CS 105

Week 15

Final Exam

Thursday, December 17

1:30pm – 4:30pm

TA Review Session

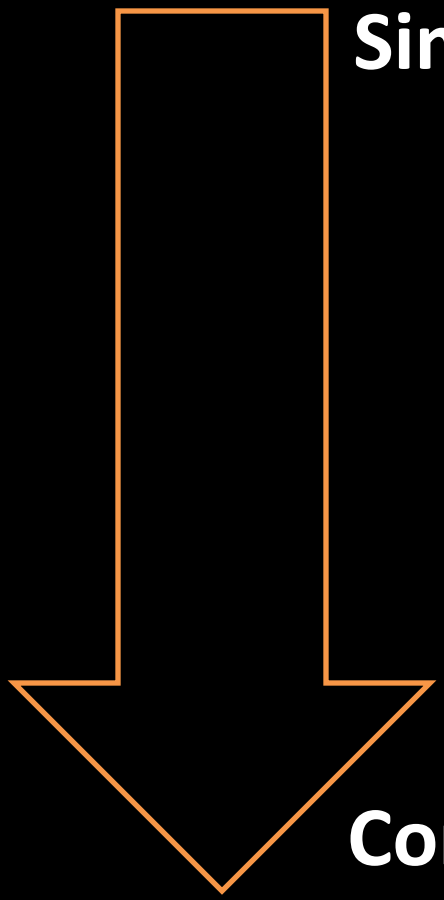
What’s your preference?

TA Review Session

A) Thursday (Reading Day)
B) Saturday
C) Sunday
D) Tuesday (Two days before exam)
E) Wednesday (Day before exam)

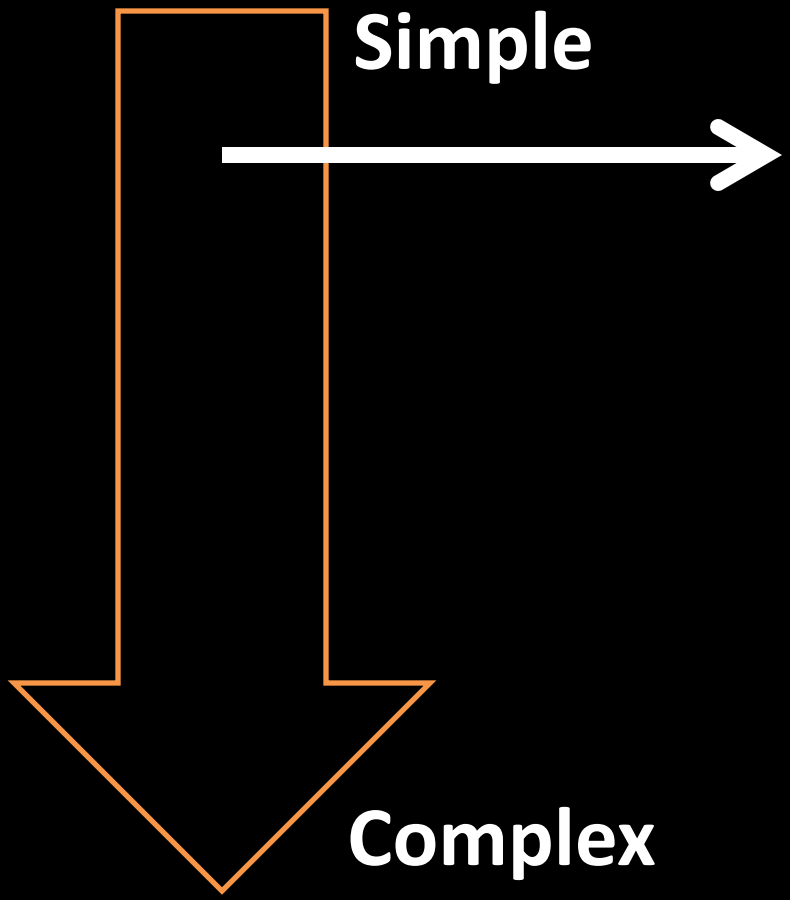
CS 105: Data Processing

You have learned a lot about data processing:



