2D Arrays
(Savitch, Chapter 7.5)

TOPICS

• Multidimensional Arrays
• 2D Array Allocation
• 2D Array Initialization
• TicTacToe Game

Declaring and initializing 2D arrays

// setting up a 2D array
final int M=3, N=4;
int [][] matrix = new int [M][N];
for(int i=0; i<M; i++) {
    for (int j=0; j<N; j++) {
        matrix[i][j] = fileScanner.nextInt();
    }
}

Printing 2D arrays

// printing from a 2D array
final int M=100, N=200;
int [][] matrix = new int [M][N];
for(int i=0; i<M; i++) {
    for (int j=0; j<N; j++) {
        System.out.print(matrix[i][j] + " ");
    }
    System.out.println();
}

Adding two matrices

// setting up a 2D array
final int M=100, N=200;
int [][] m1 = new int [M][N];
int [][] m2 = new int [M][N];
// First write code to initialize the matrices m1 and m2 as an exercise
int [][] m3 = new int [M][N];
for(int i=0; i<M; i++) {
    for (int j=0; j<N; j++) {
        m3[i][j] = m1[i][j] + m2[i][j];
    }
}
More on 2D arrays

- `int[][] matrix = new int[3][4];`
- What is `matrix.length`? It is 3
- What is `matrix[0].length`? It is 4
  - So is `matrix[1].length`, `matrix[2].length`, and `matrix[3].length`
- You can access a particular row using `matrix[i]` where `i` refers to the row number between 0 and 2
- Each row is a one-dimensional array
- You cannot access a column like that 😞
- Exercises:
  - Write code that subtracts one matrix from another
  - Write code that transposes the given matrix

Review (Java)

- Assignments & expressions
- Sequential control: if & switch
- Looping control: while, for, do
- Organization: classes & methods
- Tools: Eclipse & debugging

Why? So you can program…

Programming

- … but programming isn’t about syntax
  - You can program in many languages
- Programming is about problem solving
  - Problem definition/refinement
  - Problem decomposition
  - Managing complexity

Challenge Problem

- So here is a problem to be worked through together:
  - *Write a person versus computer TicTacToe game.*
  - Player goes first, plays ‘X’, select with mouse.
  - Machine selects random, legal moves
  - Update the board after every move
  - Program detects when game is over
  - Ability to show user the current state of the board.
Decomposition

- Game board
  - State, Initialization
- Making moves
  - User: inputs integers for row and column
  - Computer: selects random, legal moves
- Game status
  - Tie Game, Player or Computer Wins, In Progress

Code

- Focusing on the game:
  - Board state needed by multiple subtasks
  - Good candidate for an class variable
  - Obvious mapping to 2D array (3 rows, 3 columns)
- Initialize the board : method
- Player moves: method
- Computer moves: method
- Detect game over : method(s)
- Print the board: method

 Constructors

- The syntax for constructors is unique
  - Constructors can take parameters, but they never return a value
  - The constructor name is always the same as the class name
  - The default constructor has no parameters, but we can add them
  - The constructor is generally used to initialize class instance variables

Eclipse Demo

- Write the program for TicTacToe
- Will be posted on the progress page