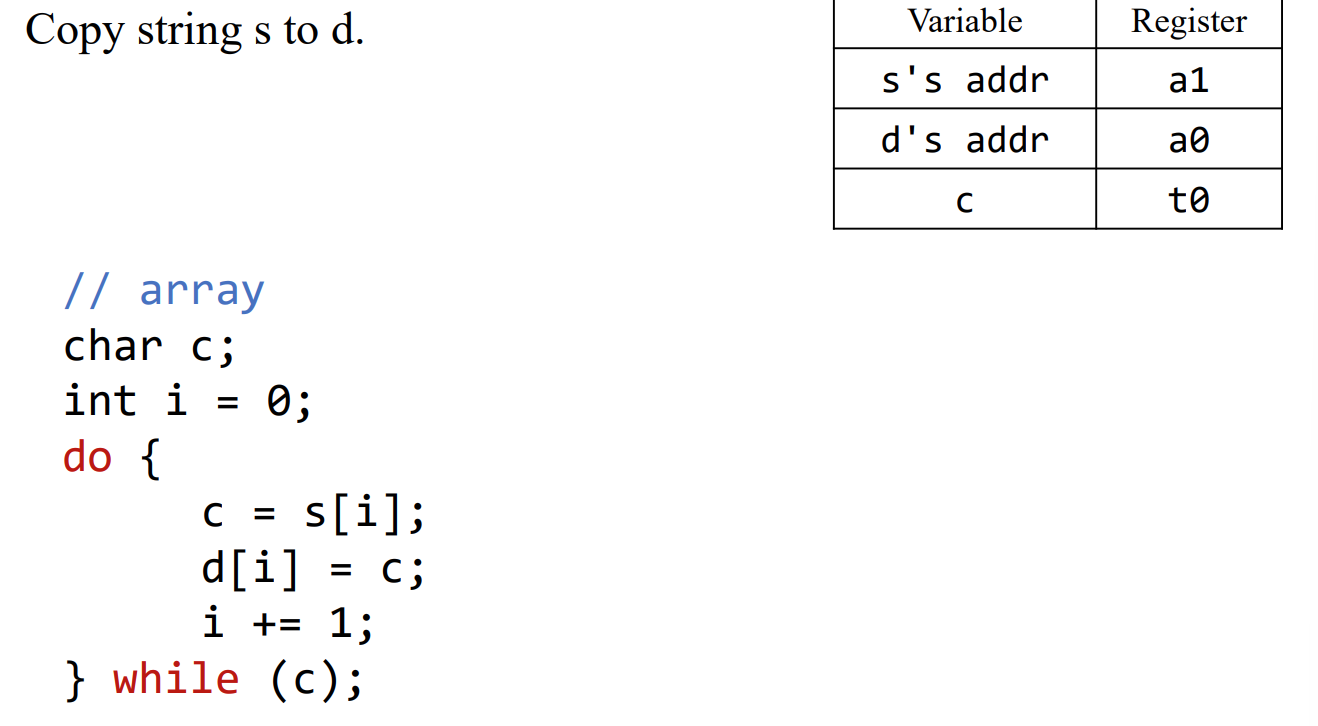
Review previous slides:

