

CS 161

Intro to CS I

Begin Structs vs. Classes

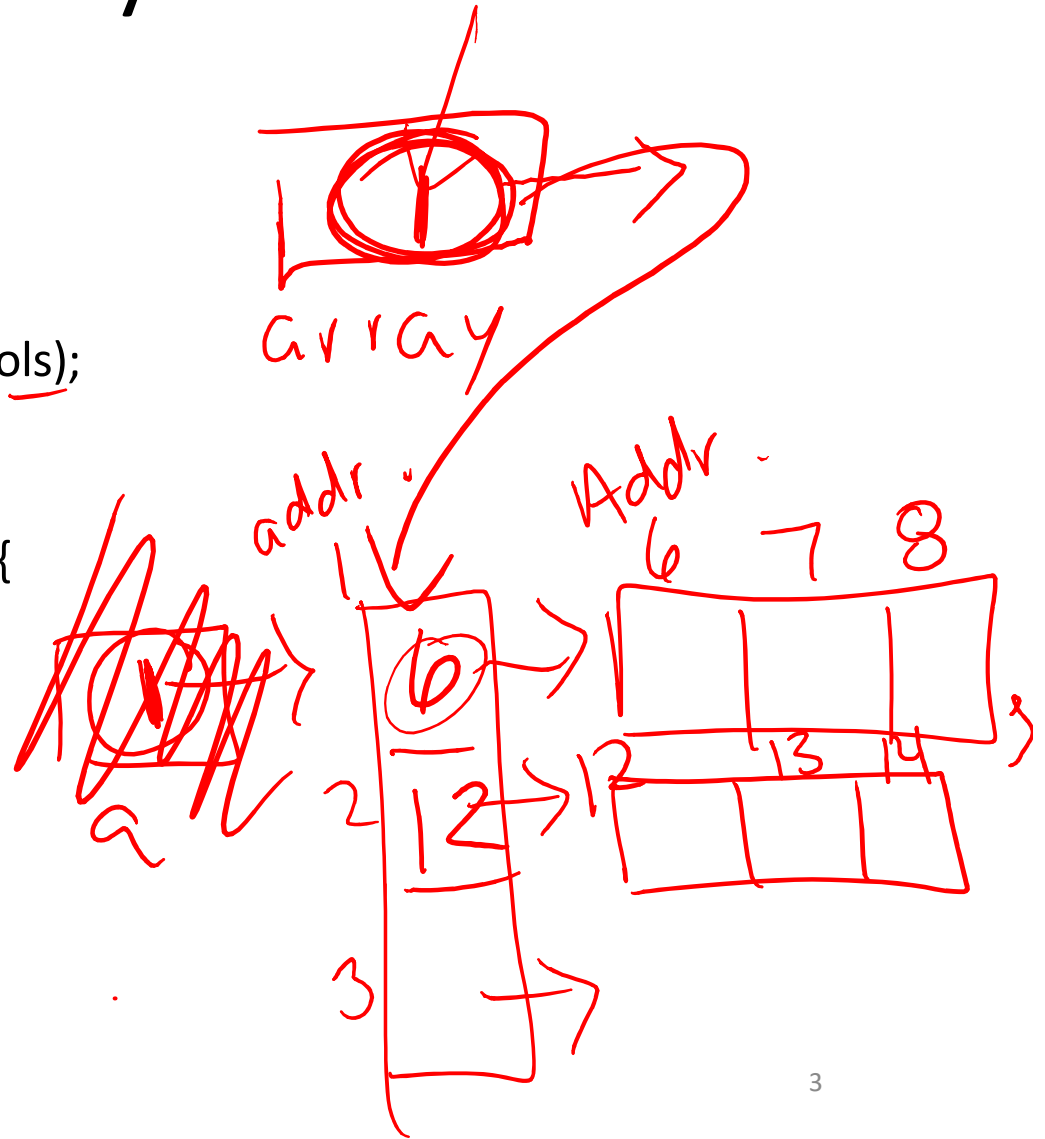
Odds and Ends

- Demo Assignment #5 this week!!!
- Assignment #6 due Sunday
 - This assignment is not demoed!!!
- Final Exam next Thursday, 9:30am!

- Any questions on Assignment 6 or arrays?

