

Matricola □□□□□□

Esercizi su Numeri e Assembler – 7 Giugno 2017

Cognome □□□□□□□□□□□□□□□□ Nome □□□□□□□□□□□□□□□□

Esercizio 1

Data la rappresentazione $A = (11110001)_2$ di un intero in complemento a 2, trovarne il valore in base 10.

| |
|------------------|
| a = _____ |
|------------------|

Nello spazio sottostante riportare i passaggi più significativi della soluzione

Esercizio 2 Esercizio 4.2 Dire cosa stampa a video il seguente programma assembler, qualora vengano fornite in ingresso le due cifre meno significative del proprio numero di matricola.

```
_main:  CALL  input
        MOV   %AL,%BL
        CALL  input
        MOV   %AL,%CL
        OR    %BL,%AL
        NOT  %AL
        CALL  outbyte
        NOT  %BL
        NOT  %CL
        MOV  %BL,%AL
        AND  %CL,%AL
        CALL  outbyte
        RET

.INCLUDE "utility"
```

PROMEMORIA

CALL input attende da tastiera l'immissione di un carattere. Mette il relativo codice ASCII nel registro AL. Esempio: se si inserisce '0' in AL ci finisce l'esadecimale 0x30, se si inserisce '9' ci finisce l'esadecimale 0x39

OR *srg, dst* modifica *dst*, in modo tale da contenere l'OR bit a bit tra il precedente contenuto di *dst* e *srg*.

AND *srg, dst* modifica *dst*, in modo tale da contenere l'AND bit a bit tra il precedente contenuto di *dst* e *srg*.

CALL outbyte stampa a video la coppia di caratteri ASCII associati alla parte alta e bassa del naturale contenuto nei 4 bit più significativi e meno significativi di AL, rispettivamente. Esempio: qualora in AL vi fosse 0011-0101, stamperebbe a video i caratteri "35"

Uscita del programma assembler: _____

Nello spazio sottostante riportare i passaggi più significativi della soluzione

Soluzione

Soluzione Esercizio 1

Dopo aver notato che **A** inizia per 1 e che pertanto **a** non può che essere negativo, il suo valore assoluto può essere trovato facilmente calcolando il complemento ad uno di **A** e sommandoci uno:

$$\bar{A} = 00001110, \text{abs}(\mathbf{a}) = 00001110 + 1 = 00001111. \text{ Pertanto } \mathbf{a} = (-15)_{10}$$

Soluzione Esercizio 2

Il programmino assembler testa di validità della Legge di De Morgan fra **BL** e **CL**:

$$\text{NOT (BL OR CL) } \Leftrightarrow \text{NOT(BL) AND NOT(CL)}$$

Ad esempio, se il proprio numero di matricola termina per "35", in **BL** verrà caricato il codice ASCII di '3' (ossia 0x33) ed in **CL** quello di '5' (ossia 0x35).

Il loro OR logico darà:

$$\begin{array}{r} 0011-0011 (0x33) \text{ OR} \\ 0011-0101 (0x35) = \\ \hline 0011-0111 (0x37) \end{array}$$

La cui negazione logica è: 1100-1000 (0xC8)
Dunque a video viene stampato "C8" tramite la prima outbyte.

Nella seconda parte si negano prima **BL** e **CL**:

$$\begin{array}{l} \text{NOT(BL)} = 1100-1100 (0xCC) \\ \text{NOT(CL)} = 1100-1010 (0xCA) \end{array}$$

poi se ne effettua l'AND:

$$\begin{array}{r} 1100-1100 (0xCC) \text{ AND} \\ 1100-1010 (0xCA) = \\ \hline 1100-1000 (0xC8) \end{array}$$

Dunque verrà stampato ancora una volta "C8" dalla seconda outbyte. Riassumendo: "C8C8".

NB: Qualunque sia la coppia di cifre finali del proprio numero di matricola, le uscite delle due outbyte devono coincidere, in virtù della Legge di De Morgan.