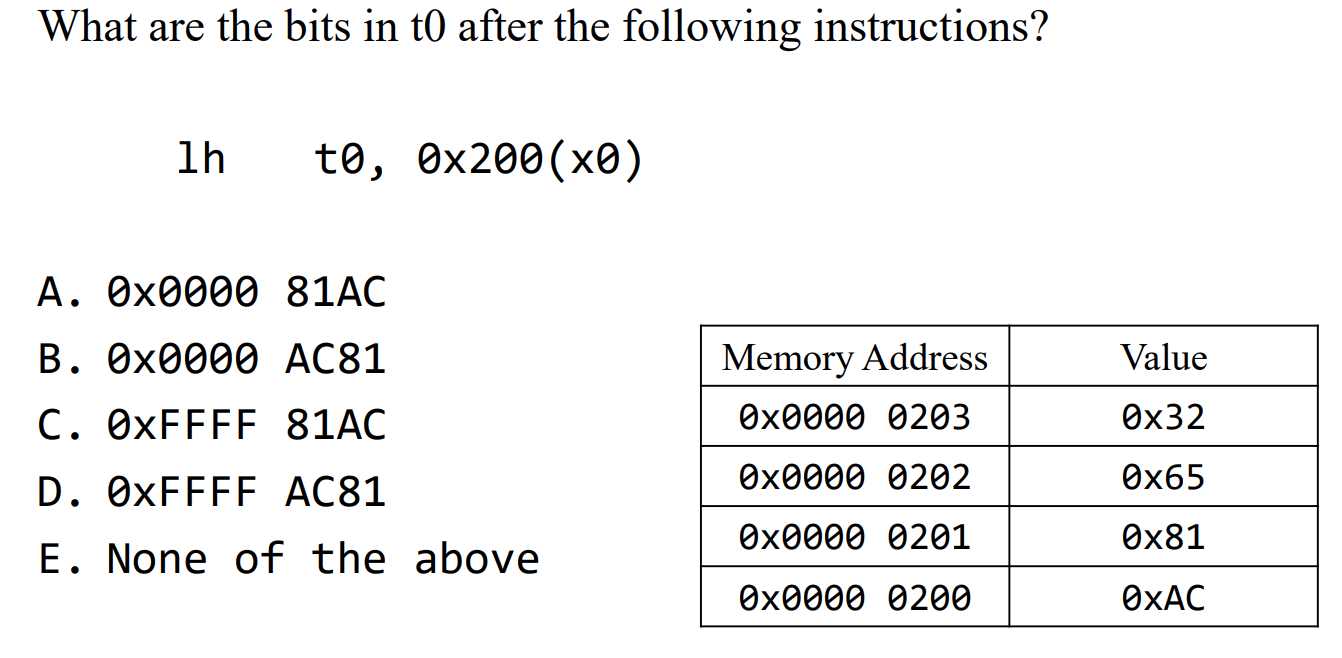


c = s[i];

lb t0, 0(t2)

Pitfalls

* A word has four bytes
  + LW loads four bytes
    - There are four bytes in a word! They are located at sequential addresses
  + Sequential word addresses are incremented by 4!
* Sequential half words/bytes are NOT incremented by 4
  + Pay attention to the size
  + Sequential bytes do have sequential addresses
* Offset is a 12-bit 2's completed number, sign extended to 32 bits – If offset is too large, add offset with instructions
* Byte order matters



The byte order (lower address is lower)

Correct is C

Find Bytes 81AC then sign extension

Instruction Encoding - 1

Best Strategy is not to memorize but understand

Representing instructions with bits

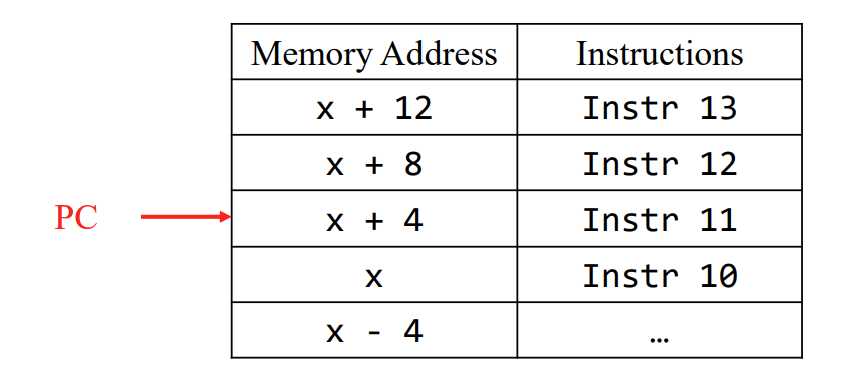
* We use bits to represent numbers, characters, etc.
* We also use bits to represent instructions

Design Questions:

* How many bits should we use to encode instructions?
* Are we using the same number of bits to encode all instructions?
  + Do all instructions have the same length?
    - Add,Addi,beq

Risc-v instruction words

* Base ISA are encoded as 32-bit instruction words
  + Encoded instructions are also called machine (language) code
* Both instructions and data stored in memory
* Program Counter (PC) points to the current instruction
  + Incremented by 4 inn normal flow for sequential execution



Information that you want to keep in the instruction word?