Assignment 3

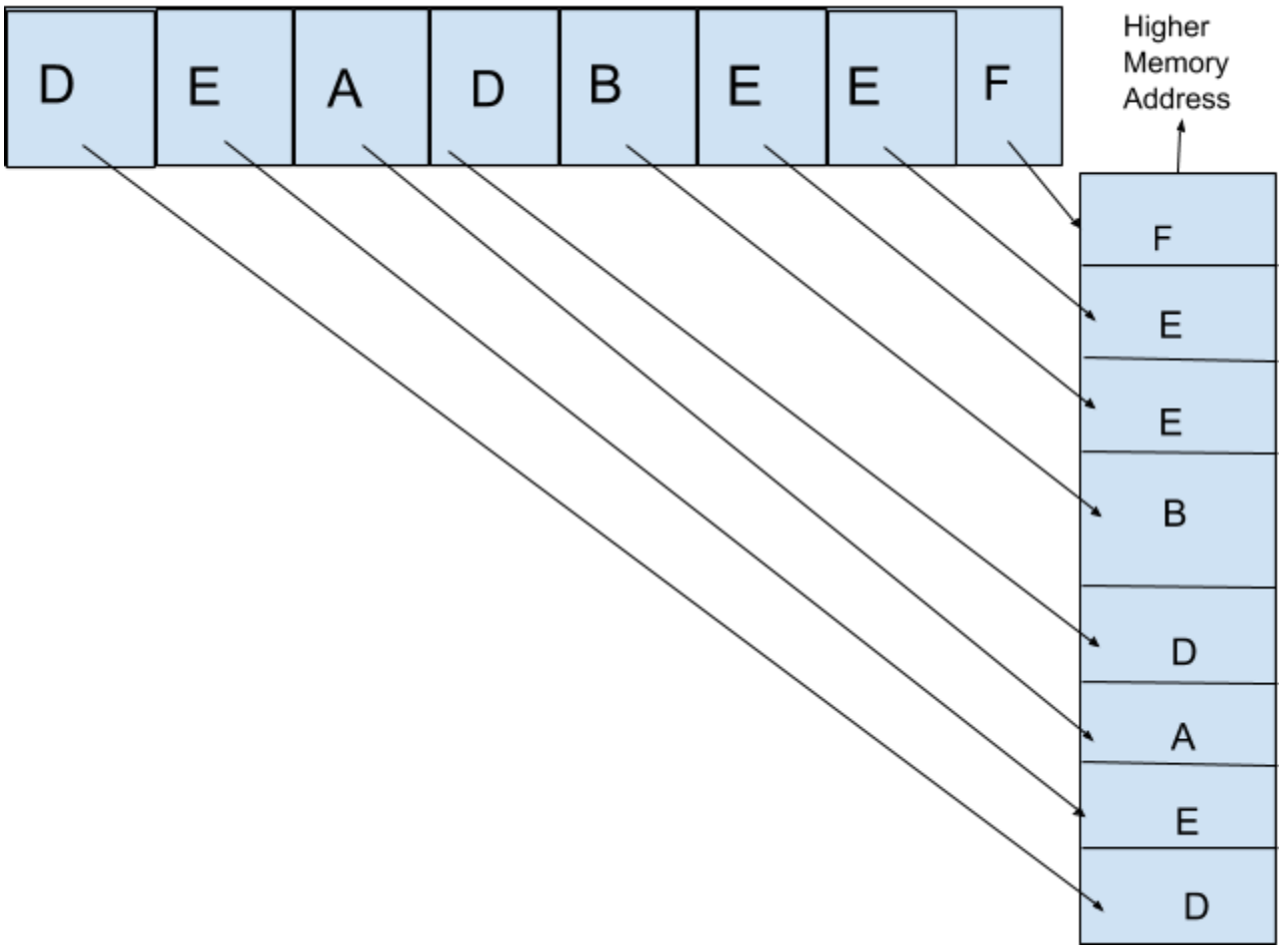
Aditya Menon
(AM.EN.P2CSN18001)

1) What is the difference between text file and a binary file?
    a) In a text file, the data is stored in the form of alphanumeric characters and other special symbols as ASCII values in human readable format. On the other hand, a binary file stores a sequence of bits and bytes which are not human readable.
    b) Error in text files can easily be detected and eliminated. A single error in a binary file can corrupt the whole file.
    c) 'newline' character are converted to carriage return-linefeed combination and then written to the disk. When read in a text file, the carriage return-linefeed combination is converted to newline. When read as binary file, this conversion does not take place.
    d) In text mode the end-of-file(EOF) is represented by a ASCII code 26 which is inserted at the end of the file. In binary mode, end of file is derived from the directory entry of the file which stores the number of characters present in the file.
2) What is the ELF FILE Format? What is a hexdump?
    a) Extensible Linking Format(ELF) is a standard file format for executables, object code, shared libraries and core dumps. ELF format is flexible, extensible, and cross-platform. ElF files constitutes a file header and the file data. The data might include Program header table, Section header table and the data referred to by them.
    b) Hexdump is the hexadecimal representation of computer data present in the RAM or a file or a storage device. In a hex dump, eachbyte is represented as a 2-digit hexadecimal number. They are represented in rows of 8 or 16 bytes, separated by whitespaces. Some hex dumps have the hexadecimal memory address and/or a checksum byte at the end of each line.
3) What kind of information is stored in a hexdump?
    a) The various kinds of information stored in hexdump

