Multidimensional Arrays

CS A150 - C++ Programming 1

Multidimensional Arrays

- **Multidimensional arrays** have ***more*** than one index
- C++ allows any number of indices
  - ***Typically*** no more than two
- As with one-dimensional array, capacity **cannot** be changed.
- Still stores as a sequence in memory.

Example: Two-dimensional Array

```
const int NUMBERS_ROWS = 11;
const int NUMBERS_COLS = 6;
...
int numbers[NUMBERS_ROWS][NUMBERS_COLS];
numbers[0][0] = 32;
numbers[0][1] = 47;
...
numbers[0][5] = 12;
numbers[1][0] = 23;
numbers[1][1] = 54;
...
numbers[10][5] = 76;
```

Filling In and Accessing Data

- To fill or access a two-dimensional array, use *nested for loops*:

```
for (int i = 0; i < NUMBERS_ROWS; ++i)
{
     for (int j = 0; j < NUMBERS_COLS; ++j)
     {
          // access value of row i, column j
          cout << numbers[i][j];
     }
}
```

Passing to Functions

- Specify the ***number of columns*** when declaring a 2D array parameter

```
void printNumbers(const int numbers[][NUMBERS_COLS], int rows)
{
	for (int i = 0; i < rows; ++i)
	{
		for (int j = 0; j < NUMBERS_COLS; ++j)
			cout << numbers[i][j] << endl;
	}
}
```

- The compiler ignores the first index, if supplied.

Example

- Project: Two-Dimensional Arrays

Questions?

(Multidimensional Arrays)