

Esercizio 1

Il formato DICOM è un complesso formato utilizzato per le immagini mediche. Supporta moltissime opzioni e parametri e consente di far comunicare sistemi tra loro molto diversi in modo standard. Lo standard consente anche di memorizzare immagini in luce visibile, ovvero le immagini fotografiche classiche a cui siamo abituati.

Semplificando in modo estremo, un file DICOM può essere costruito con un header composto da 128 byte a 0 seguiti dai caratteri DICM, seguiti poi da una serie di *Data Element*. Un elemento (data element) è composto di:

- 1) Data Element Tag (suddiviso in Group Number e Element Number): due numeri a 16 bit senza segno. Identifica la semantica del dato.
- 2) Value Representation (VR): due caratteri ASCII. Identifica il tipo di dato.
- 3) Value Length (VL): 16 o 16+32 bit. È la lunghezza del dato in byte. **Deve sempre essere pari.**
- 4) Value Field: il dato vero e proprio. **Deve sempre essere composto da un numero pari di byte.**

Per quanto ci interessa consideriamo solo i seguenti VR:

US: unsigned short, VL è a 16 bit e vale 2: intero senza segno a 16 bit.

UL: unsigned long, VL è a 16 bit e vale 4: intero senza segno a 32 bit.

CS: code string, VL è a 16 bit ed è variabile: stringa breve.

UI: unique identifier (UID), VL è a 16 bit ed è variabile: è una stringa con un codice speciale registrato nello standard o dai produttori.

OB: byte string, VL è composto da 16 bit a 0, seguiti dalla lunghezza espressa come intero a 32 bit: una sequenza di byte.

Se la lunghezza di CS è dispari va aggiunto uno spazio, mentre se UI o OB hanno lunghezza dispari, va aggiunto un byte a 0 (0x00). **Tutti i dati sono in little endian.**

Lo standard specifica alcune centinaia di possibili tag, ma per immagini a colori semplici possiamo limitarci ad utilizzare i tag elencati di seguito:

Iniziamo con il gruppo 2, che descrive il file:

Campo	Tag (in hex)	Tipo	Descrizione
File Meta Information Group Length	0002,0000	UL	Il numero di byte di tutti gli elementi del gruppo 2 (escluso questo)
File Meta Information Version	0002,0001	OB	due byte: 00, 01
Media Storage SOP Class UID	0002,0002	UI	Usare "1.2.840.10008.5.1.4.1.1.77.1.4", che indica un'immagine fotografica.
Media Storage SOP Instance UID	0002,0003	UI	Dipende dal produttore, usare "1.2.392.200036.9125.0.19950720112207"
Transfer Syntax UID	0002,0010	UI	Usare "1.2.840.10008.1.2.1", che indica dati non compressi.
Implementation Class UID	0002,0012	UI	Dipende dal produttore, usare "1.2.392.200036.9125.0.1234567890"

Segue il gruppo 8, con info sul contenuto:

Campo	Tag (in hex)	Tipo	Descrizione
Image Type	0008,0008	CS	Usare "ORIGINAL\PRIMARY"
SOP Class UID	0008,0016	UI	Usare "1.2.840.10008.5.1.4.1.1.77.1.4".
SOP Instance UID	0008,0018	UI	Usare "1.2.392.200036.9125.0.19950720112207"

Poi il gruppo veramente fondamentale, con i dati dell'immagine:

Campo	Tag (in hex)	Tipo	Descrizione
Samples per Pixel	0028,0002	US	Numero di canali per pixel, usare 3
Photometric Interpretation	0028,0004	CS	"RGB"
Planar Configuration	0028,0006	US	Organizzazione dei dati (0=RGBRGB..., 1=RRR...,GGG...,BBB...). Usare 0.
Rows	0028,0010	US	Altezza dell'immagine
Columns	0028,0011	US	Larghezza dell'immagine
Bits Allocated	0028,0100	US	Bit per canale, usare 8
Bits Stored	0028,0101	US	Bit per canale, usare 8
High Bit	0028,0102	US	Bit più significativo, usare 7
Pixel Representation	0028,0103	US	Usare 0 (numeri senza segno)
Lossy Image Compression	0028,2110	CS	Compressione, usare "00", cioè nessuna compressione.

Da ultimo, bisogna mandare i pixel, quindi:

Campo	Tag (in hex)	Tipo	Descrizione
Pixel Data	7FE0,0010	OB	I byte dell'immagine in RGB.

Si scriva un programma a linea di comando che accetti le seguenti opzioni:

```
ppm2dcm <input file .PPM> <output file .DCM>
```

Il programma deve gestire la linea di comando, estrarre l'immagine PPM e salvarla in formato DICOM, utilizzando i campi indicati qui sopra. Per verificare che il salvataggio sia avvenuto correttamente, utilizzare il visualizzatore MicroDicom e i programmi dcdump.exe e dciodvfy.exe, forniti sul sito.