COMPSCI 105

Principles of Computer Science

Nested Loops
Multi-Dimensional Arrays
Assignment 01

Revision

Look at the code below and answer the following questions:

```
String[] skiResorts = {
       "Whistler Blackcomb", "Squaw Valley", "Brighton",
       "Snowmass", "Sun Valley", "Taos"
};
```

1. What is the index of "Brighton" in the following array?
2. What is the value of the expression `skiResorts.length`?
3. What is the index of the last item in the array?
4. What is the value of the expression `skiResorts[4]`?

Previously in 105…

Multi-dimensional Arrays

Nested loops

Any questions?

- Lab locations
- Textbook
- Web site
- Forums
- Tuition Fees due

Using Multi-dimensional arrays

Game of Life

Image Processing

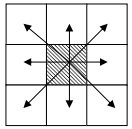
Tabular data

Sudoku

Conway's Game of Life

Mathematician John Conway

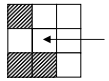
- Grid of cells
- Cell is on (alive) or off (dead)
- Simple rules dictate whether a cell is alive or dead



The neighbourhood of a cell consists of the eight other cells which surround it.

Rules for "life"

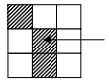
If a cell is dead, but has three live neighbours then it will become alive in the next generation



this cell will become alive in the next generation

"BIRTH"

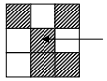
If a cell is already alive and has either two or three live neighbours then it will stay alive the next generation



this cell will stay alive in the next generation

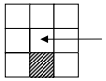
"SURVIVAL"

In all other conditions, a live cell will die in the next generation



this cell will die in the next generation

"OVERCROWDING"



this cell will remain dead in the next generation

"LONELINESS"

Code for "life"

```
boolean[][] cells;
```

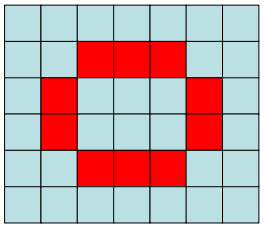
The neighbours of `cell[r][c]` are:

```
cells[r-1][c-1]
cells[r-1][c]
cells[r-1][c+1]
cells[r][c-1]
cells[r][c+1]
cells[r+1][c-1]
cells[r+1][c]
cells[r+1][c+1]
```

```
liveNeighbours = 0;
for (int i = -1; i < 2; i++) {
       for (int j = -1; j < 2; j++) {
          if (((i != 0) || (j != 0)) && cells[r+i][c+j])
                     liveNeighbours++;
       }
}
```


Image Processing

An image is really a grid of dots



This grid is often represented as an array of int values

When we manipulate the image, we apply changes to a 2-d array

Simple Filters

We can alter the colour of a single dot, based on it's immediate neighbours

This is known as a simple filter. It can be used to blur or sharpen an image, and can be used in simple edge detection algorithms

0	0	0
0	1	0
0	0	0

We multiply the value of the neighbour by the value in the corresponding filter cell.

Sum all the values and use them for the value of the center pixel.

Example - Ragged Array

Imagine that you are collecting data about noise pollution around a particular location. You have given surveys to a number of people who have been asked to make a note of the time whenever they hear a loud noise.

The table of data might look like the following:

Abe	8	9	10	11	12	1	2
Bart	12						
Carl	9	2	5				
Dolph	10	4					
Edna	7	8	10	2	3		

We can store this data efficiently using a two-dimensional array.

Example Code

Ask the user the number of people

Create a the correct number of rows

For each person

- Ask the user how many entries
- Create the correct number of columns for that person
- Enter the data for each column

```
int[][] data;

int numberOfPeople = getNumberOfPeople();
data = new int[numberOfPeople][];

for (int n = 0; n < numberOfPeople; n++) {
        System.out.println("How many times did you record noise?");
        int noise = getNumberOfEntries();
        data[n] = new int[noise];

        for (int e = 0; e < noise; e++) {
                System.out.println("Enter the time:");
                int time = getTime();
                data[n][e] = time;
        }
}
```


Sudoku

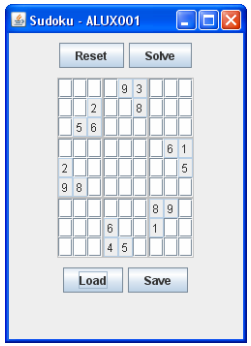
Fill in the grid so that every row, every column, and every 3x3 box contains the digits 1 through 9.

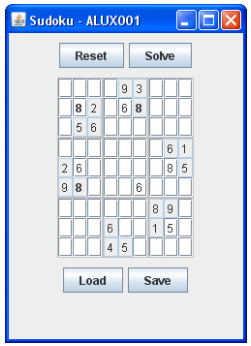
	6		1		4		5	
		8	3		5	6		
2								1
8			4		7			6
		6				3		
7			9		1			4
5								2
		7	2		6	9		
	4		5		8		7	

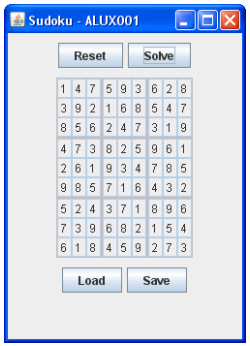
Sudoku Application

Features

- load or save a puzzle
- clear the grid
- find a solution to the puzzle
- invalid moves are highlighted (bold)







Assignment One

Complete a version that has a reduced set of features

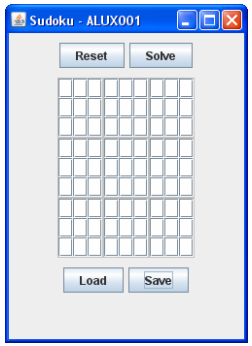
- Reset should clear the puzzle
- Save should print out a text representation to std. output
- Invalid moves should be highlighted (bold)

User Interface

- Consists of many classes
- Some advanced features
- Does not have to be altered

SudokuBoard class

- You have to implement this class
- Contains the data for the puzzle
- Uses a two-dimensional array



Questions?