CS 105: Data Processing

You have learned a lot about data processing:



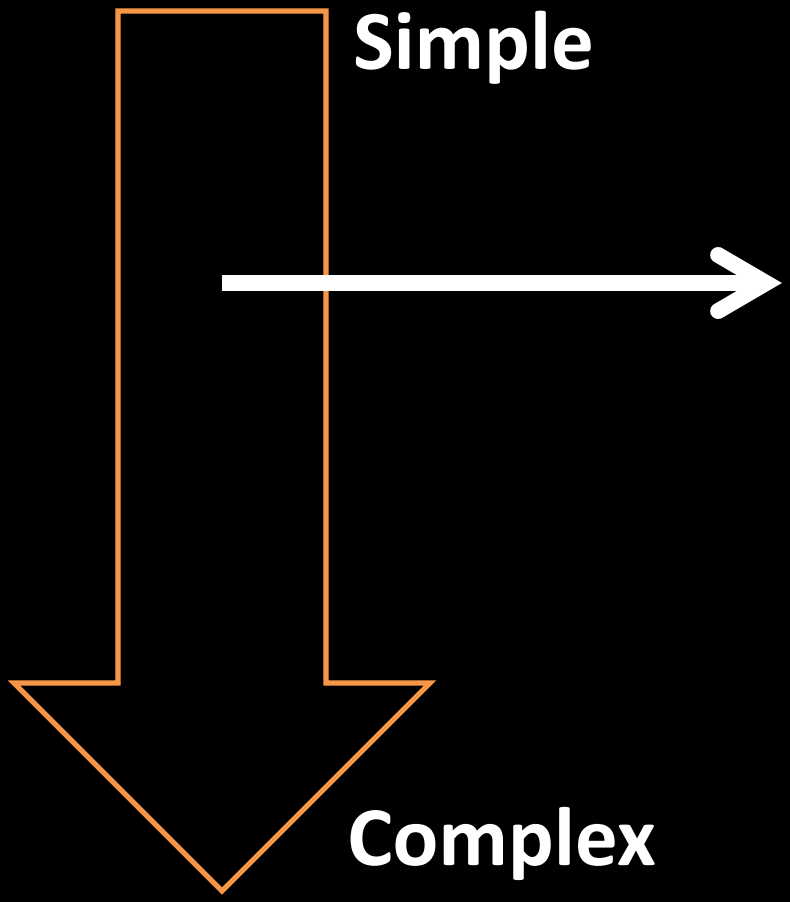
Tool: Calculator

Good Until:

~100 data points

CS 105: Data Processing

You have learned a lot about data processing:



