3 MIPS projects

Each project should be delivered as an *.asm file and a set of related test cases. The test cases should confirm that the program works correctly. Each project should contain a directory named *tests*. The sub directories of the directory *tests* correspond to individual tests. They should contain:

- input file(s),
- output file(s),
- a text file (named *description.txt*) containing short description what was being tested and the values of the input parameters (if any).

3.21 Food processor

The input BMP image [1] (24-bit RGB) contains one of three food types from the given set (Table 1). Your task is to recognize the food and print in the console window its three-character code given in Table 1.

The recognition should be performed in two three steps:

1. Preparation of the histogram of given color component (table 1). The histogram contains number of pixels for each value of given color component.
2. Based on the histogram the mode[4] should be calculated. Mode of a set of data values is the value that appears most often.
3. Mode ranges of images belonging to different food types are different. So, you can make decision which food type the image should be assigned to.

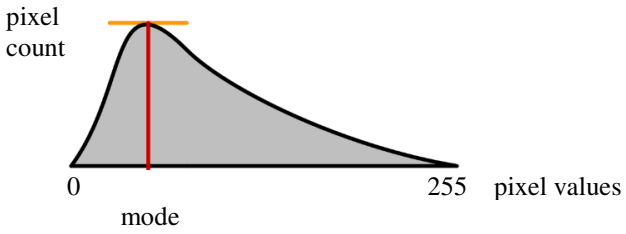


Fig 1. Histogram and mode [4]

Input

- BMP file containing the source image:
  - sub format: 24 bit RGB – no compression,
  - size: width 200px, height up to 200 px,
  - file name: "source.bmp"

Output

- Console window – plain text

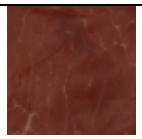
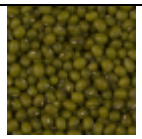
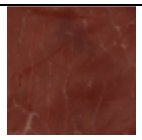
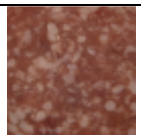
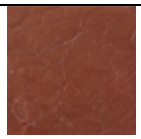
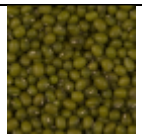
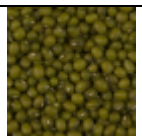
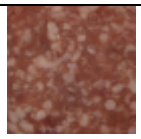
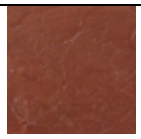
Remarks:

1. Please pay attention to the efficiency of the program (getPixel function is not very efficient)
2. Check the input data and signal errors (e.g. wrong file format)

References:

[1] "**file-format-bmp**", https://en.wikipedia.org/wiki/BMP_file_format
[2] Example program for bmp reading/writing,
http://galera.ii.pw.edu.pl/~zsz/ecoar/bmp/bmp_mips.zip
[3] RawFooT DB: Raw Food Texture Database,
http://www.ivl.disco.unimib.it/minisites/rawfoot/textures.php
[4] Mode (statistics), https://en.wikipedia.org/wiki/Mode_(statistics)
[5] Hexadecimal file editor, https://hexed.it/

Table 1. Data set description - example images

Set no.	Food #1	Food #2	Food #3	Recognition based on the histogram of
1	 acb	 adz	 cur	Red component
2	 acb	 len	 sal	Red component
3	 oat	 pep	 qui	Red component
4	 bre	 car	 lin	Red component
5	 cor	 spe	 ste	Blue component
6	 adz	 pea	 qui	Green component
7	 len	 lin	 oat	Blue component
8	 adz	 chi	 pep	Green component
9	 car	 len	 pep	Green component
10	 bre	 cor	 spe	Blue component
11	 cur	 sal	 ste	Green component
12	 lin	 pea	 qui	Blue component

