


PackBits

 Submit Solution

Nel file `compress.cpp` implementare la procedura corrispondente alla seguente dichiarazione:

```
1. void PackBitsEncode(const mat<uint8_t>& img, std::vector<uint8_t>& encoded);
```

che applica l'algoritmo PackBits ad un piano colore, `img`, salvando il risultato in un vettore di `uint8_t`.

Come visto a lezione, PackBits è uno schema di compressione veloce e semplice senza perdita di dati per la codifica dei dati in run-length.

Il flusso di dati PackBits è costituito da pacchetti con un'intestazione da un byte seguita da uno o più byte di dati compressi.

Nella tabella seguente è riportata l'associazione tra il valore del byte di intestazione, `L`, e i byte dei dati.

header (L)	byte di dati che seguono l'header
da 0 a 127	(L+1) byte letterali di dati (copia)
da 129 a 255	un byte di dati, ripetuto (257-L) volte nell'output decompresso (run)
128	End of data marker (EOD)

L'algoritmo deve scorrere l'input e per ogni sequenza di byte uguali emettere una run, altrimenti una copia per le sequenze prive di ripetizioni e al termine dei dati un EOD.