Sia* $h_1 : \{0,1\}^{2m} \to \{0,1\}^m$ *una funzione hash resistente alle collisioni.*

Definire $h_2 : \{0,1\}^{4m} \to \{0,1\}^m$ *nel modo seguente:*

(a) porre $x = x_1||x_2$ *dove* $x_1, x_2 \in \{0,1\}^{2m}$

(b) definire $h_2(x) = h_1(h_1(x_1)||h_1(x_2))$

Dimostrare che anche h_2 *è resistente alle collisioni.*

b) (facoltativo) Generalizzare il punto precedente ad ogni intero $i > 2$,
Definire $h_i : \{0,1\}^{2^i m} \to \{0,1\}^m$ ricorsivamente nel modo seguente:
(a) porre $x = x_1 || x_2 \in \{0,1\}^{2^i m}$ dove $x_1, x_2 \in \{0,1\}^{2^{i-1} m}$
(b) definire $h_i(x) = h_1(h_{i-1}(x_1) || h_{i-1}(x_2))$
Dimostrare che per ogni i, h_i è resistente alle collisioni.

Esame del 19/07/2004 – Tempo 3h00 - Appello B

Esercizio 25. *(20pt) Sia $p(x) = x^3 + x^2 + 1$ si consideri la S-box di AES adattata opportunamente a 3 bit e che incorpora le operazioni sul campo finito $\mathbb{F}_{2^3} = \mathbb{Z}_2[x]/p(x)$ (utilizzare la costante applicata in* SubBytes *di AES troncata ai primi 3 bit).*

(1) Definire la funzione KeyExpansion *di AES adattata a 5 round con blocco e chiave da 48bit (porre $RCon[i] =$* `0x000000000000` *per $i = 1, \ldots, 5$).*
(2) Calcolare il KeySchedule *ottenuto a partire dalla chiave* `0x0BADBADC0DE0`.

Esercizio 26. *(16pt) Il seguente crittogramma derivato da un testo in lingua inglese*

```
001 EHSFSDSOQL NPHSEYBBPQ PHSNYOQXVS YISEMPHRUK PMISDSOQLN
051 OQUFKSBZSK YLNQUFKSBZ SKPQYBBGYL WMLNXFSSNQ GMKMLNMZMK

101 MDBSYLNEHS LQLSGYLMKS LKBYZSNYBB YFSLQPXFSS EHSLYBBYFS
151 XFSSPHSLES IYLBQQWXQF EYFNPQPHYP NYOEHSLPHM KIMPOEMBBD

201 SRQMLSNYKQ LSYLNPHMKI QULPFOYLNP HMKCFSYPIQ LPMLSLPQXS
251 UFQVSMLYVS YISXUBYLNH QVSXUBCBQD SEHSLPHYPN YOXMLYBBOI

301 QGSKYKMPEM BBPHSVSQVB SQXESKPDSF BMLIYLPYWS KQDSFKYPMK
351 XYIPMQLMLP HSXYIPPHYP PHSOESFSML PHSXFQLPBM LSKXQFYBGQ

401 KPPEQNSIYN SKYBBXFSSG SLEHSFSZSF PHSOGYOBMZ SYFSIMPMTS
451 LKQXDSFBML YLNPHSFSXQ FSYKYXFSSG YLMPYWSVFM NSMLPHSEQF

501 NKMIHDMLSM LDSFBMLSF
```

presenta il seguente schema di frequenze:

A	B	C	D	E	F	G	H	I	J	K
.0000	.0597	.0038	.0193	.0308	.0578	.0154	.0482	.0270	.0000	.0482
L	M	N	O	P	Q	R	S	T	U	V
.0848	.0713	.0405	.0231	.0790	.0655	.0038	.1540	.0019	.0135	.0135
W	X	Y	Z							
.0077	.0347	.0848	.0116							

(1) Stabilire se il cifrario è polialfabetico; **Soluzione.** La lunghezza del cifrario è 520, l'indice di coincidenza del cifrario è $IC = 0.0697584$ vicino a 0.065 indice di coincidenza della lingua inglese, implica che il cifrario è monoalfabetico. □
(2) in caso affermativo,
- *eseguire il Kasiski test e stabilire la lunghezza della chiave;*
- *verificare che lunghezza della chiave stabilita al punto precedente è corretta;*

in caso negativo,

- *decidere se si tratta di testo cifrato con un cifrario* shift *o* affine*;*
- *calcolare la chiave;*

(3) decifrare il messaggio.