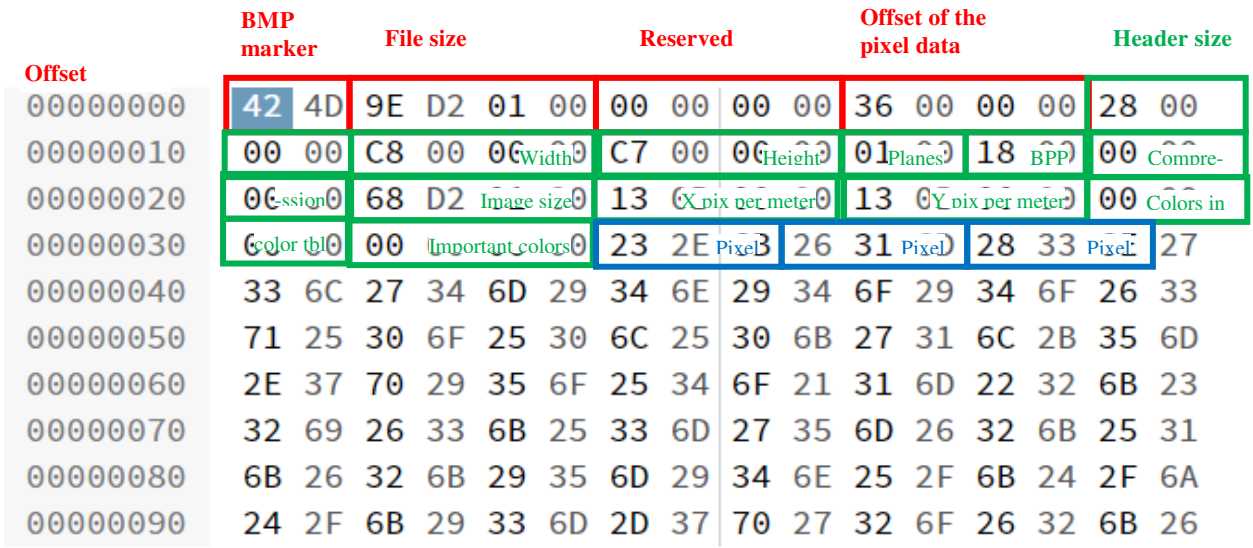
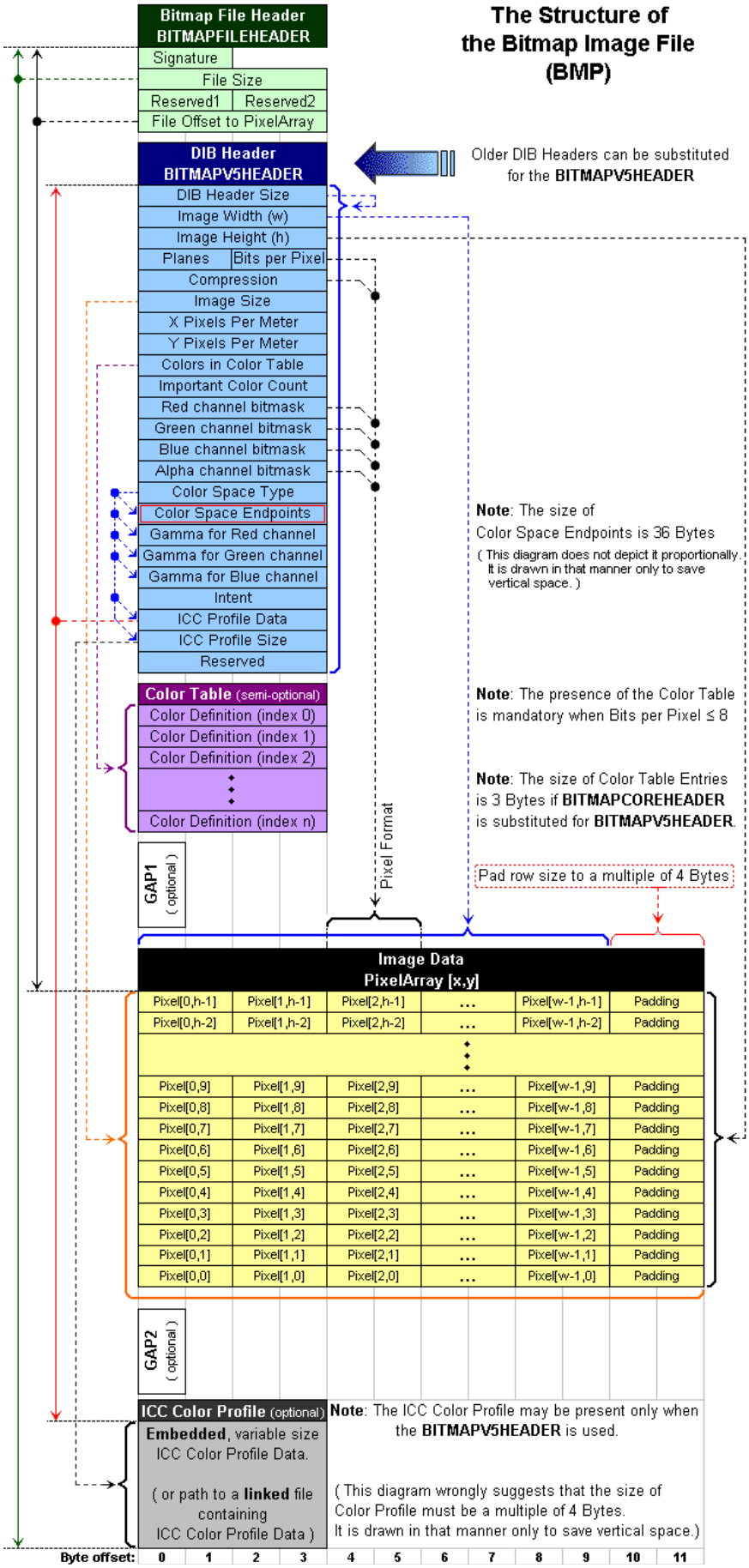


