CS 162
Intro to CS II

Arrays/Vectors
Chap. 7.3
Odds & Ends

• Assignment #1 demos start next week...
• Assignment #2...
```cpp
#include <iostream>
#include <stdlib.h>
#include <stdio.h>

using namespace std;

void create_2d_c(int ***d_arr) {
    *d_arr=(int **)malloc(sizeof(int *)*5);
    for(int i=0; i<5; i++)
        (*d_arr)[i]=(int *)malloc(sizeof(int)*5);
    (*d_arr)[0][0]=100;
}

int main () {
    int s_arr2[5][5], **d_arr2= NULL;

    /*d_arr2 = new int*[5];
    for(int i=0; i<5; i++)
        d_arr2[i] = new int[5];
    cout << "C++ addressing: " << d_arr2 << endl;
    //Memory Leak!!! we didn't delete C++ new...
    d_arr2 = (int **)malloc(sizeof(int *)*5);
    for(int i=0; i<5; i++)
        d_arr2[i] = (int *)malloc(sizeof(int)*5);
    printf("C addressing: %x\n", d_arr2);
    */
    create_2d_c(d_arr2);
    cout << "C d_arr2: " << d_arr2[0][0] << endl;
```
Vectors vs. Arrays

• What is the difference?
• What are the similarities?
• How do we declare a vector?
  
  ```
  #include <vector>
  std::vector<Base_Type> v(10);
  ```
  
• How do we assign a value?
  
  ```
  v[0] = 20;
  v[1] = 30;
  ... 
  ```
More Vectors...

• What if we don’t know the size?
  
  ```cpp
  vector<int> v;
  v.push_back(20);
  v[0] = 5;
  ```

• We can get the size: `v.size()`

• We can resize using: `v.resize()`

• We can get the capacity: `v.capacity()`

• We can set the capacity: `v.reserve()`
```cpp
#include <iostream>
#include <vector>

using namespace std;

int main () {
    vector<int> v;

    cout << "Vector size: " << v.size() << endl;
    cout << "Vector capacity: " << v.capacity() << endl;
    cout << "Vector address: " << &v << endl;

    v.resize(10);
    cout << "Vector size: " << v.size() << endl;
    cout << "Vector capacity: " << v.capacity() << endl;
    v.reserve(v.size()+20);
    cout << "Vector size: " << v.size() << endl;
    cout << "Vector capacity: " << v.capacity() << endl;

    cout << "Vector address: " << &v << endl;
    cout << "Vector address[0]: " << &v[0] << endl;
    cout << "Vector address[1]: " << &v[1] << endl;
    cout << "Vector address[2]: " << &v[2] << endl;
    cout << "Vector address[10]: " << &v[10] << endl;
    cout << "Vector address[10]: " << &v[10] << endl;

    cout << "Vector 10: " << v[10] << endl;
    v.push_back(10);
    cout << "Vector size: " << v.size() << endl;
    cout << "Vector 3D: " << v[10] << endl;
    cout << "Vector capacity: " << v.capacity() << endl;
}
```
#include <iostream>
#include <vector>
using namespace std;

int main() {
  vector<vector<int>> v;

  //create 20 x 20 2-d "array"
  for(int i = 0; i < 20; ++i){
    vector<int> tmp(20);
    v.push_back(tmp);
  }

  return 0;
}
Quiz #2

- Get into groups of 4 – 5.
- Bitwise Ops/Bit Shifting

```cpp
unsigned char a = 0b01;

// Bit Shifting...
cout << (a << 4) << endl;
cout << (a >> 1) << endl;

a = ~a;
cout << (int) a << endl;
cout << (a >> 1) << endl;

// Bitwise Ops...
cout << (a & 0b00110011) << endl;
cout << (a | 0b00110011) << endl;
cout << (a ^ 0b00110011) << endl;
```

- Finish `create_2d_cpp()` { }