Università degli Studi di Milano

Corso di Laurea in Sicurezza dei sistemi e delle reti informatiche

Fondamenti di informatica per la sicurezza

anno accademico 2006–2007 docente: Stefano Ferrari

01.12.2006 — Soluzione del primo compitino — versione A

valutazioni **1** (5) ____ **2** (5) ____ **3** (5) ____ **4** (4) ____ **5** (4) ____ **6** (9) ____

Cognome _____ **Nome** _____

Matricola _____ **Firma** _____

Esercizio 1

Per ogni numero k, calcolare il corrispondente numerale nella base n indicata:

a) $k = (512)_7$, $n = 10$

b) $k = (47)_{10}$, $n = 2$

c) $k = (2\text{B})_{16}$, $n = 2$

d) $k = (432)_8$, $n = 2$

e) $k = (143)_5$, $n = 2$

f) $k = (1110101)_2$, $n = 16$

Soluzione

a) $(512)_7 = 5\cdot7^2+1\cdot7^1+2\cdot7^0 = 5\cdot49+1\cdot7+2\cdot1 = 245+7+2 = 254$

$(512)_7 = (254)_{10}$

b)

quoziente	resto
47	
23	1
11	1
5	1
2	1
1	0
0	1

$(47)_{10} = (101111)_2$

c)

base 16	2	B
base 2	0010	1011

$(2\text{B})_{16} = (101011)_2$

d)

base 8	4	3	2
base 2	100	011	010

$(432)_8 = (100011010)_2$

e) $(143)_5 = 1\cdot5^2+4\cdot5^1+3\cdot5^0 = 1\cdot25+4\cdot5+3\cdot1 = 25+20+3 = 48$

quoziente	resto
48	
24	0
12	0
6	0
3	0
1	1
0	1

$(143)_5 = (110000)_2$

f)

base 2	0111	0101
base 16	7	5

$(1110101)_2 = (75)_{16}$

Esercizio 2

Dati $a = -15$, $b = 16$ e $n = 5$, calcolare in complemento a 2 a n bit, specificando se si verifica un overflow:

1. le stringhe binarie s_a e s_b che codificano rispettivamente a e b;
2. la somma delle stringhe binarie s_a e s_b;
3. la differenza delle stringhe binarie s_a e s_b.

Soluzione

Con la codifica in complemento a 2 a 5 bit possono essere rappresentati tutti i numeri interi compresi fra -2^{5-1} e $2^{5-1}-1$. Possono pertanto essere rappresentati senza causare overflow tutti e soli i numeri x che rispettano la condizione $-16 \leq x \leq 15$.

1. $2^n+a = 2^5-15 = 17$. Codificando 17 in binario e troncando tale codifica a 5 bit si ottiene: $s_a = 10001$.

Poiché $-16 \leq -15 \leq 15$, non si è verificato un overflow.

$2^n + b = 2^5 + 16 = 48$. Codificando 48 in binario e troncando tale codifica a 5 bit si ottiene: $s_b = 10000$.

Poiché $b = 16 > 15$, si è verificato un overflow.

2. La somma binaria di 10001 e 10000, troncata a 5 bit è: $s_a + s_b = 00001$.

   Poiché s_a e s_b hanno il primo bit uguale, ma diverso dal primo bit della loro somma, 00001, si è verificato un overflow.

3. La differenza viene calcolata come somma di s_a e di $-s_b$.

10000		sottraendo, s_b
01111	+	negazione delle cifre di s_b, $\overline{s_b}$
1	=	
10000		$-s_b$
		s_b e $-s_b$ hanno lo stesso segno: si è verificato un overflow
10000	+	$-s_b$
10001	=	s_a
100001		si devono considerare solo gli ultimi 5 bit
00001		$s_a - s_b$

   Poiché s_a e s_b hanno il primo bit uguale, non si è verificato un overflow.