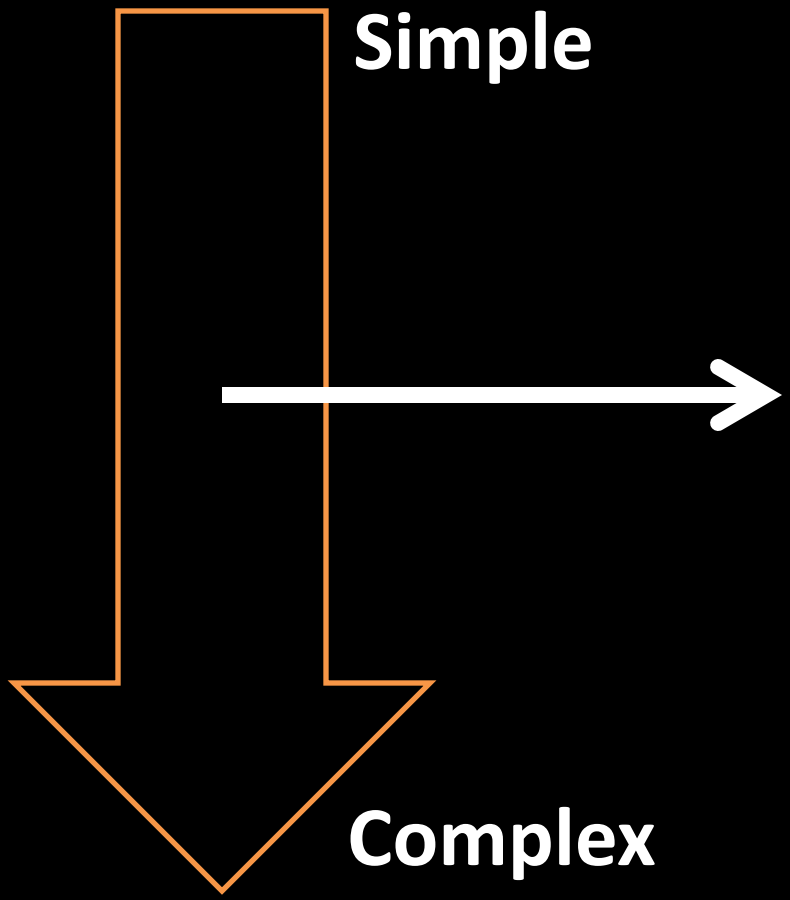
Tool: Excel

Good Until:

~1m data points

CS 105: Data Processing

You have learned a lot about data processing:



Tool:

Computer Scripts

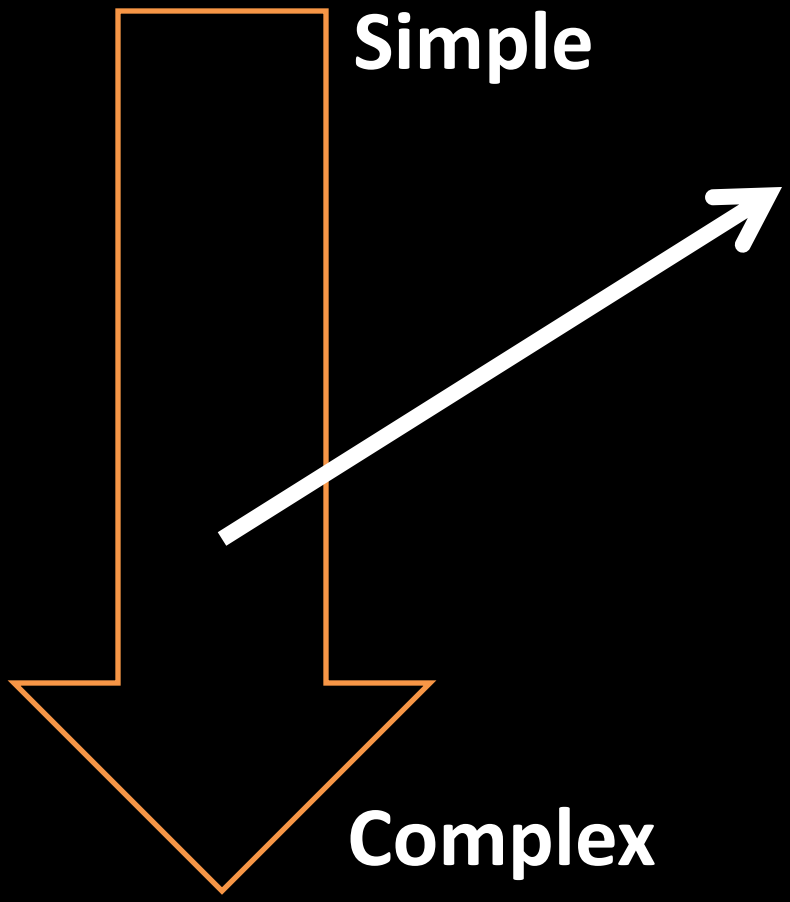
(eg: JavaScript)

Good Until:

~1b data points

CS 105: Data Processing

You have learned a lot about data processing:



Tool:

"Big Data" Tools

Good Until:

>1t data points

Data Visualization

Your data presentation is sometimes as important or more important than your analysis!