Esercizio 27. *(8pt) Data la seguente S-box:*

$$\pi_S = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 5 & 4 & 3 & 0 \end{pmatrix}.$$

Calcolarne la tabella della distribuzione differenziale N_D.

Esercizio 28. *(8pt) Considerare il seguente* key-stream*:*

$$(001011000010101)^\infty$$

dire se sia stato generato da un LFSR e in caso affermativo determinare il grado e i coefficienti della relazione di ricorsione lineare che lo genera.

Esercizio 29. *(3pt) Illustrare le principali modalità operative per i cifrari a blocchi.*

Esercizio 30. *(12pt) Sia $p(x) = x^3 + x + 1$ e $\mathbb{F}_{2^3} = \mathbb{Z}_2[x]/p(x)$.*

Un crittosistema è ottenuto utilizzando la funzione SUBBYTES *di AES opportunamente adattata al numero di bit. Ogni parola di bit $a_2a_1a_0$ è trasformata in*

$$SubBytes(a_2a_1a_0) = b_2b_1b_0.$$

Codificare il messaggio $m =$ `0xDEADBEEF` *usando la modalità CFB con $IV = 101$.* **Soluzione.** La funzione SubBytes di AES fornisce una S-box ottenuta convertendo una sequenza di bits $a_2a_1a_0$ nel corrispondente polinomio $\sum_{i=0}^{2} a_i x^i$, e dunque gli elementi di $\mathbb{Z}_2^3$ corrisponderà un polinomio, del quale bisognerà calcolare l'inverso in $\mathbb{F}_{2^3}$ e poi riconvertire in sequenza binaria (si omette per questo esercizio la combinazione con le costanti presenti in AES).

Dunque calcoliamo gli inversi per gli elementi di $\mathbb{F}_{2^3}$ utilizzando l'algoritmo di Euclide generalizzato.

L'inverso di 1 è 1 dunque avremo

$$SubBytes(001) = 001.$$

L'inverso di x si ottiene mediante divisioni successive: ma già dopo la prima divisione $x^3 + x + 1/x$ abbiamo

$$q(x) = x^2 + 1 \qquad r(x) = 1$$

e dunque $x^2 + 1$ è inverso di x,

$$SubBytes(010) = 101.$$

Per calcolare l'inverso di x^2 applichiamo l'algoritmo di Euclide generalizzato:

a	b	t_0	t	q	r	$temp = t_0 - t\,q$
x^3+x+1	x^2	0	1	x	$x+1$	$0 - 1\,x = x$
x^2	$x+1$	1	x	$x+1$	1	$1 - x(x+1) = x^2+x+1$
$x+1$	1	x	x^2+x+1	$x+1$	0	$x - (x^2+x+1)(x+1) = x$

