CS 107
Lecture 6:
Arrays, Pointers

Friday, October 13, 2017

Computer Systems
Fall 2017
Stanford University
Computer Science Department

Lecturers: Julie Zelenski and Chris Gregg
Today’s topics

- Pointer == array?
- Pointer arithmetic, array indexing
- Const pointers
- Stack allocation vs heap allocation
Array declaration

```c
int nums[5];
```
Allocates space for 5 ints, contiguous memory, indexed from 0 to 4, length not tracked
Use of array name "decays" to address of first element

Pointer declaration

```c
int *ptr;
```

What is same? What is different?

- **Operations that work on both**
  - Dereference
  - Pointer arithmetic
  - Array indexing

- **Operations that only work only on pointer**
  - Can reassign the pointer to hold a different address, not so with array

Does the memory layout look the same? Let's draw it!
Pointer arithmetic

- **Array indexing just "syntactic sugar" on pointer arithmetic**
  
  
  \[
  \text{ptr} + i \quad \leftrightarrow \quad \&\text{ptr}[i]
  \]
  
  \[
  *(\text{ptr} + i) \quad \leftrightarrow \quad \text{ptr}[i]
  \]

- **Arithmetic scaled by sizeof(pointee)**
  
  \[
  \text{ptr} + 1 \text{ adds one if ptr is char *}, \text{what if ptr is int *?}
  \]
  
  What happens if you cast to different size pointee before arithmetic?

- **No bounds checking**
  
  Programmer is responsible for tracking size, pointers/arrays don’t know it
Let's code & draw!

/afs/ir/class/cs107/samples/lect6

ptrarr.c
Time travel into your future