Data Visualization

Excel: Great tool for simple visualizations
(Used by everyone, all fields, all the time)

Limited to pre-determined visualizations.

Data Visualization

d3.js is a JavaScript visualization library.

Significantly more complex to create, limitless in possibilities.

JavaScript Libraries

A **JavaScript library** is JavaScript code that:

JavaScript Libraries

A **JavaScript library** is JavaScript code that:

1. *Adds new functionality to JavaScript*

JavaScript Libraries

A **JavaScript library** is JavaScript code that:

1. *Adds new functionality to JavaScript*
2. *Able to be imported into another project*

JavaScript Libraries

A **JavaScript library** is JavaScript code that:

1. *Adds new functionality to JavaScript*
2. *Able to be imported into another project*
3. *Documented for others to use*

d3.js

Every d3.js has three majors components:

1. *Data*

d3.js

Every d3.js has three majors components:

1. *Data*
2. *Boilerplate Code*

d3.js

Every d3.js has three majors components:

1. *Data*
2. *Boilerplate Code*
3. *Drawing the Visualization*

Component 1: Data

```
var games = [
  { score:[4, 1], opponent: "Oakland" },
  { score:[1, 0], opponent:
                    "Illinois State" },
  { score:[5, 2], opponent: "TCU" },
  ...
];
```

Component 2: Boilerplate Code

```
var margin = { top: 50, right: 0,
               bottom: 100, left: 150 },
    width = 3000 - margin.left - margin.right,
    height = 3200 - margin.top - margin.bottom;

var svg = d3.select("#chart")
            .append("svg")
            .attr("width", width + margin.left
                           + margin.right)
            .attr("height", height + margin.top
                            + margin.bottom)
```

Component 3: Visual Components

```
svg.selectAll()
    .data( games )
    .enter()
    .append("rect")
    .attr("x", function (d, i) {
      return d.score[0];
    })
    .attr("y", function (d, i) {
      return d.score[1];
    })
```

Component 3: Visual Components

```
svg.selectAll()
    .data( games )
    .enter()
```

Component 3: Visual Components

```
svg.selectAll()
    .data( games )
    .enter()
```

Component 3: Visual Components

```
.append("rect")
```

Component 3: Visual Components

```
.append("rect")
```

Component 3: Visual Components

```
.attr("x", function (d, i) {
  return d.score[0];
})
```

Component 3: Visual Components

```
.attr("x", function (d, i) {
   return d.score[0];
})
```

Component 3: Visual Components

```
.attr("x", function (d, i) {
   return d.score[0];
})
```

Component 3: Visual Components

```
.attr("y", function (d, i) {
   return d.score[1];
})
```

Component 3: Visual Components

```
  .attr("y", function (d, i) {
    return d.score[1];
  })

// d is an element of data:
d == { score: [4, 1],
       opponent: "Oakland" }
```

d3.js shape attributes

Rectangles have four core attributes:

1. *width*
2. *height*
3. *x*
4. *y*

d3.js shape attributes

Circles have four core attributes:

1. *cx* *(x-coordinate of center)*
2. *cy* *(y-coordinate of center)*
3. *r* *(radius)*

d3.js shape attributes

All shapes can have styles applied to them:

1. *fill* *(fill color of shape, a string)*
2. *stroke* *(outline of shape, a string)*

Extra Credit MP: MPx

A d3.js-based MP will be released later today, for extra credit!

(Can replace any single MP or +5 points)

ICES Forms

Wade Fagen

Lectures: 0-6, 15

Martin Hellwig

Lectures: 8-12, 14