Chapter 1 Homework
Annie Hayes

Due: Tuesday, January 19, 2021, 11:59 PM
Honestly, this homework may be tedious, but it should take you less time to complete than it took me to write it. It is not that hard. To make this as easy as possible I tried to keep the list of questions in the order the slides are in the slide deck.

Submission:
Submit this assignment through Canvas.

Do not hand write your answers. Typing the answers makes grading much easier. You are required type your answers in RED. You are also required to show what slide number you found the answer on.

To answer these questions, use the Chpt1Jan13.pdf, provided to you with this HW assignment. It is also listed on the January 13 notes module on the Canvas home page.

1. Slide # 2 How many bits are in 1 byte.
   There are 8 bits in one byte

2. Slide # 5 In our discussions, we determined information is basically a bunch of bits and that a series of bits can represent different data. As an example, the binary number 00100011 can represent the decimal number 35 or it can represent the ascii value of 35, which is '#'. The thing that distinguishes different data is the context in which we view the data.

3. Slide# 6 What are the 4 phases of the compilation system?
   Pre-processor
   Compiler
   Assembler
   Linker

4. Slide# 6 If you want to save all the temporary files created during the compile process what flag would you need to use when compiling?
   -save-temps
5. Slide# 7 - 10 Assume you have a file hello.c and you compile this file using the flag to save the temporary files and you use the flag -o to name the executable hello. What temporary file will be created by the:

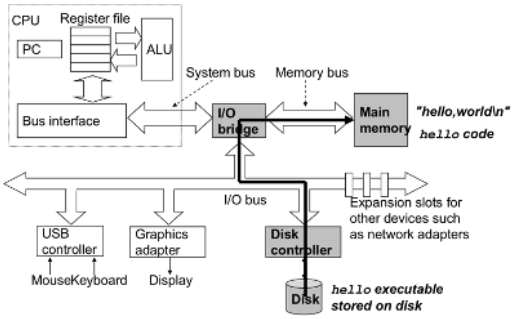
a. Pre-processing phase hello.i
b. Compiler phase hello.s
c. Assembler phase hello.o
d. Linker phase executable file

6. Slide# 11 What does the compile flag -O0 mean?
   This flag tells the compiler not to optimize your code.

7. Slide #13 With respect to hardware organization of a system, what does CPU stand for? Central Processing Unit
   We discussed three main areas of the CPU.
   List them: ALU - arithmetic/logical unit, PC - program counter, USB - universal serial bus

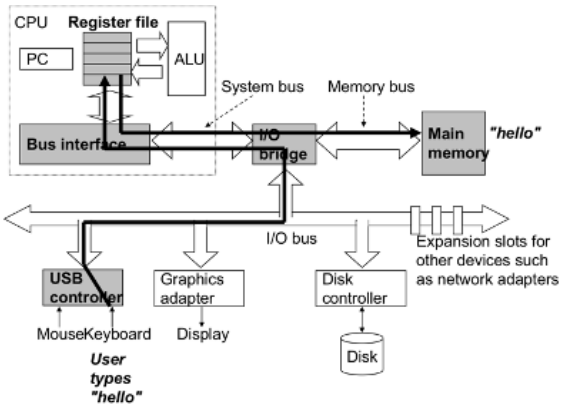
8. Slide #15 Describe what I/O devices are? Input/output devices are connected to the system from the outside world. List the 4 that were given in our example.
   a. Mouse
   b. Keyboard
   c. Display
   d. Disk drive
9. Slide #16 What is main memory? temporary storage device that holds a program and the data it manipulates while the processor is executing the program.
10. Slide #17 Central processing unit is the engine that executes the instructions stored in main memory. It has a word size register called the PC (program counter) that contains the address of some instruction in main memory.
11. Slide #18 We discussed 4 operations that the CPU might carry out at the request of an instruction. Name and describe the 4 operations:
    a. Load - copy a byte or word from main memory into a register, overwriting the previous content of the register
    b. Store - copy a byte or a word from a register to a location in main memory, overwriting the previous contents of that location
    c. Operate - Copy the contents of two registers to the ALU, perform an arithmetic operation on the two words, and store the results in a register, overwriting the previous data
    d. Jump - Extract a word from the instruction itself and copy that word into the PC, overwriting the previous value