Create 2-D Array in Functions

```
int main() {  
  int **array;  
  ...  
  array = create_2darray(rows, cols);  
  ...  
}  
int **create_2darray(int r, int c) {  
  int **a;  
  a = new int*[r];  
  for(int i=0; i<r; i++)  
    a[i] = new int[c];  
  return a;  
}
```

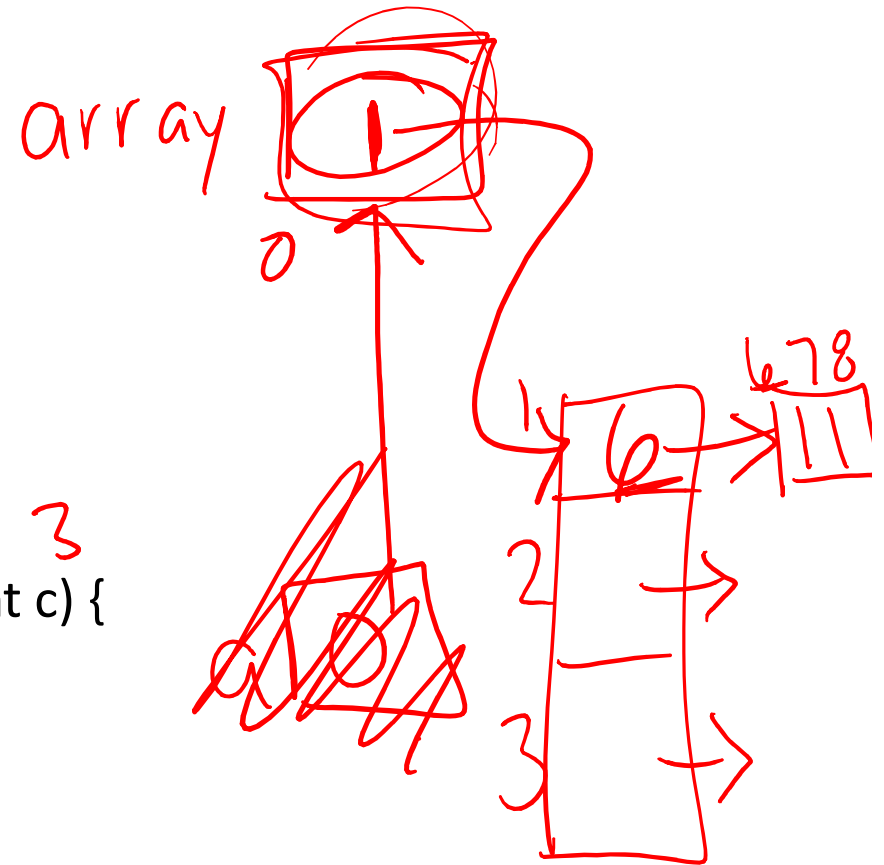


Create 2-D Array in Functions

```
int main() {  
    int **array;  
    ...  
    create_2darray(&array, rows, cols);  
    ...  
}  
void create_2darray(int ***a, int r, int c) {  
    *a = new int*[r];  
    for(int i=0; i<r; i++)  
        (*a)[i] = new int[c];  
}
```

Handwritten annotations:

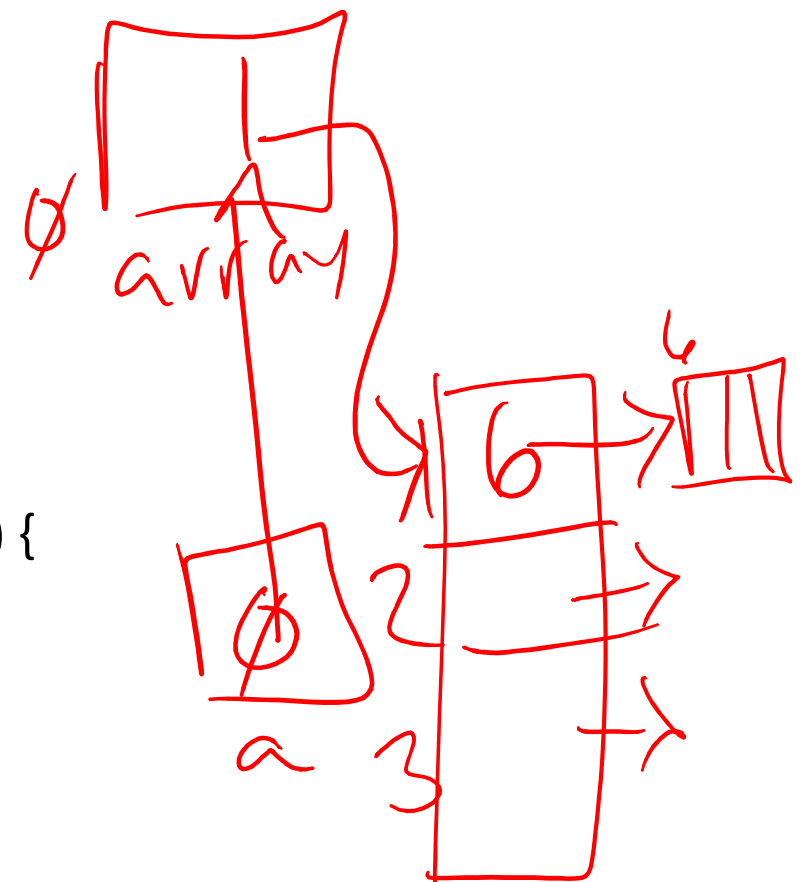
- int** (with a scribble) next to `int **array;`
- int** (with a scribble) next to `int ***a`
- new int*[r] (with a scribble) next to `*a = new int*[r];`
- new int[c] (with a scribble) next to `(*a)[i] = new int[c];`
- Handwritten numbers `3 3` next to `int r, int c`



