Learn more about our research, discover data science, and find other great resources at:

http://www.dataminingapps.com
Chapter 1
A General Introduction to Programming
Overview

• The programming process
• OO programming: a sneak preview
• Programming errors
• Principles of software testing
• Software maintenance
• Principles of structured programming
The programming process

• A program (aka application) is a set of instructions to solve a particular problem
• Programming is the activity of writing a program
• Example: BMI
The programming process

- Requirements
- Design
- Coding
- Deployment
- Maintenance
- Testing
- Translation
The programming process

- Algorithm is a procedure needed to solve the problem
- Can be designed using pseudo-code or flowcharts
  - Pseudo code: structured English without strict grammar rules
  - Flowchart: represents the algorithm in a visual diagram
The programming process

ask user: height
ask user: weight
if height = 0 or weight = 0:
  error: "Incorrect input values"
return to beginning (ask height and weight)
end if
x = weight / (height * height)
message: "Your BMI is ",x
# The programming process

<table>
<thead>
<tr>
<th>FLOWCHART SYMBOL</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Terminator" /></td>
<td>A terminator shows the start and stopping points of the program.</td>
</tr>
<tr>
<td><img src="image" alt="Arrow" /></td>
<td>An arrow shows the direction of the process flow.</td>
</tr>
<tr>
<td><img src="image" alt="Rectangle" /></td>
<td>A rectangle represents a process step or activity.</td>
</tr>
<tr>
<td><img src="image" alt="Diamond" /></td>
<td>A diamond indicates a decision point in the process.</td>
</tr>
<tr>
<td><img src="image" alt="Document" /></td>
<td>This symbol represents a document or report.</td>
</tr>
<tr>
<td><img src="image" alt="Rhombus" /></td>
<td>This rhombus represents data used as inputs/outputs to/from a process.</td>
</tr>
<tr>
<td><img src="image" alt="Cylinder" /></td>
<td>This cylinder represents a database.</td>
</tr>
</tbody>
</table>
The programming process

- Start
- Ask user height and weight
  - weight = 0 or height = 0
    - Yes: Display error message
    - No: Compute BMI
- Display BMI
- Stop
public class BMICalculator {
    private double weight, height, BMI;

    public BMICalculator( double weight, double height ) {
        this.weight = weight;
        this.height = height;
    }

    public void calculate() {
        BMI = weight / (height * height);
    }

    public boolean isOverweight() {
        return (BMI > 25);
    }
}
Programming Errors

• Also referred to as a bug (debugging)

• Debugging steps
  – Detect that there is an error
  – Locate the error
  – Solve the error

• Types of bugs
  – Syntax/Compilation errors
  – Runtime errors
  – Logic/Semantic errors
public void calculate()
{
    BMI = weight / (height*height),
}

Syntactic/Compilation Error!
public void calculate(){
    BMI = weight / (height*height);
}

• Runtime error if 0 entered for height!
public void calculate()
{
    BMI = (weight*weight) / height;
}

Logic/Semantic error!
Principles of Software Testing

• Verification versus Validation
Principles of Software Testing

• Desk check the program manually
• Static testing
  – Inspect and review the code
  – Detailed walk-throughs
• Dynamic testing
  – Execute with selected test cases
• White box strategy
  – Test cases selected based upon code inspection
• Black box strategy
  – Test cases selected not based upon code inspection
• Alpha versus beta testing
Software maintenance

• Adjusting the program after it was taken into production

• Adaptive maintenance
  – Modify program to accommodate changes in the environment
  – E.g., new Windows release

• Perfective maintenance
  – Support new or changed user requirements
  – E.g. enter height in feet units and weight in pound units

• Corrective maintenance
  – Fix runtime errors (emergency fixes vs. routine debugging)

• Preventive maintenance
  – Prevent future errors
  – E.g. Y2K, Euro
Principles of Structured Programming

• Stepwise refinement
• Documentation
• Meaningful names
Conclusion

- The programming process
- OO programming: a sneak preview
- Programming errors
- Principles of software testing
- Software maintenance
- Principles of structured programming