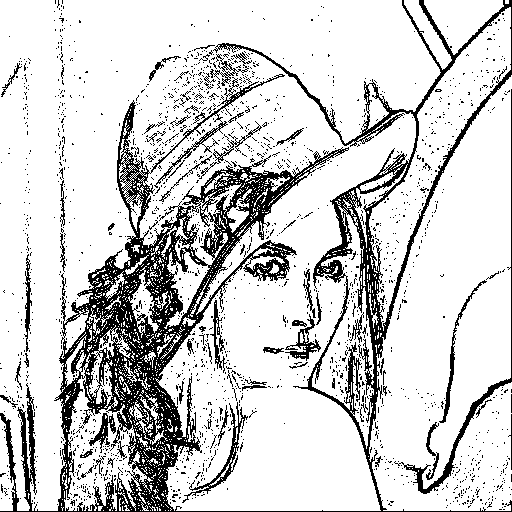
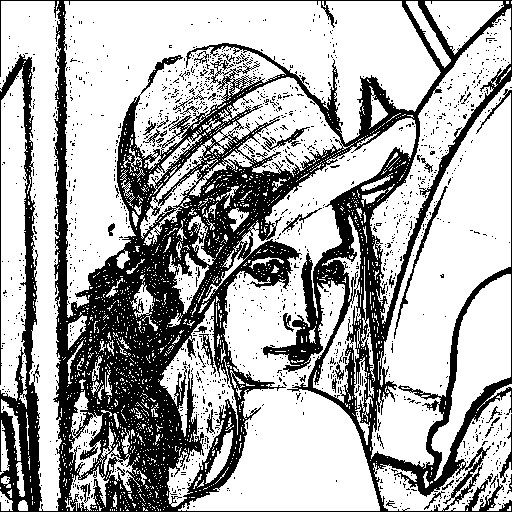
# CV Homework 9

R02922124 葉信良  
$ g++ -o hw9 hw9.cpp

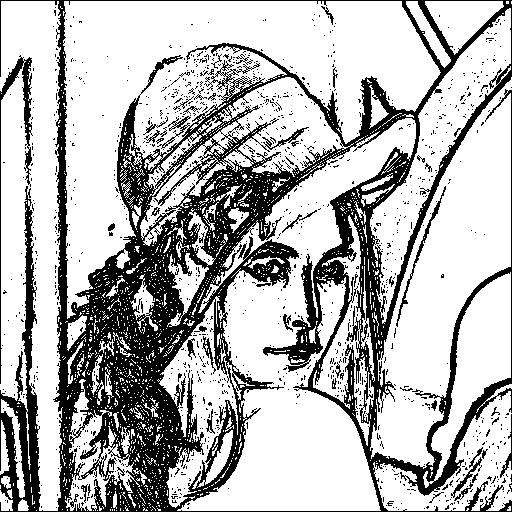
Part 1 Roberts Operator

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution算出Gx跟Gy，再將它們平方相加再開更號得到G，如果G超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我輸出了threhold 5~20的圖，覺得threhold=16時效果是比較好的。  
$ ./hw9 roberts 16 圖片檔名是Roberts\_16.bmp  


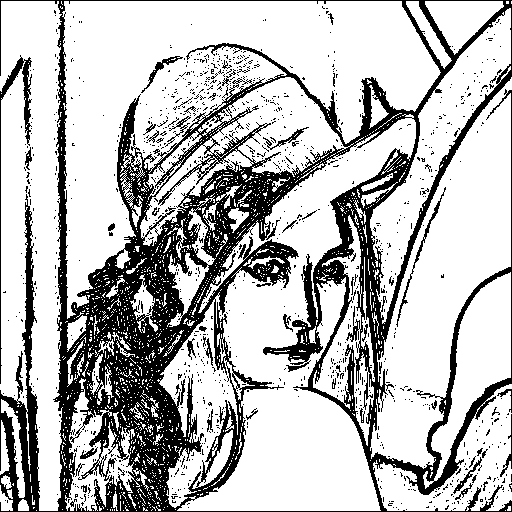
Part 2 Prewitt

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution算出p1跟p2，再將它們平方相加再開更號得到g，如果g超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我覺得threhold=30時效果是比較好的。  
$ ./hw9 prewitt 30 圖片檔名是Prewitt\_30.bmp  