Esercizio 3

Una azienda specializzata in trucchi e travestimenti esibisce nel proprio catalogo i seguenti materiali:

- naso: appuntito, a patata, corvino;
- parrucca: a scodella, a spazzola, alla mohicana, a coda di cavallo, calvo;
- barba: pizzetto, alla Garibaldi.

I prodotti sono commercializzati in due kit:

- kit *transform*: una confezione composta da un naso, una parrucca e una barba;
- kit *superspy*: una confezione composta da due nasi, tre parrucche e una barba.

Si calcoli:

a) il numero di bit necessari per codificare ciascun tipo di materiale (naso, parrucca e barba);

b) il numero di bit necessari per codificare i possibili kit *transform*;

c) il numero di bit necessari per codificare i possibili kit *superspy*.

Soluzione

a)
- 3 nasi: $\lceil \log_2 3 \rceil = 2$ bit;
- 5 parrucche: $\lceil \log_2 5 \rceil = 3$ bit;
- 2 barbe: $\lceil \log_2 2 \rceil = 1$ bit.

b) Per la regola moltiplicativa, ci sono $3 \times 5 \times 2 = 30$ possibili kit *transform*, quindi servono $\lceil \log_2 30 \rceil = 5$ bit.

c) Ogni kit è composto da due nasi (fra tre disponibili), da tre parrucche (fra cinque disponibili) e da una barba (fra due disponibili). Il numero di kit possibili risulta quindi pari al prodotto del numero di nasi, parrucche e barbe formabili compatibilimente con i vincoli specificati.

Appare evidente che l'ordine degli oggetti non sia importante. È invece discutibile la possibilità di considerare le ripetizioni (i duplicati possono essere di riserva).

Nel caso siano consentite le ripetizioni, quindi, il numero di kit è dato da:

$$C_r(3,\,2) \cdot C_r(5,\,3) \cdot C_r(2,\,1) =$$
$$C(4,\,2) \cdot C(7,\,3) \cdot C(2,\,1) =$$
$$= \frac{4!}{2!\,2!}\,\frac{7!}{4!\,3!}\,\frac{2!}{1!\,1!} = \frac{7 \cdot 6 \cdot 5 \cdot 4}{2} = 2^2 \cdot 105$$

Poiché la prima potenza di 2 che supera 105 è 2^7, per codificare le possibili maglie serviranno $\lceil \log_2(2^2 \cdot 105) \rceil = \lceil \log_2 2^2 + \log_2 105 \rceil = \lceil 2 + \log_2 105 \rceil = 2 + \lceil \log_2 105 \rceil = 2 + 7 = 9$ bit.

Nel caso non siano consentite le ripetizioni, invece, il numero di kit è dato da:

$$C(3,\,2) \cdot C(5,\,3) \cdot C(2,\,1) =$$
$$= \frac{3!}{2!\,1!}\,\frac{5!}{3!\,2!}\,\frac{2!}{1!\,1!} = 5 \cdot 4 \cdot 3 = 2^2 \cdot 15$$

Poiché la prima potenza di 2 che supera 15 è 2^4, per codificare le possibili maglie serviranno $\lceil \log_2(2^2 \cdot 15) \rceil = \lceil \log_2 2^2 + \log_2 15 \rceil = \lceil 2 + \log_2 15 \rceil = 2 + \lceil \log_2 15 \rceil = 2 + 4 = 6$ bit.

Esercizio 4

Sia data la seguente formula, F:

$$F = ((p \vee q) \leftrightarrow \neg r) \rightarrow (q \wedge \neg p)$$

a) Costruire la tavola di verità di F.

b) F è una tautologia? Motivare la risposta.