Create 2-D Array in Functions

```
int main() {  
    int **array;  
  
    ...  
    create_2darray(array, rows, cols);  
  
    ...  
}  
  
void create_2darray(int **&a, int r, int c) {  
    a = new int*[r];  
    for(int i=0; i<r; i++)  
        a[i] = new int[c];  
}
```

that is a ref to



How does freeing memory work?

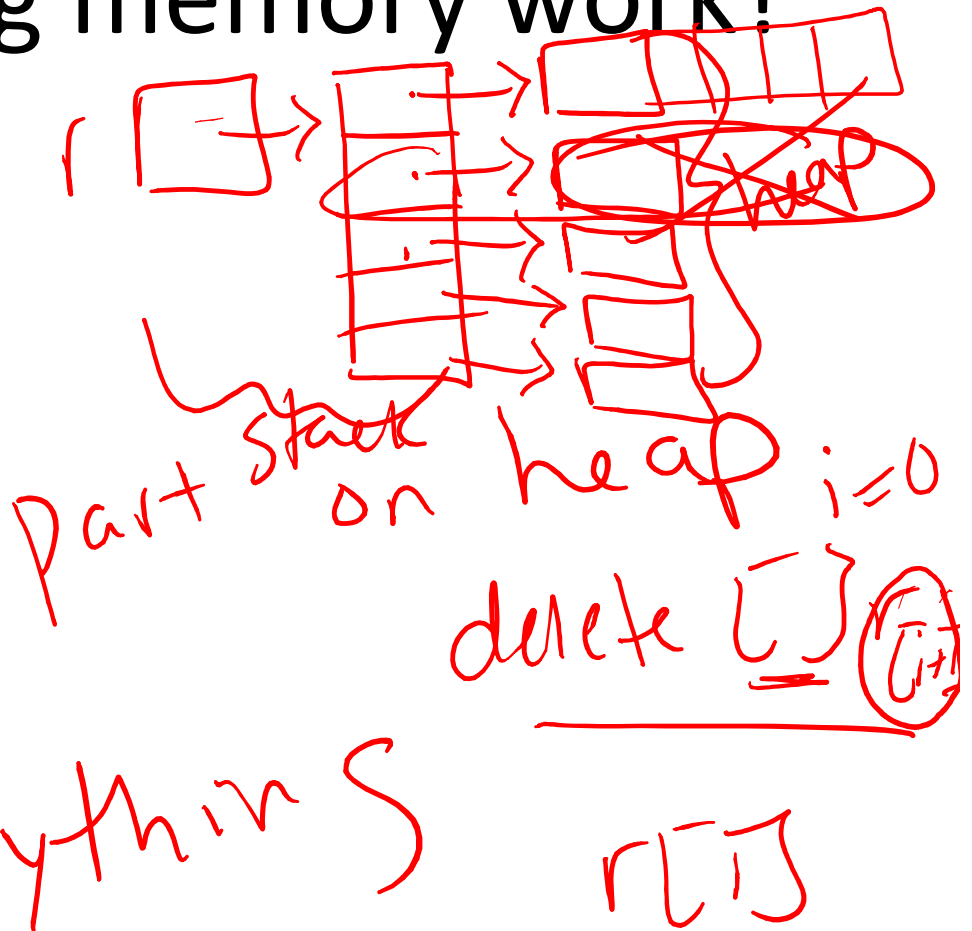
```
int *r[5], **s;
```

```
for(int i=0; i < 5; i++)  
  r[i]=new int;  
for(int i=0; i < 5; i++)  
  delete r[i];
```

```
for(int i=0; i < 5; i++)  
  r[i]=new int[5];  
for(int i=0; i < 5; i++)  
  delete [] r[i];
```

```
s=new int*[5];  
for(int i=0; i < 5; i++)  
  s[i]=new int[5];  
for(int i=0; i < 5; i++)  
  delete [] s[i];  
delete [] s;
```

