

CS 161

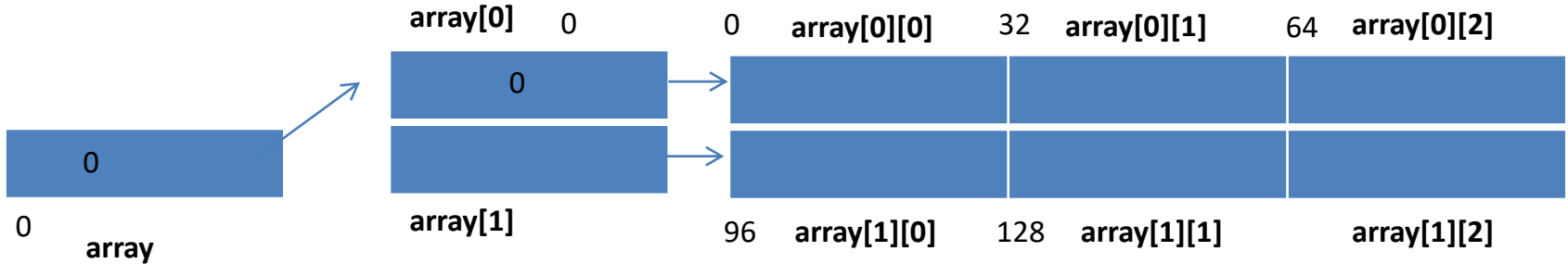
Intro to CS I

2d arrays and Command-Line
Arguments

Odds and Ends

- Demo Assignment 4
- Assignment 5 questions
- Any other questions???

Static 2-D arrays...



Demo...

Passing a 2-D Array (Static)

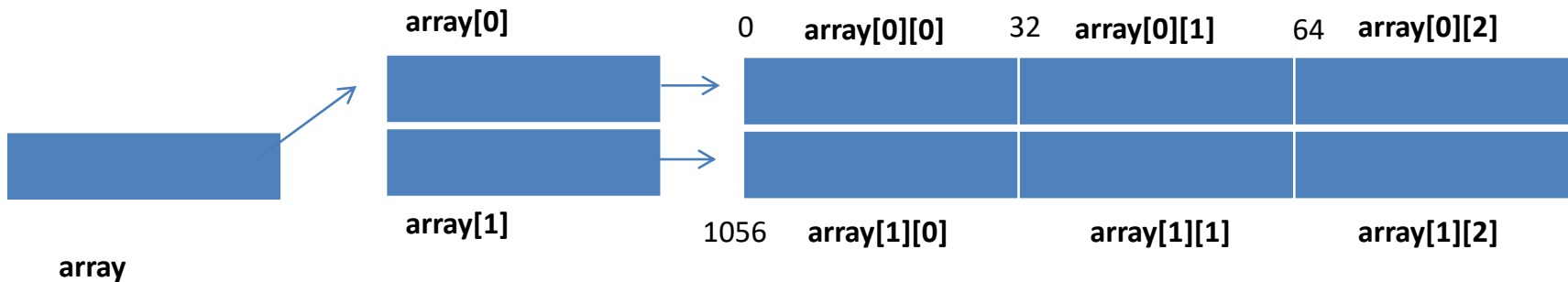
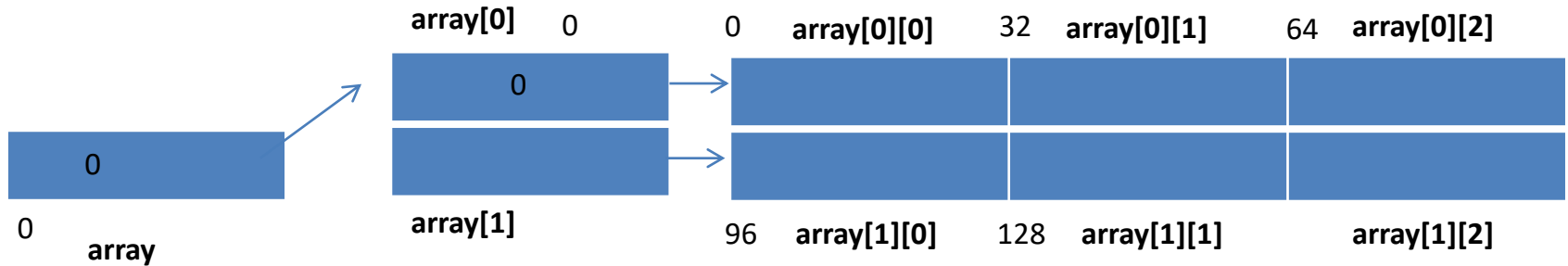
```
int main() {  
    int array[5][5];  
    ...  
    pass_2darray(array);  
    ...  
}  
  
void pass_2darray(int a[5][5]) {  
    cout << "Array at zero: " << a[0][0] << endl;  
}  
  
OR  
  
void pass_2darray(int a[][5]) {  
    cout << "Array at zero: " << a[0][0] << endl;  
}
```

Command-line Arguments

Command-line Arguments

Command-line Arguments

Static vs. Dynamic 2-D arrays...

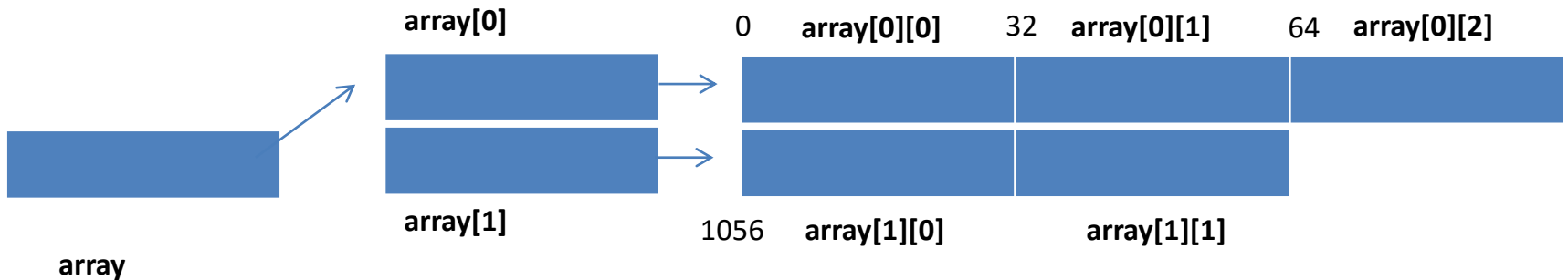


Passing a 2-D Array (Dynamic)

```
int main() {  
    int **array;  
    ...  
    pass_2darray(array);  
    ...  
}  
  
void pass_2darray(int *a[]) {  
    cout << "Array at zero: " << a[0][0] << endl;  
}  
  
OR  
  
void pass_2darray(int **a) {  
    cout << "Array at zero: " << a[0][0] << endl;  
}
```

Jagged Arrays

```
int *array[2];  
array[0] = new int[3];  
array[1] = new int[2];
```



Passing a 2-D Array (Dynamic)

```
int main() {  
    int *array[2];  
    ...  
    pass_2darray(array);  
    ...  
}  
  
void pass_2darray(int *a[]) {  
    cout << "Array at zero: " << a[0][0] << endl;  
}  
  
OR  
  
void pass_2darray(int **a) {  
    cout << "Array at zero: " << a[0][0] << endl;  
}
```

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];  
}
```

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];  
}
```

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;
```