Assignment Two

Implement the other features

- Ability to load
- Ability to save
- Solve the puzzle using recursion

Processing two-dimensional arrays

Use a nested loop to access each element

Declare and create the array

```
int[][] nums = new int[3][4];
```

Initialise each element

```
for (int i = 0; i < 3; i++) {
       for (int j = 0; j < 4; j++) {
              nums[i][j] = 100;
       }
}
```


Processing two-dimensional arrays

Declare and create the array

```
int[][] nums = new int[3][4];
```

Sum the elements in the array

```
int sum = 0;
for (int i = 0; i < 3; i++) {
       for (int j = 0; j < 4; j++) {
              sum += nums[i][j];
       }
}
```

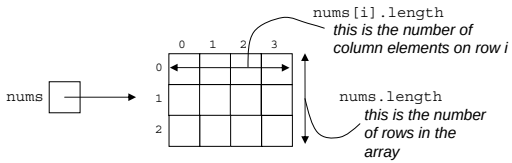

Style: Eliminate dependencies

Number of rows

- `nums.length`

Number of columns

- `nums[i].length`




```
int sum = 0;
for (int i = 0; i < nums.length; i++) {
       for (int j = 0; j < nums[i].length; j++) {
              sum += nums[i][j];
       }
}
```


Exercise

Given a two-dimensional array of integer values, write a method called printArray that will take a 2d array of integers as a parameter and print out each row of the array as a single line of output.

For example, the code:

```
int[][] a = { {1, -1, 1} , {2, 6, 21, 6} };
printArray(a);
```

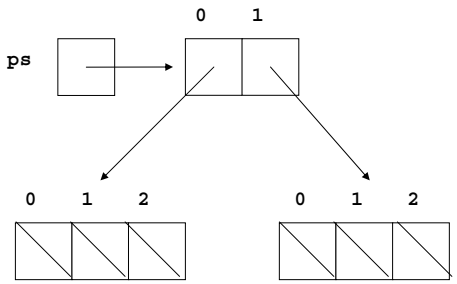
Would result in the output:

```
1 -1 1
2 6 21 6
```

Creating an array of objects

Syntax for declaring and creating arrays of objects is the same as arrays of primitive types

```
Point[][] ps = new Point[2][3];
```

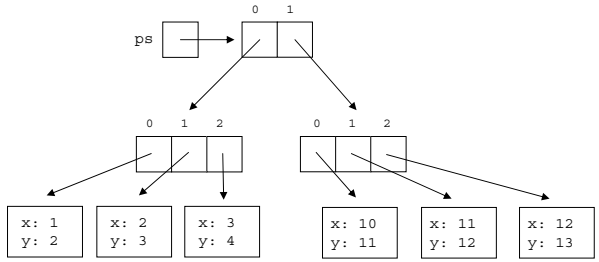


Creating each object

Each element in the array is `null`

- Must create the object referred to by each element in the array

```
ps[0][0] = new Point(1, 2);
ps[0][1] = new Point(2, 3);
ps[0][2] = new Point(3, 4);
ps[1][0] = new Point(10, 11);
ps[1][1] = new Point(11, 12);
ps[1][2] = new Point(12, 13);
```



Exercise

Explain what this code does. In other words, what is the purpose of this code?

```
public int simpleMethod(int[][] numbers) {
	int result = 0;
	for (int i = 0; i < numbers.length; i++) {
		for (int j = 0; j < numbers[i].length; j++) {
			if (numbers[i][j] % 2 == 0) {
				result++;
			}
		}
	}
	return result;
}
```


Nested Loops

Code tracing techniques

- Write down the value of each variable
- Cross the value out when it changes

```
int SIZE = 6;
for (int i = 0; i < 6; i++) {
	for (int j = 0; j < i; j++) {
		if (j == 0 || j == i-1 || i == SIZE-1)
			System.out.print("*");
		else
			System.out.print("-");
	}
	System.out.println();
}
```

Exercise

Write a method that will print a hollow square to standard output. The size of the square is given as an integer parameter.

Size 5

```
*****
*   *
*   *
*   *
*****
```

Size 3

```
***
* *
***
```


Exercise

Write a method called `shortestString()` that accepts an array of Strings as a parameter. The method should return the shortest String contained within the array.

Exercise

The number of tickets sold to a given event have been recorded in a table as follows:

	Mon	Tues	Wed	Thurs	Fri
9					
10					
11					
12					
1					
2					
3					
4					
5					

The data from this table has been stored in a two-dimensional array. Write a program that will print out the day in which the most tickets were sold.

```
int[][] data = new int[5][9];
```