COMP 3731: Computer Forensics
Department of Computer Science
University of Denver

ASSIGNMENT 1

This is a warm up exercise in reading binary files and interpreting values using C/C++.

Description

For this assignment, you will create a C/C++ program called **ByteRead** that has at least two functions – getNumeric and binaryDumpAsASCII. A template for the file is at the end of this document.

getNumeric

```
unsigned long getNumeric (FILE *f,
                          long offset,
                          int size,
                          char type,
                          char endian) {
      …
}
```

The getNumeric function should read size number of bytes starting at byte number offset (zero based indexing) using the FILE pointer f. Then it returns a number (unsigned long value) corresponding to the byte sequence. The value of size will be 1, 2 or 4. When computing the return value, two conditions must be accounted.

1. If endian is LITTLE then byte sequences are stored in reverse order in the file. For example, the number 0x12AB34CD is actually stored as 0xCD 0x34 0xAB 0x12 in the file. If endian is BIG then byte sequences are stored using the most significant to least significant order.
2. If type is SIGNED then the byte sequence represents a number stored in 2's complement format. Otherwise, when the type is UNSIGNED, the byte sequence corresponds to the absolute value of a number.

Most hex editors will allow you to select a sequence of bytes, and view the corresponding decimal representation when the bytes are treated as a signed/unsigned number stored in little/big endian format. Use it to test the correctness of your implementation. The following are few sample outputs when getNumeric is called on the file file.bin available in the assignment page. The file has the byte sequence 0x12 0xAB 0x34 0xCD (view it using a hex editor).

```
getNumeric(…, 0, 4, SIGNED, LITTLE)           -852186350
getNumeric(…, 0, 4, SIGNED, BIG)              313210061
getNumeric(…, 1, 2, SIGNED, LITTLE)           13483
getNumeric(…, 1, 2, SIGNED, BIG)              -21708
```

binaryDumpAsASCII

```
void binaryDumpAsASCII(FILE *fin) {
      …
}
```

The binaryDumpAsASCII function takes as input a FILE pointer. It then reads the file and prints the hexadecimal representation of the bytes and their ASCII equivalent in a formatted manner. The formatting should be as follows.

- Each line will contain 16 byte values (except the last one if there are not enough bytes remaining) and 16 character representations of those bytes if printable.
- Each byte will be printed with a trailing whitespace.
- After printing the bytes (and the trailing spaces), there should be a tab space. If you have less than 16 bytes to print for the last line, then an appropriate number of whitespaces must be added before adding the tab space (see sample output below).
- Following on, the ASCII representations of printable bytes are printed (without any spaces in between). If a byte's value is less than 32 or greater than 126, then print a whitespace; otherwise print the ASCII representation.

