Bitmap Header Reader

Goals:

1. Experience using C++ struct's and methods.
2. Experience reading from binary files.

Tasks:

1. Create Three data types (BitmapFileHeader, BitmapImageHeader, and Image)
    a. The attributes of each of the header types should be the same as the file and info attributes listed at http://en.wikipedia.org/wiki/BMP_file_format.
    b. I will help you in class decide what to make the data type of each header attribute.
    c. Add display methods for each of the header types. These methods should display all of the attributes of the type in an organized manner.
    d. The Image type should have a BitmapFileHeader and a BitmapImageHeader.
2. You are to write a c++ program that will:
    a. Create a variable of type Image named image.
    b. Ask the user for the name of a bit map file. (i.e. junk.bmp)
    c. Open the file for reading in "binary" mode.
    d. Read the attributes from the file headers and store them in to the image's header attributes.
    e. Call the display methods on the headers.
    f. Close the file.

The following tables are from http://en.wikipedia.org/wiki/BMP_file_format

Offset#	Size	Purpose
0000$_h$	2 bytes	the header field used to identify the BMP & DIB file is 0x42 0x4D in hexadecimal, same as BM in ASCII. The following entries are possible: ▪ **BM** – Windows 3.1x, 95, NT, ... etc. ▪ **BA** – OS/2 struct Bitmap Array ▪ **CI** – OS/2 struct Color Icon ▪ **CP** – OS/2 const Color Pointer ▪ **IC** – OS/2 struct Icon ▪ **PT** – OS/2 Pointer
0002$_h$	4	the size of the BMP file in bytes

	bytes	
0006_h	2 bytes	reserved; actual value depends on the application that creates the image
0008_h	2 bytes	reserved; actual value depends on the application that creates the image
$000A_h$	4 bytes	the offset, i.e. starting address, of the byte where the bitmap image data (pixel array) can be found.

Offset #	Size	Purpose
0Eh	4	the size of this header (40 bytes)
12h	4	the bitmap width in pixels (signed integer).
16h	4	the bitmap height in pixels (signed integer).
1Ah	2	the number of color planes being used. Must be set to 1.
1Ch	2	the number of bits per pixel, which is the color depth of the image. Typical values are 1, 4, 8, 16, 24 and 32.
1Eh	4	the compression method being used. See the next table for a list of possible values.
22h	4	the image size. This is the size of the raw bitmap data (see below), and should not be confused with the file size.
26h	4	the horizontal resolution of the image. (pixel per meter, signed integer)

2Ah	4	the vertical resolution of the image. (pixel per meter, signed integer)
2Eh	4	the number of colors in the color palette, or 0 to default to 2^n.
32h	4	the number of important colors used, or 0 when every color is important; generally ignored.