- a file read from a disk device
- raw bytes read from a disk device (displaying the partitioning or filesystem structures)
- something you captured from a serial line
- something you captured from a network device (IP packets, Ethernet frames)
- a memory dump
- data captured from some communications bus (USB, I2C etc.)

4) Diagrammatically represent how the hex characters "DEADBEEF" will be represented in Big Endian and Little Endian formats.

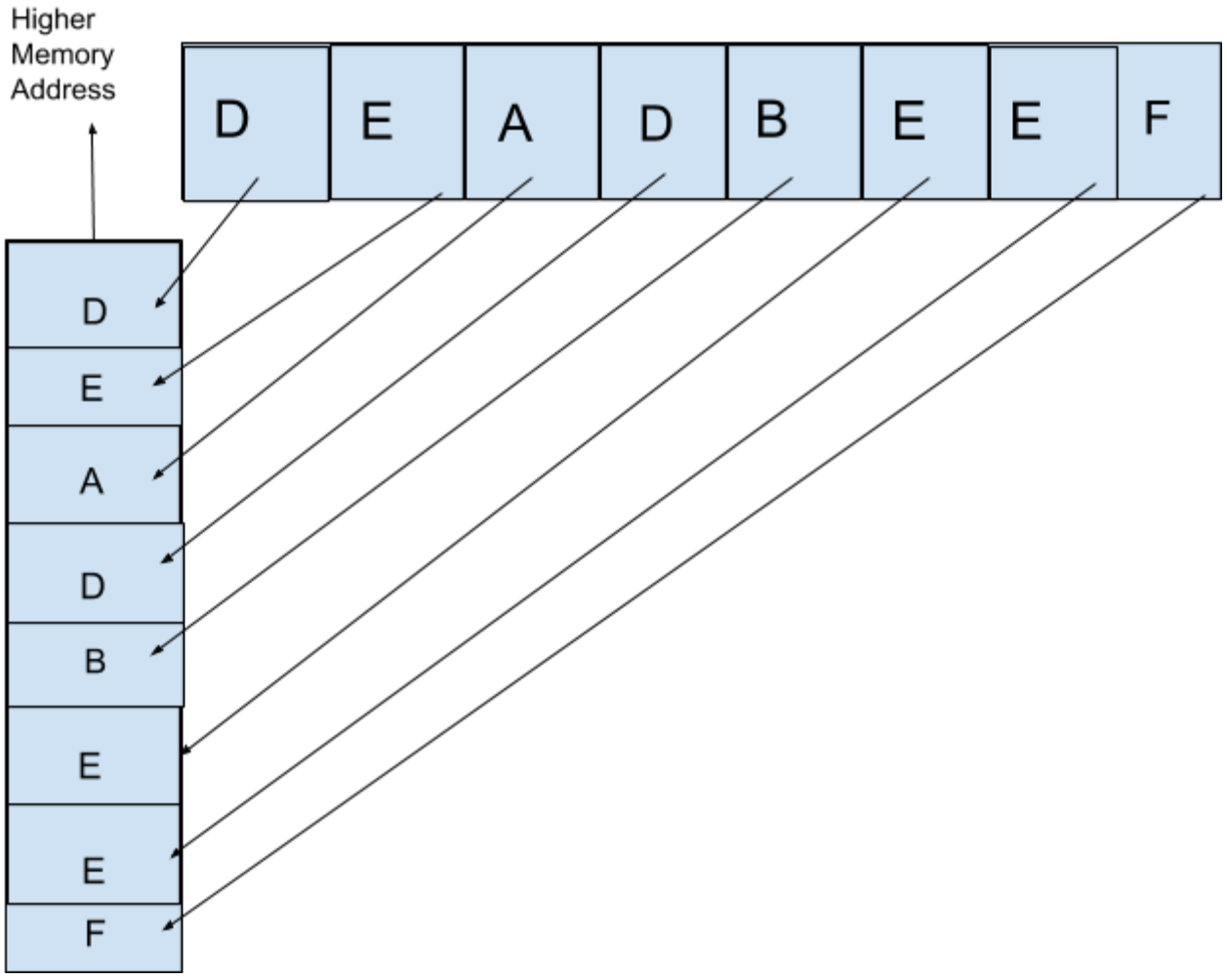
a) Big Endian



b)

c) Little Endian

d)