12. Slide #19 With respect to running the hello world program, describe what is happening in the images below.Now we will explore what happens when we run the hello world program:



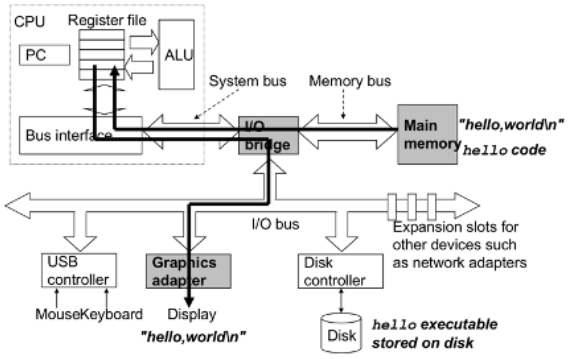
As you type characters on the command line the characters are being read and saved in a register and then copied to main memory

13. Slide #20 With respect to running the hello world program, describe what is happening in the images below.



When we hit the enter key the shell knows we are finished defining the command. The shell then loads the executable file by coping the code and data in the hello object file, stored on the disk, to the main memory. This step uses direct memory access (DMA) and does not need to pass through the processor

14. Slide #21 With respect to running the hello world program, describe what is happening in the images below.



Once the instructions from the object file are all loaded in memory, the processor begins executing the machine-language instructions in the hello program's main routine. In the end, the hello world string is produced and copied to the registers, then sent to the output display.

15. Slide #22 What was the motivation of developing Cache storage?

System designers designed smaller faster storage devices called cache memories to serve as temporary staging areas for information that the processor is likely to need in the near future.

16. Slide #24-25 The OS is a layer of software interposed between the application program and the hardware. This layer of software has 2 primary purposes. What are they:
    a. To protect the hardware from misuse by runaway applications
    b. To provide applications with simple and uniform mechanisms for manipulating complicated and often wildly different low-level hardware devices
17. Slide #26 With respect to an operating system, what is a process?
    A process is the operating system's abstraction for a running program
18. Slide #30 Starting at the zeroth address, list the 5 areas that make up virtual memory for Linux?
    Program code and data
    - Size does not change after program gets started

    Heap
    - Grows and shrinks as needed - dynamic memory

    Shared libraries (stdio, stdlib, etc)
    Stack
    - Grows and shrinks as needed
    - Used to implement function calls

    Kernel Virtual Memory
    - Reserved for the OS

19. Slide #33 What are two demands that have been a constant force in driving improvements in computers and how can they be achieved?
    We want them to do more
    And, do more faster
20. Slide #36 What is a multi-core processor and describe how it is organized?
    Multi-core processors have several CPU's (referred to as "cores") integrated onto a single integrated-chip
        Each core has its own L1 and L2 cache
        L1 is divide in two parts
            One holds instructions
            One holds data

    Experts predict one day dozens or even hundreds of cores will be on a single chip

21. Slide #39 Describe the listed 2 ways multiprocessing improves system performance.
    By reducing the need to simulate concurrency when performing multiple tasks

By running a single application program faster if that program is expressed in terms of multiple threads that can effectively execute in parallel

22. Slide #39 Describe the two approaches to instruction level parallelism.
    Hardware: dynamic parallelism means the process decides at run time which instructions to execute in parallel
    Software: Static parallelism means the compiler decides which instructions to execute in parallel
23. Slide #41 When and who developed the “C” programming language?
    C was developed from 1969 to 1973 by Dennis Ritchie of Bell Laboratories