The following shows a sample output of a small PNG file (available on the assignment page).

```
89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52    PNG        IHDR
00 00 00 18 00 00 00 18 08 06 00 00 00 E0 77 3D                  w=
F8 00 00 00 04 73 42 49 54 08 08 08 08 7C 08 64        sBIT    | d
88 00 00 00 09 70 48 59 73 00 00 0B 12 00 00 0B        pHYs
12 01 D2 DD 7E FC 00 00 00 16 74 45 58 74 43 72       ~     tEXtCr
65 61 74 69 6F 6E 20 54 69 6D 65 00 30 33 2F 30   eation Time 03/0
31 2F 31 36 67 F1 0E EC 00 00 00 1C 74 45 58 74   1/16g       tEXt
53 6F 66 74 77 61 72 65 00 41 64 6F 62 65 20 46   Software Adobe F
69 72 65 77 6F 72 6B 73 20 43 53 36 E8 BC B2 8C   ireworks CS6
00 00 03 DF 49 44 41 54 48 89 AD 96 5B 6C 54 55       IDATH   [lTU
14 86 BF 7D 2E 33 9D 99 76 4A EF D3 0B ED B4 B4      }.3  vJ
2A 32 12 2F 90 C6 C4 60 2C 9A 4A 4C 1F D0 20 06   *2 /   `, JL
62 6A A2 46 23 81 04 D3 28 06 4C C5 84 A8 51 13   bj F#   ( L   Q
35 26 EA 03 24 5A 44 90 78 29 D2 F8 80 D1 E0 8B   5&  $ZD x)
97 22 28 AD 02 6D 89 15 7B 23 9D 4E 3B 6D E7 D2    "(  m  {# N;m
33 73 F6 F6 A1 52 9D 32 53 B1 E9 7A 39 FB AC 9D   3s   R 2S  z9
FC DF 59 FF 5E 2B FB 88 AE 80 FF 29 E0 0D C0 64     Y ^+     )   d
69 23 0C EC 16 5D 01 7F 3F 50 B5 C4 E2 57 62 52   i#   ]  ?P   WbR
03 BC F3 B3 9A C7 43 ED D1 0E 0A B6 34 A3 E7 78         C     4  x
B9 E1 DB 9F F0 DE 7D EF 62 00 5E 03 90 F3 B3 32         } b ^    2
12 21 DE 7B 81 AC 1B 03 B8 07 FE 44 18 06 D1 B3    ! {       D
67 16 03 40 CB B4 31 7A E0 5D B2 6A AF C7 F7 F4   g  @  1z ] j
2E 42 87 DB 50 B1 18 65 BB 5F 24 EF FE CD 08 C3   .B  P  e _$
20 FB F6 3B 30 7D A5 78 D6 D4 A3 7B 73 FF 3F 60      ;0} x   {s ?`
A6 AF 07 6B E0 12 8E EA 1A 42 47 0F 51 BC 6D 27      k     BG Q m'
59 2B 57 91 FF C0 43 E4 AC 6B A0 E8 89 ED 78 6E   Y+W   C  k    xn
AB A7 E8 C9 ED 38 2A 33 1F 61 46 00 40 F4 74 27        8*3 aF @ t'
56 FF EF D8 93 61 9C D5 2B B8 FC D6 6B C4 7B CE   V    a  +   k {
E3 AC BB 0E 7B 3C 84 92 36 2A 3E 03 4A 2D 0E 80       {<  6*> J-
6E 00 20 84 86 4A 24 10 0E 07 08 01 52 A1 A4 04   n    J$      R
DB 46 D9 49 C8 AC BF 30 40 68 1A E8 FA EC DA E1    F I   0@h
40 59 16 4A 49 10 02 61 9A A8 64 12 21 34 10 99   @Y JI  a  d !4
35 8C 85 00 CA 4E 22 74 1D 25 6D 62 5D 3F 53 FE   5    N"t %mb]?S
C2 4B D8 53 93 84 0E BD 4F D6 8A 3A 4A 76 B4 A0    K S    O  :Jv
6C 1B 39 3D 9D F9 23 BB 02 FE 20 50 90 96 5E 58   l 9=  #    P  ^X
84 D3 5F 43 E4 D4 0F 00 E4 6E 68 62 A6 AF 87 78     _C     nhb   x
EF 05 8C FC 02 CC 8A E5 D8 C1 20 D6 D0 C0 B5 03
F4 DC 65 E4 6D DC 84 59 56 81 8C 46 90 B1 28 7A     e m  YV  F  (z
B6 17 65 27 91 53 53 08 A7 13 CD ED 46 C6 E3 28     e' SS     F  (
CB 42 18 06 7A 8E 17 6B 68 90 89 63 9F 60 87 27    B  z  kh  c ` '
32 5B 64 96 95 53 B1 EF 75 3C 6B EA 17 72 2E 63   2[d  S  u<k  r.c
78 EF 6C 60 60 4F 0B 89 91 E1 F4 15 D4 B5 9F 40   x l``O         @
CE C4 19 7D EF 6D 8C 82 42 12 23 C3 E4 6F DE 4A      } m  B #  o J
CE BA 06 00 92 63 41 42 47 0E B2 AC 69 23 8E E5        cABG   i#
E9 7B 3F 76 AE 9B 8B 0F 36 CD BD A7 74 91 B3 A6    {?v    6   t
16 CD ED 41 38 9D 84 3E FE 10 D3 57 8A 59 54 32      A8  >   W YT2
B7 9F 1C 0B 12 6C DB 4F FC DC 6F 73 39 65 59 58        l O  os9eYX
03 97 E6 66 C1 B5 32 90 02 BC AA 8B 9C 55 D5 14      f  2      U
3E FC 28 91 1F BF A3 78 DB 4E C6 3F 3D 42 FC 62   > (    x N ?=B b
2F 89 CB C3 98 C5 3E 9C 95 D5 04 0F 1E 60 F2 E4   /     >      `
57 94 EE 6A 65 E4 D5 7D 44 CF 9C C2 BB BE 91 92   W  je  }D
1D 2D A0 A5 76 7E DA 39 B0 A7 A7 40 08 92 A1 31    -  v~ 9   @   1
A6 4E 7E CD 4C 5F 0F 91 CE EF B1 86 06 D0 DC 6E    N~ L_         n
0A 1F 79 1C A4 24 74 B8 8D C4 D0 20 B9 8D F7 31     y  $t        1
D1 F1 39 76 38 7C 95 D6 82 73 20 84 40 CB CE 46     9v8|   s  @  F
B8 5C 68 1E 0F C2 30 40 29 8C FC 02 B4 EC 1C 50    \h   0@)      P
0A 6B 64 10 EB 8B 3F 70 DF BA 16 CD ED BE 46 80    kd   ?p      F
52 20 25 4A 29 48 24 D1 9C 59 B3 79 29 E7 26 19   R %J)H$  Y y) &
21 90 91 08 A6 AF 8C C2 E6 C7 90 D1 68 5A A9 B4   !           hZ
16 A9 78 1C 39 3D 8D D0 34 94 9D 64 E2 F8 67 B3     x 9=  4  d  g
07 AB 14 F6 F8 38 A3 FB DF 61 A6 E7 3C AE D5 37        8   a  <  7
23 84 20 7C BC 9D D0 47 1F A0 92 C9 85 2B 88 75   #  |   G     + u
9F C5 15 58 8D A3 CA 4F FE D6 66 CC 92 52 F2 37      X   O  f  R 7
6D 21 72 BA 13 47 45 25 AE 55 37 61 FA CA 88 75   m!r  GE% U7a   u
FF 82 79 CB 5A BC EB 1B 91 B1 18 E1 8E 76 7C CF     y Z         v|
EC 41 F3 78 88 FD DA 95 6A 73 57 C0 1F 02 F2 00    A x    jsW
F4 BC 3C CA F7 BE 82 F7 AE 7B D2 96 FB 5F 31 F9     <       {   _1
CD 09 06 5B 9F C5 1E 1F 4F 01 F4 F3 AF 4B 5F CB      [    O    K_
72 51 BE F7 65 72 37 34 CD 76 13 CC 7A AE D4 3F   rQ  er74 v  z  ?
CF 2B 39 DB 06 5D 47 F7 64 33 F1 E5 31 86 5A 9F    +9  ]G d3  1 Z
43 C6 63 29 4C F1 F7 6F CB 9B F3 ED 5A 82 98 00   C c)L  o    Z
9E FF 0B 79 D1 82 63 E5 82 D9 DE 00 00 00 00 49      y  c        I
45 4E 44 AE 42 60 82                              END B`
```

