CS 161
Intro to CS I

Finish 1-D arrays/C-style string
Odds and Ends

• Last week to demo Assignment 4
• I’ll be out of town Wed./Fri.
• Exam II next Wed.
Class Exercise

• How do I initialize an array in a function?
• How can I print the contents of the array in a function?
• How about printing the address of the pointer to the array and the address of where the array begins in a function?

• How would I create an array in a function?
```cpp
using namespace std;

void init_arrays(int *array, int array_heap[]) {
    // initialize the elements in the arrays
    for(int i=0; i<10; i++) {
        array[i]=i;
        array_heap[i]=i+1;
    }
}

void print_pointer_addr(int (*array)[10], int **array_heap) {
    // print where the pointer to the array lives in memory
    cout << "stack array pointer lives: " << array << endl;
    cout << "heap array pointer lives: " << array_heap << endl;
}

int main() {
    // create an integer array with 10 elements on stack and one on the heap
    int array[10]={3}; // does not initialize all elements to 3, only 1st
    int array1[]={1,2,3,4,5,6,7,8,9,10};
    int *array_heap;
    array_heap=new int[10];

    init_arrays(array, array_heap);
    print_pointer_addr(&array, &array_heap);
```
Finish Implementing...
What are C-style strings?

• Ended by ‘\0’ character
• Need to include <cstring>
C-Style Strings Demo...
Multidimensional Arrays

• data_type array_name[rows][cols];
  – int array[2][3];
  – int array[4][2][3];
  – int array[2][4][2][3];

• What are examples of these?
  – 2-D – Matrices, Spreadsheet, Minesweeper, Battleship, etc.
  – 3-D – Multiple Spreadsheets, (x, y, z) system
  – 4-D – (x, y, z, time) system
Initializing 2-D Arrays

- **Declaration:**
  ```c
  int array[2][3] = {{0,0,0},{0,0,0}};
  ```

- **Individual elements:**
  ```c
  array[0][0]=0; array[0][1]=0; array[0][2]=0; array[1][0]=0;
  array[1][1]=0; array[1][2]=0;
  ```

- **Loop:**
  ```c
  for(i = 0; i < 2; i++)
      for(j = 0; j < 3; j++)
          array[i][j]=0;
  ```

- **Why do we need multiple brackets?**
Reading/Printing 2-D Arrays

• Reading Array Values
  for(i = 0; i < 2; i++)
    for(j = 0; j < 3; j++) {
      cout << “Enter a value for “ << i << “,” << j << “: ”;
      cin >> array[i][j];
    }

• Printing Array Values
  for(i = 0; i < 2; i++)
    for(j = 0; j < 3; j++)
      cout << “Array: “ << array[i][j] << endl;