CS 161
Intro to CS I
Static/Dynamic Arrays
Odds and Ends...

• Design due Sunday by 5pm on Canvas
• Friday recitations go to different one
• No class Friday, Happy Veterans Day!
Read/Print 1-D Array Values

• Read Values From User
  
  ```cpp
  for(i=0; i<5; i++) {
    cout << "Enter final grade for student: ";
    cin >> student_grades[i];
  }
  ```

• Print Values
  
  ```cpp
  for (i=0; i<5; i++) {
    cout << "Student\'s final grade is " << student_grades[i] << endl;
  }
  ```
```cpp
#include <iostream>
#define NUM 5
using namespace std;

int main() {
    int a[NUM], *b;  // static array on stack
    int num;
    cout << "enter number of elements: ";
    cin >> num;
    // int a[num];  // variable length array on stack, not dynamic
    b = new int[num];  // dynamic array on the heap

    cout << &b << " " << b << " " << &b[0] << endl;
    // fill b and print b
    delete [] b;  // delete the array I point to
    b = a;  // can make a dynamic array point anywhere

    for(int i = 0; i < NUM; i++)
        *(b+i) = i;  // same as b[i]
    cout << &a << " " << a << " " << &a[0] << endl;
    for(int i = 0; i < NUM; i++)
        cout << a[i] << " ";
    cout << endl;

    return 0;
}  
```
Static vs. Dynamic 1-D arrays...
Passing a 1-D Array (Static/Dynamic)

```c
int main() {
    int array[5];
    ...
    pass_1darray(array);
    ...
}
void pass_1darray(int *a) {
    cout << "Array at zero: " << a[0] << endl;
}
OR
void pass_1darray(int a[]) {
    cout << "Array at zero: " << a[0] << endl;
}
```
How does freeing memory work?

```cpp
int *p, *q;

p=new int;
q=new int[5];

delete p;
delete [] q;
```
What are the similarities/differences?

- **String Object vs. C String**
  - Which library to include?
    - `<string>` **VS.** `<string.h>` or `<cstring>`
  - How do we create it?
    - `string str_obj;` **VS.** `char str_arr[20];`
  - How do we access it?
    - `str_obj.at(3)` or `str_obj[3]` **VS.** `str_arr[3]` or `*(str_arr+3)`
  - How do we get the length?
    - `str_obj.size()` or `str_obj.length()` **VS.** `strlen(str_arr)`
  - How is length of string determined?
    - Size member variable **VS.** ‘\0’, null character at end
```cpp
#include <iostream>
#include <string.h>

using namespace std;

int main(){
    char str[256];

    cout << "give string: " << endl;
    cin >> str; //puts '\0' on end of string

    cout << strlen(str) << endl; //get length of string
    cout << str << endl; //see the whole string
    cout << (int *)str << endl; //see the address of where string is

    return 0;
}
```
Revisit Creating Memory in Functions

Advantages to Dynamic Memory
int *i=NULL;//created in main function

create_mem(&i); //call in main
void create_mem(int **m) {
  *m = new int[5];
}
OR
i = create_mem(); //call in main
int * create_mem() {
  return new int [5];
}