Resource

You will need file I/O functions such as `fseek` and `fread` to implement these functions. The Open Group Base Specifications page at *http://pubs.opengroup.org/onlinepubs/9699919799/* is a good resource to look up their definitions (and others in general).

Submission

You **must** fully comment your program. Put comments in the beginning of the program specifying your name and the purpose of the program. Submit a single source file **ByteRead.c** (or **.cpp)** file in Canvas.

Grading

The assignment is worth **50 points** (clean, commented code: 5 pts., `getNumeric`: 25 pts., `binaryDumpAsASCII`: 20 pts.) A program that does not compile is a program that you did not submit at all. Remember the GTA is not required to debug your program to give you partial points.

The late policy is available in the course syllabus. **You must work alone on this assignment.**

```
#include <stdio.h>
#include <stdlib.h>

#define LITTLE  0
#define BIG 1
#define SIGNED 0
#define UNSIGNED 1

unsigned long getNumeric (FILE *f, long offset, int size, char type, char
endian) {

    ...

}

void binaryDumpAsASCII(FILE *fin) {

    ...

}


int main() {
    FILE *fin;
    FILE *fin2;

    fin = fopen("file.bin", "r");
    printf("%ld\n",getNumeric(fin,0,4,SIGNED,LITTLE));
    fclose(fin);

    fin2 = fopen("icon.png","r");
    binaryDumpAsASCII(fin2);
    fclose(fin2);

    return 0;

}
```