e dunque l'inversa di x^2 è $x^2 + x + 1$ e la funzione

$$SubBytes(100) = 111.$$

□

Esercizio 31. *(23pt) Il seguente crittogramma derivato da un testo in lingua inglese*

```
001 GLNGMVRBYR BJWFSWPVXB GCXEHIRDQI EZEQWBINYP DQDMATTHYM
051 IVAIXPYTRP EQLOZREFHY CWHEIGYCPB AKDQYRRTIU KKRBHXRDPS

101 HEMVBORVRH WFOIYRQHLD EELVLERXBS JUCOQRAEQB DLNGMVRYQN
151 XIDDBIRPLR GMIVAILERX RRRBCKVFBJ SCKGRNRGEY SQSELRRXUV

201 WJCXIENXLM XSSTIUKKRF UEVCKVARHW FOVVTLWRYF RSVHCSRPYY
251 GGXKGUIUGQ LGGSXLSXRG LHGBJNZMOG OWNAHWFOME AEWGYRVAPD

301 QDMATTHYMI JVXKEYSQJM OJDSNYPSCY TYRCRSVMIR MQYNISRRGC
351 NMFYEQBYJS EIHBYQOHXB MEVYVJHGCT NEXRDDLRZE LLCSYRXPCK

401 WXLSXYCMPY SVCDSYVJWW YYERCHQLIL BRGRRIQNRJ CBWBSXRBKC
451 GBXKCRSCRW RDDSZBVUMG FRLSQBDLRS VHCNSZZIUC VCBSXKGCGV

501 GCRDLIEYMQ MBCBHVFMER GECRDQIEZE QWDSGUIDBF EAPIRDPVRR
551 HRKOZRECZF OVROIBMXHG UIZYVPGBXK CNELBJSCKG RJMWFTYFGM

601 FCLILBRGWY YEFIOTOWNA HRSBWRYZHQ DSNYPPYXOV AHIPOIQBQL
651 QSRQVZLQSF YREQBGLRAS QCWEAVWHLC PNIIGYVPNE IQMDJERIZF

701 ORNYPDPOJE RIWFORJRGD LVSBXJRPGE EQXRRREGQE BURIAGLLQM
751 MGLALJVFRW SLLOHNFSQC KRQGLLQMSH AXUWKRQGLL QQVRNXFMXX

801 VAIQRYJRHV RNOMANTHYM ISHPDLNLBC IISVKYBFHU RIAGLDRNEL
851 SMQYVPLPSP CCEFVXZGVP GUISCYTYRS IUOWGOIUJS RPNRWYUIFB

901 FHPCEGVWIY MXVBRLLDLR SEFRDLNGXK CIAREILLDL RSVRLDPVAI
951 VDYVNYQRQD XJBHHAKHRF EOJPVRRQHL GLREIYCBXU RCPYIPVIID

1001 POGVGMCCXW BSFHPVMANR GRRIERJRPO EFNJUCOQNA MWYUICEMGC
1051 SRGUIZMBHF VGKZSRRVRE CBPVAIU
```

presenta il seguente schema di frequenze:

A	B	C	D	E	F	G	H	I	J	K
.025	.042	.046	.033	.050	.030	.051	.035	.055	.023	.021
L	M	N	O	P	Q	R	S	T	U	V
.052	.035	.030	.026	.035	.043	.102	.048	.013	.021	.052
W	X	Y	Z							
.029	.035	.050	.016							

(1) Stabilire se il cifrario è polialfabetico; **Soluzione.** La lunghezza del cifrario è 1077, l'indice di coincidenza del cifrario è $IC = 0.0463968$ lontano da 0.065 indice di coincidenza della lingua inglese, implica che il cifrario non è monoalfabetico. □

(2) in caso affermativo,

- *eseguire il Kasiski test e stabilire la lunghezza della chiave;*

- *verificare che lunghezza della chiave stabilita al punto precedente è corretta;*

(3) decifrare il messaggio.

Esercizio 32. *(17pt) Data la seguente S-box:*

$$\pi_S = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 2 & 0 & 3 \end{pmatrix}.$$

(1) Calcolarne la tabella di approssimazione lineare.

(2) Dedurne un attacco di crittoanalisi lineare per uno schema SPN con blocco a 4 bits che impiega la seguente permutazione:

$$P = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 \end{pmatrix}.$$

K^1

K^2

K^3

K^4

(3) stabilire la quantità di coppie (x_i, y_i) (plaintext-ciphertext) sufficienti a ricavare un bit della sottochiave K^4 della SPN.

Esame del 09/07/2004 – Tempo 3h00

Esercizio 33. *(8pt) Considerare il seguente* key-stream*:*

$$110100(1110010000001101011110010100001111011110100100000011011 1)^\infty$$

generato da un LFSR. Determinare il grado e i coefficienti della relazione di ricorsione lineare che lo genera.

Esercizio 34. *(2pt) Ricordare la definizione di cifrario di tipo Feistel e mostrare che non è necessario avere una* round-function *invertibile.*

Esercizio 35. *(10pt) Sia $p(x) = x^3 + x^2 + 1$ e $\mathbb{F}_{2^3} = \mathbb{Z}_2[x]/p(x)$.*

Un crittosistema è ottenuto utilizzando la funzione SubBytes *di AES opportunamente adattata al numero di bit (e senza considerare la combinazione finale con i bit costanti). Ogni parola di bit $a_2a_1a_0$ è trasformata in*

$$\text{SubBytes}(a_2a_1a_0) = b_2b_1b_0.$$

Codificare il messaggio $m =$ `0xDEADBEEF` *usando la modalità CBC con $IV = 010$.*