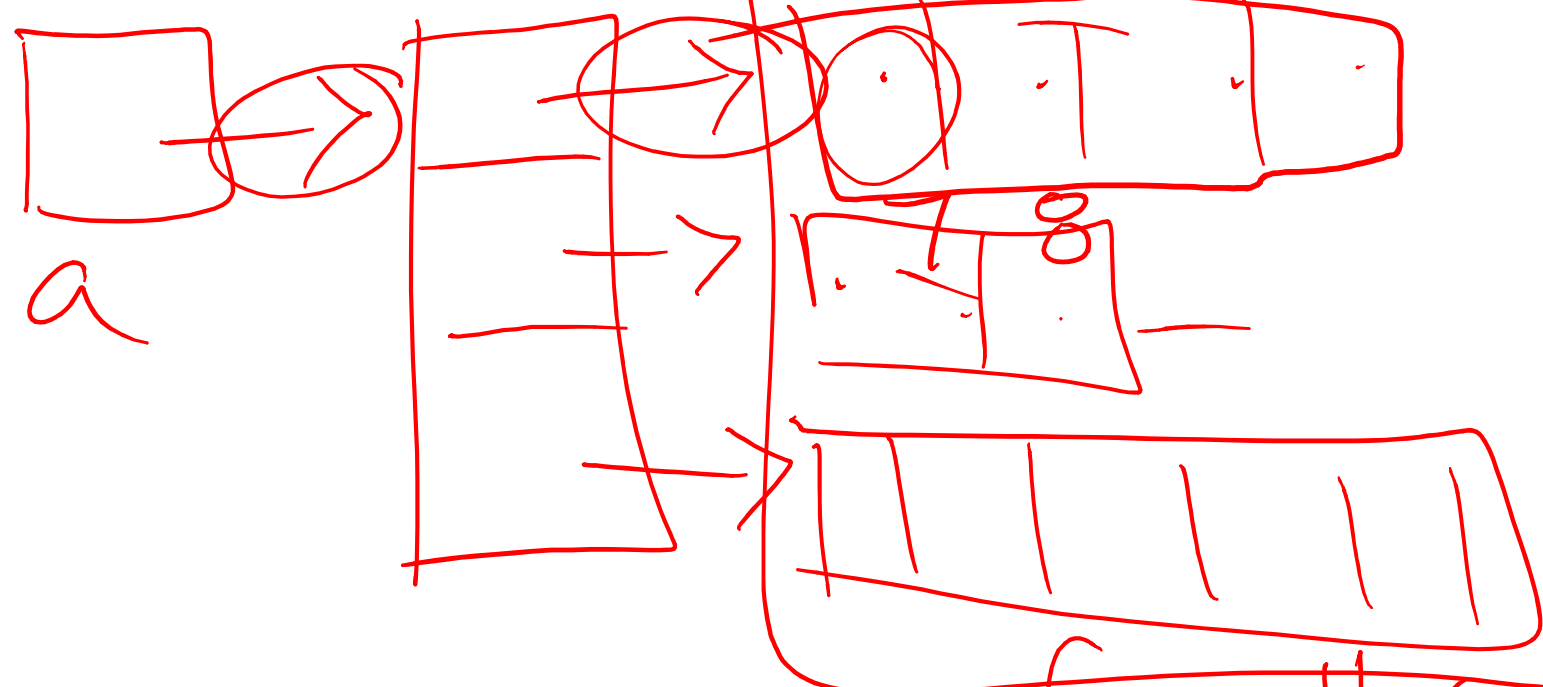
create
delete



int **a;

a [0] [0]

0 1 2 3



a = new int * [3]
for (all row ptrs
a[i] = new int [?]

for all row ptrs
delete [3] a[i]

```
8 int main(int argc, char **argv) {
9     //how do I create a 2-d array (3 x 3) on stack?
10    int array[3][3]={1,2,3}, {4,5,6}, {7,8,9}};
11
12    fun(array, 3, 3);
13    //how do I print address of pointer to 1st row pointer?
14    cout << &array << endl;
15
16    //how do I print address of 1st row pointer?
17    cout << array << endl;
18    cout << &(array[0]) << endl;
19
20    //how do I print address of the 1st element in 1st row?
21    cout << array[0] << endl;
22    cout << &(array[0][0]) << endl;
23
24    //how do I print address of the 1st element in 2nd row?
25    cout << array[1] << endl;
26    cout << &(array[1][0]) << endl;
27
28    //print contents of 2nd element in 2nd row?
29    cout << array[1][1] << endl;
30    cout << (*(array+1)+1) << endl; //works on dynamic/static
31    cout << *(array+1*3+1) << endl; //why will this only work with static
32
33    return 0;
34 }
```

* Create dynamic 2-d

* draw picture

Structures

user defined data type

- Data Structures So Far...
 - Variables
 - Arrays
- What if we want mixed types?
 - Record: name, age, weight, etc.
 - Use struct type

Struct/Members

making container

name
struct doc_record {

char name[50];

int age;

float weight;

};

Pieces of info

- What does this do?
- How do we use it?

Struct Type

```
struct doc_record{
    char name[50];
    int age;
    float weight;
}; //creates a user defined type, doc_record
int main() {
    doc_record garrett; //use it as a type
    ...
}
```