Figure A1. Contents of an example BMP file (all numbers in hexadecimal notation).

Remarks:

1. Little endian byte order – the first byte in a field is the least significant one
2. Order of color components of a pixel: first byte - blue, second byte - green, third byte -red.


The Structure of the Bitmap Image File (BMP)
Bitmap File Header
BITMAPFILEHEADER
Signature
File Size
Reserved1
Reserved2
File Offset to PixelArray
DIB Header
BITMAPV5HEADER
DIB Header Size
Image Width (w)
Image Height (h)
Planes
Bits per Pixel
Compression
Image Size
X Pixels Per Meter
Y Pixels Per Meter
Colors in Color Table
Important Color Count
Red channel bitmask
Green channel bitmask
Blue channel bitmask
Alpha channel bitmask
Color Space Type
Color Space Endpoints
Gamma for Red channel
Gamma for Green channel
Gamma for Blue channel
Intent
ICC Profile Data
ICC Profile Size
Reserved
Older DIB Headers can be substituted for the BITMAPV5HEADER
Note: The size of Color Space Endpoints is 36 Bytes
(This diagram does not depict it proportionally. It is drawn in that manner only to save vertical space.)
Color Table (semi-optional)
Color Definition (index 0)
Color Definition (index 1)
Color Definition (index 2)
Color Definition (index n)
Note: The presence of the Color Table is mandatory when Bits per Pixel ≤ 8
Note: The size of Color Table Entries is 3 Bytes if BITMAPCOREHEADER is substituted for BITMAPV5HEADER.
GAP1 (optional)
Pixel Format
Pad row size to a multiple of 4 Bytes
Image Data
PixelArray [x,y]
Pixel[0,h-1] Pixel[1,h-1] Pixel[2,h-1] ... Pixel[w-1,h-1] Padding
Pixel[0,h-2] Pixel[1,h-2] Pixel[2,h-2] ... Pixel[w-1,h-2] Padding
Pixel[0,9] Pixel[1,9] Pixel[2,9] ... Pixel[w-1,9] Padding
Pixel[0,8] Pixel[1,8] Pixel[2,8] ... Pixel[w-1,8] Padding
Pixel[0,7] Pixel[1,7] Pixel[2,7] ... Pixel[w-1,7] Padding
Pixel[0,6] Pixel[1,6] Pixel[2,6] ... Pixel[w-1,6] Padding
Pixel[0,5] Pixel[1,5] Pixel[2,5] ... Pixel[w-1,5] Padding
Pixel[0,4] Pixel[1,4] Pixel[2,4] ... Pixel[w-1,4] Padding
Pixel[0,3] Pixel[1,3] Pixel[2,3] ... Pixel[w-1,3] Padding
Pixel[0,2] Pixel[1,2] Pixel[2,2] ... Pixel[w-1,2] Padding
Pixel[0,1] Pixel[1,1] Pixel[2,1] ... Pixel[w-1,1] Padding
Pixel[0,0] Pixel[1,0] Pixel[2,0] ... Pixel[w-1,0] Padding
GAP2 (optional)
ICC Color Profile (optional)
Embedded, variable size ICC Color Profile Data.
(or path to a linked file containing ICC Color Profile Data)
Note: The ICC Color Profile may be present only when the BITMAPV5HEADER is used.
(This diagram wrongly suggests that the size of Color Profile must be a multiple of 4 Bytes.
It is drawn in that manner only to save vertical space.)
Byte offset: 0 1 2 3 4 5 6 7 8 9 10 11