Topics: Arrays, Pointers, Functions

Approach: Introduce Pointers, Explore their use

Main Ideas:

Quick Review - Forms, Connector, Scripts, Tools in C
- Focus tonight on memory usage in C

Recall Memory
- A sequence of boxes, each numbered
- All data and code live in memory
- Today we learn to program with the addresses of memory

Why?
- Pass by reference : useful!
- Linked data structures: Really useful
- dynamic memory : super useful
- other reasons : no so important

Types of storage:
- single values : char, int, float
- array: contiguous sequence of one type
- struct: varied types in one container

Single values:
- Where are the values stored?
  - ex1pa.c -- print the addresses
  - ex1sa.c -- store the addresses
- What can we do with pointers?
  - ex1dp.c -- dereference pointes
  - ex1cp.c -- compare pointers
  - ex1pf.c -- pass pointers to functions
- Question: Do pointers have addresses?

Arrays:
- ex2.c -- take address, deref, compare
- What can we do with pointers to arrays?
  - ex2ia.c -- index into array using [] notation
  - ex2ao.c -- arithmetic (++,--,+,-)
  - ex2ae.c -- more exercises -- trace these

Structs:
- ex3.c -- pointers and structs
- What can we do?
- Take addresses, compare, assign, select members
- pass to functions by reference

Arrays of Pointers
- ex4.c -- what does this code do?
- Draw a picture.