Esercizio 36. *(12pt) Il seguente crittogramma derivato da un testo in lingua inglese*

```
0001 HOZESOWBZC SJPZAISJSS EJHANFQLHK QFLNCCNOIK TZWBYAAQOW YPCXNWJCMK QOOLSYEOHX IHHSPEBBNC IPVAEBPAFY KBPEBAJHWH
0101 FWZWKJSSOO PPKAJHUIWJ QHAOPABCNA SECVPYSJPF WHHEISLNCB AGOKFBWFNA ZHKTPDSIKI JPXAJBEJUO KPOAFRWHKN MYDWYWUKEZ
0201 HEBKEGNADK NHOKPOAFRE BCOSRAFWHS TLZKOWKJGK BWJYOJZSOY SJPUWOCYYI NNWJCOPNSC QZWNWJPSNR OHOCJPVALZ WJSPIONOHD
0301 AGLAQPNCOY CLAWJZWYWH AOHDAUWOHK XSDURNKUAJ OJZAKRWJCH KSONZGPDSA WFPDKEPVAJ CNICQOJAHC YEHULFKBSO OCNLWANGKJ
0401 CBPVAKPOAF RWHKNMWPDN EBYAHKJQKJ TENAOBONNS HHGKXGANJW PWKJOJZRAO QNEPAOHDAD DABKISJKBW OEQKHAHWGA OFAHKBPHQS
0501 BHOIAGDKHB NCIWUQJIJM IKPSSABKSF APINJMKQHK PVAIIOEQKB FWICJNOMQS HHCLHOUEBC BCNUCQEBPD SIAFEZWWJF KKAKBHDADW
0601 NYLHOVWVKP SHOWPQOPAR EJRKSBPKKJ JSSUCNGZWZ WAOOJZUAJH HAAAJTKHZK SWJCCJPVAJ SSOUERSJEB KQFXQZHAHE JOIKAAJHWC
0701 CPDSCKJANB IABPISPACN KZKCWYWZXQ FAWIDWGNAE QAGPARPDSH WFCACXOSNR OPKFEAGKBH DAQKQBPNMP KYAADWJOOP FKJCIEQWHK
0801 WPQDKBWJMB QFPDSNZWOP INXOJYSOKQ YQFEJUKJHD ADHWBAPAWN GZQSPKHDAI JQGQWZJWHQ NSKBHDEGKY QQNOJYSSAV WRSWNFWJUA
0901 ZOJEBPAFRE SSSWPDBKPS ZWGPNCJKAA NDNKTAOGKN DEAFOKBSDC SEZHCWRAIO DWORWASGKJ HDASRABPEB WBSSICIABP OKASWHHHWG
1001 SUKIPKHDAD NEBYAHKJCX OSNROPKFUW HLNWJYSPKB JAKFAFOAMS AFAPINJMKQ IJPWHPVAJH KPVAIIOEQK BFWICJNOMQ SHHCWJRDEG
1101 KNQDAGPNOS AONABKSFAW RUPCPWYAUC QPCPDSLNWJ YSPKBKXGAN JWPCNUOPLF EJQAPCJSVA NSYWFHLVEH ZELGKNQKIA AJHWPCNSWH
1201 HWJPSNRWAS DNKTAOGKNF EYVWNRLESN OCJBOIKIOW GPNCJKAANK APOGAMKQBK SHKLFEJQAP CJJSSFSNOS UCCKZSRABE JUHWREAGWJ
1301 RCABPHSIAB PDWOEGYWFH LVEHZELGOL SWGWJCHKUC QBFKIHDACX OSNROPKFUW HLNWJYSPKB EWAOPOJZWJ CWJWZWNUAO SIEQENQQHO
1401 NNCKIDEPQD XZWYYATQAL HBKFWJCXHC JCGLHWPEBP DSYAWHEBCP VNKICDHDEG KLSJEBCEQW JGAAOOLFEJ YHEBCKTOPO NOHDWHYWGP
1501 WYEJRKBTNK GPUUHKKKRS NPVAEBPNWY WHAISYDOJE GIKTPDSDQU APSHAGYKDA PVAPWYGWJC GKQBZUCQDS WNWOPVARWX NOPECJKTPD
1601 SYHCYGKKNY LNCBAGOKFL ESNOCJOHWJ ROZWNAQPHM WXCRAAAKBW OAWHZLHOPB CNIDAAFEJU PDFKQUDWUE WBPHSJOWWO YUKIPKPALO
1701 PESJPZWZWA OOJZUAJHHA AAJRQNWJCO JURAHOUPVW PAWUONEGAZ INEBCKINEB PAFRESSXSO ERAOVEOQAW GAHSOOKWPQ DKTPDSDAOR
1801 ABOLFKBSOO CNLWANGKJA WUPAEBPAFN QDPARXUHAH SLDCJACNKH DAFYKAIQBE YOPECJORQN WJCHDEGLAF EKRDAWOEBY KBOPOJPHKQ
1901 QDSWPDHDAO OPFKJCIEQW HQAJHANGKB HDAKKNZZLF KBSOOCNIOU EPACWJKINM IAOHEKBOLF KBSOOCNSCQ HRUKILHSWO SPAZHKINNO
2001 ZECWQREABY ASTWQPHMSD OPUCQOSAWG UKIKXGANJA PVALZWJSPI ONOHDNCQCV UKINPSHAGY KDAJCPDWJC IJQGQWZWPH DAAKISJPAN
2101 LVEHZELGWN SZZWOGGSEA IEBCEBWXZQ AGAWHNWBOR SNOSOPFELS OWQNKGOPVA ZWOGEQEHAZ WOPWJYHJKK XAQWQGAION OVWLDAJGPK
2201 PAPVALCEJH JAONAGPPVA AONPVEJCLL COEHEKBWOK
```