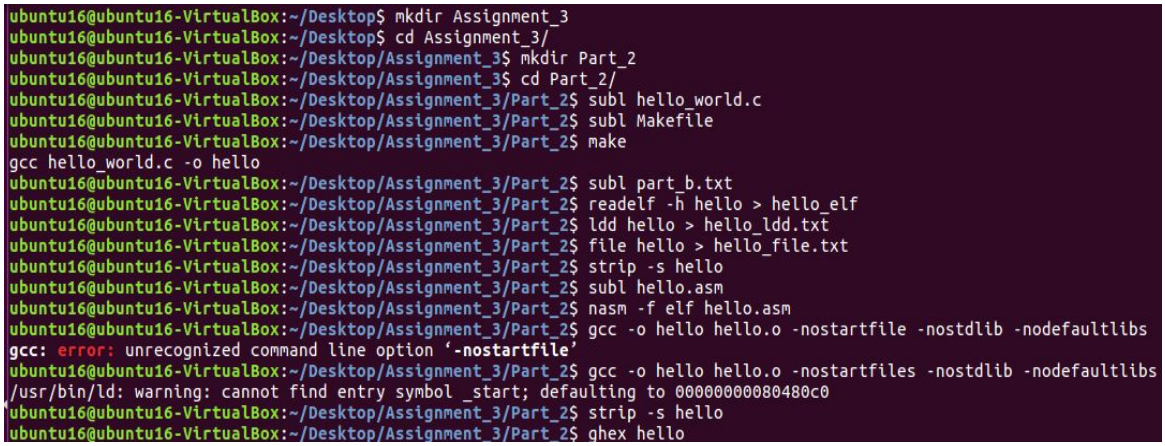


5) What does the strip command do?

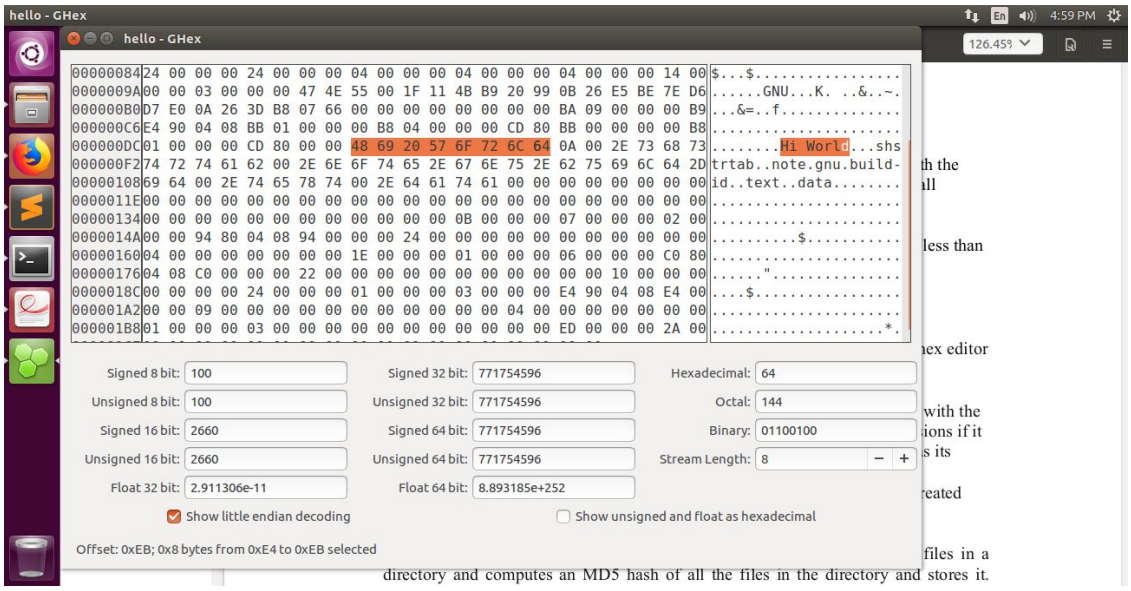
a) The GNU ‘strip’ command discards all symbols from object files and removes inessential information from executable binary programs. This inessential information refers to the data present that is not required for the normal execution of the binary file. This is done to better the performance and enable less disk space usage.

