C Language Concepts

CS 231

pointers

Using scanf

- *scanf reads a value and puts it in memory. It needs the address of memory in which to put the value. We typically do this by:*

  ```
  int a;
  scanf("%d", &a)
  ```

- *If p is a pointer to int, scanf can use this:*

  ```
  int *p;
  scanf("%d", p);
  ```

- *See program ptr1.c*

Memory and Pointers

- *A common programming error is to declare a pointer, but do not initialize it to point to a memory location before use.*
- *This is the error in ptr1.c*
- *Fix this in ptr2.c where p points at declared int*

Memory and Pointers

- *You can also allocate memory at run time and have the pointer point at the allocated memory. Program ptr3.c shows this.*
- *The pointer must be initialized to point at memory which is part of your program space, or there will be an error.*

Memory and Pointers

- *ptr4.c shows that the contents of a pointer to memory is actually the memory address*
- *This is guaranteed by the operator &.*

Arrays and sizeof

- *An array can be initialized (without specifying size) by assigning to a comma separated list of values inside curly braces {1,2,3} for example*
- *If size is specified, the first elements in array are initialized*
- *sizeof will return the size of the array*
- *see code in file arraysize.c*

Input Lines Using Pointers

- *Problem: lines are of unknown size, number of lines not known*
- *Want to input and store all lines, then output lines in reverse order (just to do something with them)*
- *Code in file inputLines.c*