* *Open source softwares:*
  + Firefox, Android, Apache
* *GNU/Linux—Open Source operatin system*
* **Kernel**: core of operating system
  + Allocates time and memory to programs
  + Handles file system and communication between software and hardware
* **Shell**: interface between user and kernel
  + Interprets commands user types in
  + Takes necessary action to cause commands to be carried out
* **Programs**
* *Ways to access:*
* **On the computer:**
  + Install or try Ubuntu
  + Run with Windows (https://wiki.ubuntu.com/WubiGuide)
  + Easy to remove **Ubuntu** from **Windows** via **Control Panel**, if you don’t need Ubuntu.
* **Virtual Machine**
  + VMWare
  + Virtual Box
* **Live CDs on BH3760 Computers** 
  + **Don’t** install Ubuntu
  + Try Ubuntu
* **SEAS Server** 
  + lnxsrv.seas.ucla.edu
  + Username: SEAS ID
  + Password: SEAS password
  + On windows: putty or cygwin
* **Connecting to SEAS from OS X or Linux**
  + Terminal
  + $ ssh [username@lnxsrv.seas.ucla.edu](mailto:username@lnxsrv.seas.ucla.edu)
  + Username = your SEAS user name
* CSC 2013
  + Machine level representation of data
  + Assembly level machine organization
  + Interfacing & communication
  + Functional organization
  + Multiprocessing & alternative architectures
  + Performance enhancements
  + Memory management
  + Parallelism intro
* Schedule
* 1st week: intro integers
* 2nd week: MLP basics, control
* 3rd week: procedures, data structures
* 4th week: pointers, midterm I (Apr 23)
* 5th week: Floating pointing, optimization
* 6th week: ILP(Instruction level parallelism) memory hierarchy
* 7th week: TLP(Thread level parallelism) I/O midterm II (May 14)
* 8th week: synchronization
* 9th week: virtual memory linking
* 10th week: error handling , exception, etc
* 40% labs (4\*10%)
* 5% homeworks(5\*1%)
* 25% midterms(2\*12.5%)
* 30% final exam
* CCLE 23:55
* HW1 Due April 10th 23:55
* Data Lab April 15th 23:55
* 1. Abstractions “c++ machine”
* 2.”real machine” X86
* 3. real machine secret sub X86 machine
* Problems with abstractions
* --performance
* --debug—incompetent programs
* --security—malware
* Goal:
* 1.Write better software
  + Avoid bugs
  + Find and fix bugs more efficiently
  + Performance bugs are bugs
* 2.Understand computer systems
  + Operating systems
  + Networks
  + Architecture
  + Parallelism
* gcc –E sum.c
* #define ADD(a,b) ((a)+(b))
* int sum(int a, int b)
* return ADD(a,b)

gcc –m32-O2-c sum.i

int sum(int a, int b)

return ((a)+(b))

* mass number(positive)
* “the amount of stuff in an object”
* Inertia: how hard is it to change the motion of an object
* Time: the quantity measured by a clock
* Length: the quantity measured by a ruler
* Combination of mass, length, time that make up a quantity is its dimension
* Quantity q
* Dimension of q [q]
* [Time]=T
* [Length]=L
* [mass]=M
* [velocity]=L/T
* Kilogram is defined by a metal bar in Europe
* Second is atomic clock frequency
* Meter is distance travelled by light (in vacuum) in 1/299792458 seconds
* Fact: the result of a computation has the wrong dimensions then its wrong-check your algebra
* Motion in one dimension
* Kinematics in !D
* Study of motion independent of its cause
* Position velocity acceleration
* Displacement change in position: x(ta), x(tb); then x(tb)-x(ta)=displacement
* Velocity average/instantaneous
* Speed = magnitude of velocity
* Speed at time t=|v(t)|
* Acceleration average/instantaneous