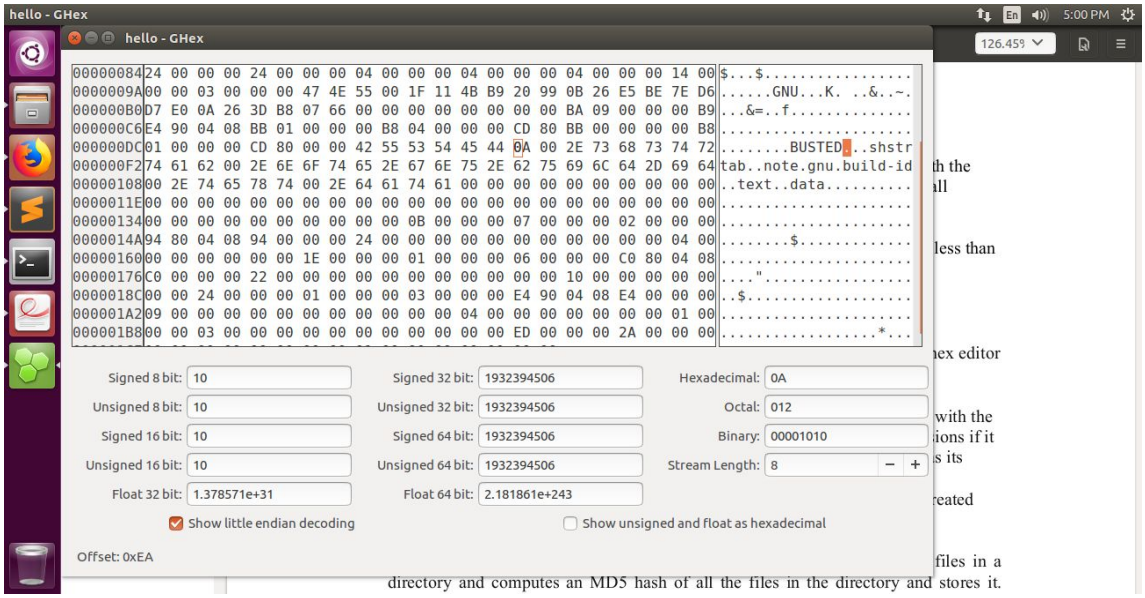
Part 2

1. Write a simple Hello World program in c. The program should have nothing but a #include and a print statement in main. Write a makefile to generate an executable called 'hello'. Record the size in number of bytes of the hello world program
   - 7.4Kb
2. Read and dump the ELF header from the executable type the command
   -
```
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           Intel 80386
  Version:                           0x1
  Entry point address:               0x8048310
  Start of program headers:          52 (bytes into file)
  Start of section headers:          6116 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         9
  Size of section headers:           40 (bytes)
  Number of section headers:         31
  Section header string table index: 28
```
3. To show all the dynamic linked libraries use the command ldd
   -
```
linux-gate.so.1 =>  (0xb7f21000)
libc.so.6 => /lib/i386-linux-gnu/libc.so.6 (0xb7d51000)
/lib/ld-linux.so.2 (0xb7f23000)
```
4. To get a description of what the file does do file hello .The output of the file command will show 'not stripped. Strip the file using the following command

- 

5. Compiling the asm: nasm -f elf hello.asm
    - (Same as above)
6. Compile & strip
    - gcc -o hello hello.o -nostartfiles -nostdlib -nodefaultlibs
    - strip –s hello
    - 

7. Open the ghex hexeditor. Using the editor replace the strings “Hi World” with the string “BUSTED”. Also embed a signature “DEADBEEFDEAD”. Remove all remaining bytes after the end of the signature.
   - 

   - Before
   - 

   - After

8. Record the number of bytes on the executable. It should have come down to less than 300 bytes
    - 284 bytes
9. Write a C program that:
    - Reads the ‘hello’ program as bytes and store its contents in memory.
    - Dump the contents of the file and compare it with what you see in the ghex editor and hexdump command to see if it matches. Write out your comments about what you see
    - Looks for any binary *.bin file, overwrites the contents of the binary file with the hello program. The overwritten file should also be given execute permissions if it already doesn’t have it. Running this program should display BUSTED as its output
    - Binary files may either be hand created using the ghex editor or can be created very simplistically using a perl or python script.
        i. **(Program ‘dump_virus.c’ attached with the mail)**
        ii. **(Input ‘make dump_virus’ to create executable)**

10. Write a scanner c program (a process that runs in a loop) that scans all files in a directory and computes an MD5 hash of all the files in the directory and stores it. Every T seconds the scans to see if the hash of a file is now different from the original hash. IF yes, it looks to see if the signature DEADBEEFDEAD exists in the file. The program can display an alert stating that the file has been infected. Typing Q should kill the process. The scanner program is run first and then the virus program is run to detect the infected files.

    i. **(Program 'scanner.c' attached with the mail)**
    ii. **(Input 'make scanner' to create executable)**
    iii. **Syntax: ./scanner <directory path>**
    iv. **e.g.**

        ./scanner . (Current directory for scanning)