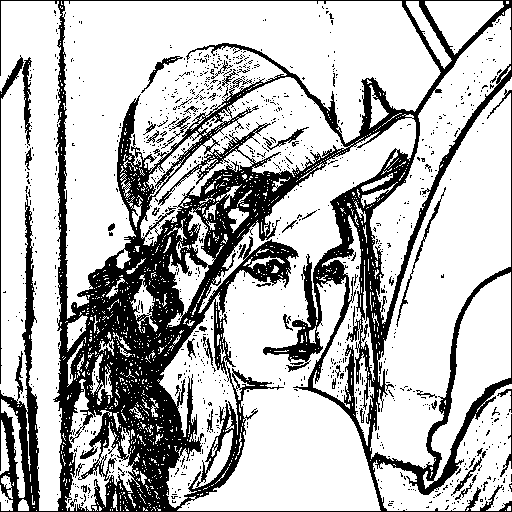
Part 3 Sobel

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution算出s1跟s2，再將它們平方相加再開更號得到g，如果g超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我覺得threhold=47時效果是比較好的。  
$ ./hw9 sobel 47 圖片檔名是Sobel\_47.bmp  


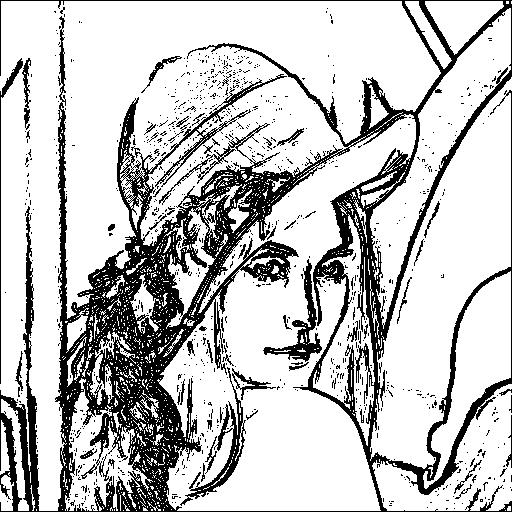
Part 4 Frei and Chen

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution算出f1跟f2，再將它們平方相加再開更號得到g，如果g超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我覺得threhold=43時效果是比較好的。  
$ ./hw9 freichen 43 圖片檔名是FreiChen\_43.bmp  


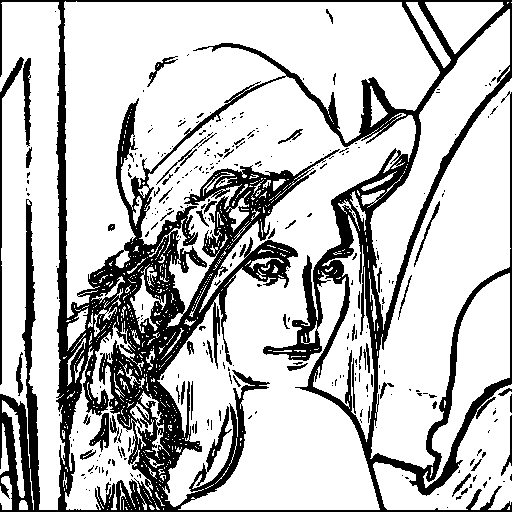
Part 5 Kirsch

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution來算出做八個3x3矩陣convolution的值，再將它們平方相加再開更號得到g，如果g超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我覺得threhold=300時效果是比較好的。  
$ ./hw9 kirsch 30 圖片檔名是Kirsch\_300.bmp  


Part 6 Robinson

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution來算出做八個3x3矩陣convolution的值，再將它們平方相加再開更號得到g，如果g超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我覺得threhold=120時效果是比較好的。  
$ ./hw9 robinson 120 圖片檔名是Robinson\_120.bmp  


Part 7 Netatia and Babu

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元讀出後直接輸出到output。接著將image的數值0~255存到unsigned char array BBMap中。   
然後依照投影片中的矩陣，對每一個pixel做convolution來算出做六個5x5矩陣convolution的值，再將它們平方相加再開更號得到g，如果g超過threhold的話就判斷是邊緣，設成0，其他則設成255。最後輸出OMap。  
而我覺得threhold=30000時效果是比較好的。  
$ ./hw9 n-babu 30000 圖片檔名是Netatia&Babu\_30000.bmp  


🡪以上的source code、執行檔、result image皆附在壓縮檔中。