Soluzione

a) La tabella di verità di F è riportata in figura 1.

b) Poiché almeno una interpretazione rende falsa la proposizione F, essa non è una tautologia.

p	q	r	$p \vee q$	$\neg r$	$(p \vee q) \leftrightarrow \neg r$	$\neg p$	$q \wedge \neg p$	$\alpha \rightarrow \beta$
F	F	F	F	V	F	V	F	V
F	F	V	F	F	V	V	F	F
F	V	F	V	V	V	V	V	V
F	V	V	V	F	F	V	V	V
V	F	F	V	V	V	F	F	F
V	F	V	V	F	F	F	F	V
V	V	F	V	V	V	F	F	F
V	V	V	V	F	F	F	F	V
					α		β	

Figura 1: Tabella di verità della proposizione dell'esercizio 4a.

Esercizio 5

Formalizzare le seguenti proposizioni (ipotizzando che chi non beve, mangi, e viceversa):

a) se Antonio mangia, Bice e Carlo bevono;

b) Carlo oppure Bice bevono;

c) Carlo non mangia, Bice o Antonio sì;

d) Bice non mangia se e solo se Antonio e Carlo bevono.

e) Antonio beve solo se anche Bice fa lo stesso;

Soluzione

Dati i seguenti simboli proposizionali:

a	Antonio mangia
$\neg a$	Antonio beve
b	Bice mangia
$\neg b$	Bice beve
c	Carlo mangia
$\neg c$	Carlo beve

le frasi dell'esercizio possono essere formalizzate tramite le seguenti proposizioni:

a) $a \rightarrow (\neg b \wedge \neg c)$

b) $\neg c \vee \neg b$

c) $c \wedge (b \vee a)$

d) $\neg b \leftrightarrow (\neg a \wedge \neg c)$

e) $\neg a \rightarrow \neg b$

Esercizio 6

Dimostrare la validità delle seguenti inferenze:

a) **Ip1** $\neg a$
Ip2 $\neg c \vee (a \wedge b)$
Tesi $\neg c$

b) **Ip1** $a \leftrightarrow b$
Ip2 $\neg b \vee (\neg a \wedge c)$
Tesi $\neg b$

c) **Ip1** $\neg(a \vee b)$
Ip2 $a \vee (c \rightarrow \neg c)$
Tesi $a \leftrightarrow c$

Soluzione

a)

(1)	$\neg c \vee (a \wedge b)$	Ip2
(2)	$(\neg c \vee a) \wedge (\neg c \vee b)$	Distributività (1)
(3)	$\neg c \vee a$	Elim. congiunzione (2)
(4)	$c \rightarrow a$	Def. implicazione (3)
(5)	$\neg a$	Ip1
(6)	$\neg c$	Modus Tollens (4) e (5)

b)

(1)	$a \leftrightarrow b$	Ip1
(2)	$(a \rightarrow b) \wedge (b \rightarrow a)$	Def. biimplicazione (1)
(3)	$b \rightarrow a$	Elim. congiunzione (2)
(4)	$\neg a \rightarrow \neg b$	Contrapp. (3)
(5)	$\neg b \vee (\neg a \wedge c)$	Ip2
(6)	$(\neg b \vee \neg a) \wedge (\neg b \vee c)$	Distrib. (5)
(7)	$\neg b \vee \neg a$	Elim. congiunzione (6)
(8)	$\neg a \vee \neg b$	Associatività (7)
(9)	$a \rightarrow \neg b$	Def. implicazione (8)
(10)	$\neg b$	Dim. per casi (4) e (9)

c)

(1)	$\neg(a \vee b)$	Ip1
(2)	$\neg a \wedge \neg b$	Legge di De Morgan (1)
(3)	$\neg a$	Elim. congiunzione (2)
(4)	$a \rightarrow c$	Ex falso (3)
(5)	$a \vee (c \rightarrow \neg c)$	Ip2
(6)	$a \vee (\neg c \vee \neg c)$	Def. implicazione (5)
(7)	$a \vee \neg c$	Idempotenza (6)
(8)	$\neg a \rightarrow \neg c$	Def. implicazione (7)
(9)	$c \rightarrow a$	Contrapp. (8)
(10)	$a \leftrightarrow c$	Def. biimplicazione (4) e (9)