presenta il seguente schema di frequenze:

A	B	C	D	E	F	G	H	I	J	K
.088	.038	.044	.033	.044	.025	.027	.056	.029	.057	.068

L	M	N	O	P	Q	R	S	T	U	V
.020	.006	.053	.068	.062	.034	.019	.054	.006	.020	.013

W	X	Y	Z
.070	.009	.022	.033

(1) verificare che il cifrario utilizzato è polialfabetico; **Soluzione.** □
(2) eseguire il Kasiski test e stabilire la lunghezza della chiave;
(3) verificare che lunghezza della chiave stabilita al punto precedente è corretta;
(4) decifrare il messaggio.

Esercizio 37. *(17pt) Considerare un crittosistema con insieme dei plaintext $P = \{a, b, c\}$, insieme delle chiavi $K = \{k_1, k_2, k_3, k_4\}$ e insieme dei ciphertext $C = \{1, 2, 3, 4, 5, 6\}$ con la*

seguente matrice di cifratura:

$$M = \begin{array}{|c|ccc|} \hline & a & b & c \\ \hline k_1 & 1 & 2 & 4 \\ k_2 & 4 & 1 & 3 \\ k_3 & 1 & 5 & 6 \\ k_4 & 2 & 6 & 3 \\ \hline \end{array}$$

e le seguenti distribuzioni di probabilità: $p(k_1) = p(k_2) = 1/4$, $p(k_3) = 3/8$, $p(k_4) = 1/8$ *e* $p(a) = 1/2$, $p(b) = 1/5$, $p(c) = 3/10$.

(1) Calcolare la distribuzione di probabilità indotta su C;
(2) calcolare l'entropia $H(P)$;
(3) mediante l'algoritmo di Huffman ricavare un codice binario per rappresentare in modo efficiente gli elementi di C;
(4) calcolare l'entropia condizionata $H(K|C)$.

Esercizio 38. *(8pt) Sia (s_i) una successione in $\mathbb{F}_2$ generata tramite una relazione ricorsiva lineare di ordine k:*

$$s_{n+k} = a_{k-1}s_{n+k-1} + a_{k-2}s_{n+k-2} + \cdots + a_0 s_n.$$

considerare la matrice associata:

$$A = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & a_0 \\ 1 & 0 & 0 & \dots & 0 & a_1 \\ 0 & 1 & 0 & \dots & 0 & a_2 \\ 0 & 0 & 1 & \dots & 0 & a_3 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & a_{k-1} \end{pmatrix}.$$

Dimostrare che per ogni n, si ha $s_n = s_0 A^n$.