

Esercizio

Una software house sta realizzando una app per dispositivi mobili e deve rispondere ad una richiesta del client con una serie di tuple estratte da un DB, ritornate in formato JSON (il formato è descritto nell'allegato "rfc4627.txt", ma non serve per risolvere questo compito). Uno dei campi richiesti è però una piccola immagine a colori RGB e JSON non supporta la trasmissione di dati binari. Per questo motivo la software house decide di utilizzare un semplice encoding Base64 (descritto nel file "From RFC 1521.doc"). Per evitare di trasmettere header standard l'immagine viene compressa a piani separati con l'algoritmo Packbits e poi ogni piano codificato in Base64. Le informazioni vengono passate come campi JSON. Ad esempio:

```
{
  "width": 6,
  "height": 6,
  "red": "3f+A",
  "green": "/gD+//4A/v/+AP7//gD+//4A/v/+AP7/gA==",
  "blue": "3QCA"
}
```

r	r	r	g	g	g
r	r	r	g	g	g
r	r	r	g	g	g
r	r	r	g	g	g
r	r	r	g	g	g
r	r	r	g	g	g

L'immagine è metà rossa (255,0,0) e metà gialla (255,255,0) e diventa le sequenze:

R: FF,FF,FF,FF,FF,FF, FF,FF,FF,FF,FF,FF, FF,FF,FF,FF,FF,FF,
FF,FF,FF,FF,FF,FF, FF,FF,FF,FF,FF,FF, FF,FF,FF,FF,FF,FF

G: 00,00,00,FF,FF,FF, 00,00,00,FF,FF,FF, 00,00,00,FF,FF,FF,
00,00,00,FF,FF,FF, 00,00,00,FF,FF,FF, 00,00,00,FF,FF,FF

B: 00,00,00,00,00,00, 00,00,00,00,00,00, 00,00,00,00,00,00,
00,00,00,00,00,00, 00,00,00,00,00,00, 00,00,00,00,00,00

Dopo la codifica Packbits:

R: DD,FF,80

G: FE,00,FE,FF,FE,00,FE,FF,FE,00,FE,FF,FE,00,FE,FF,FE,00,FE,FF,80

B: DD,00,80

Dopo l'encoding in Base64:

R: 3f+A

G: /gD+//4A/v/+AP7//gD+//4A/v/+AP7/gA==

B: 3QCA

Si realizzi un programma che esegua la seguente sequenza di operazioni:

1) Caricare una immagine a colori nel formato PPM, seguendo le specifiche allegate ("PPM Format Specification.htm"]. A titolo di esempio vengono forniti i file "test.ppm" e "facolta.ppm"

2) Dividere la matrice di valori RGB in tre matrici ognuna contenente un canale. Salvare poi 3 file binari (r.bin, g.bin, b.bin) contenenti i rispettivi valori per righe, rappresentati ognuno con 1 byte.

3) Applicare l'algoritmo Packbits ad ogni matrice R, G e B rispettivamente. Salvare poi 3 file binari (r.pkb, g.pkb, b.pkb) contenenti i rispettivi valori compressi.

4) Applicare ad ogni sequenza l'encoding in Base64. Salvare tre file di testo (r.txt, g.txt, b.txt) contenenti le sequenze codificate. Porre attenzione al fatto che ogni linea non può contenere più di 76 caratteri.

5) Produrre un file di testo (json.txt) con l'output JSON come indicato all'inizio.