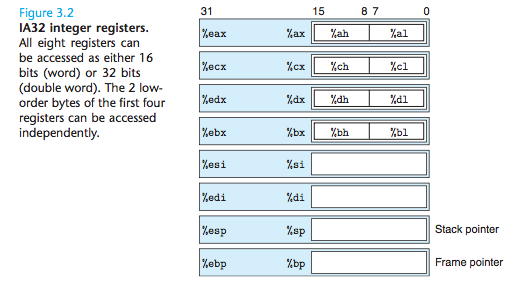
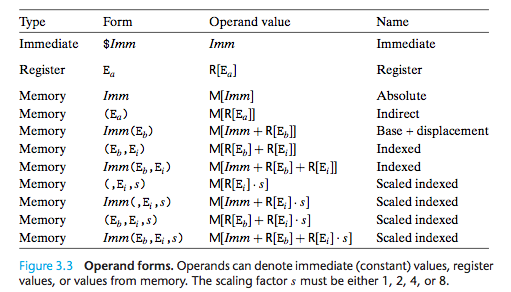
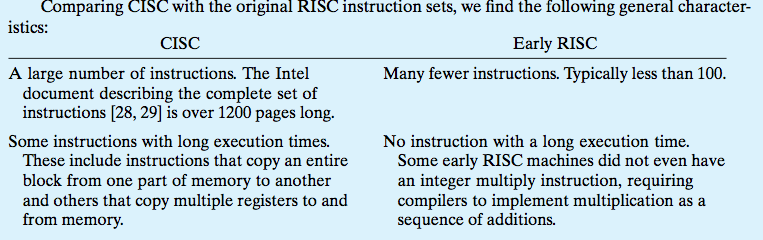
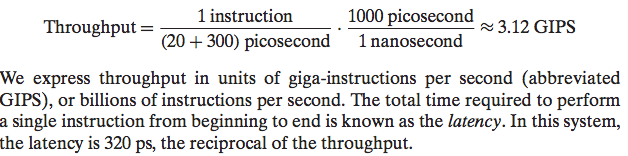
Reading notes for the Final:

* In a program with good local temporal locality, a memory location that is referenced once is likely to be referenced again multiple times in the near future
* In a program with good spatial locality, if a memory location is referenced one then the program is likely to reference a nearby memory location in the near future.
* At the hardware level the principle of locality allows computer designers to speed up main memory access by introducing small fast memories known as cache memories that hold blocks of the most recently referenced instruction and data items.
* Addl rA, rB = rB = rA + rB
* Subl rA, rB = rB = rA – rB
* 0x01 5
* 0x02 6
* 0x03 -2
* 0x04 3
* rB = 0x02
* rA = $5
* addl rA, rB = 5+6 and store it at 0x02 which is the value of rB
* Virtual memory only contains a cache of address spaces
* Unallocated pages have not been allocated by the mem system AKA do not hold any space on the disk



* 



* 
* Structural Hazards: instruction need the same hardware at same time.
* Data Hazards: data for the next instruction is not computered until a specific instruction completes
* -so you use data forwarding
* -forward to allow the next instruction to begin its fetch
* -sometime you can “move back in time”
* -ex: addressing is done at the end of an execute phase
* -solution: we stall the phase: used with the no-op command
* Control Hazards:Not going to know the outcome of an instruction by the time we need to insert an instruction (non-deterministic)
* -to fix this we have to use branch prediction
* -makes educated guesses about which instruction to propagate
* -this is in an if else form
* Locality:
* Well written programs exercise the principle of locality, that is, they reference data that is near other recently referenced data.
* Two forms of locality:
* *Spatial Locality:*
* If a memory location is referenced once, it is likely that nearby locations will be referenced in the near future.
* *Temporal Locality:*
* If a memory location is referenced once, it is likely that it will be reference multiples times in the near future
* Stride:
* Programs and functions can be evaluated in terms of their spatial and temporal locality.
* Stride refers to the reference pattern of data being accessed by a function or program.
* Clock cycle is how long one step takes
* Latency is the total time required to compute a whole instruction
* 