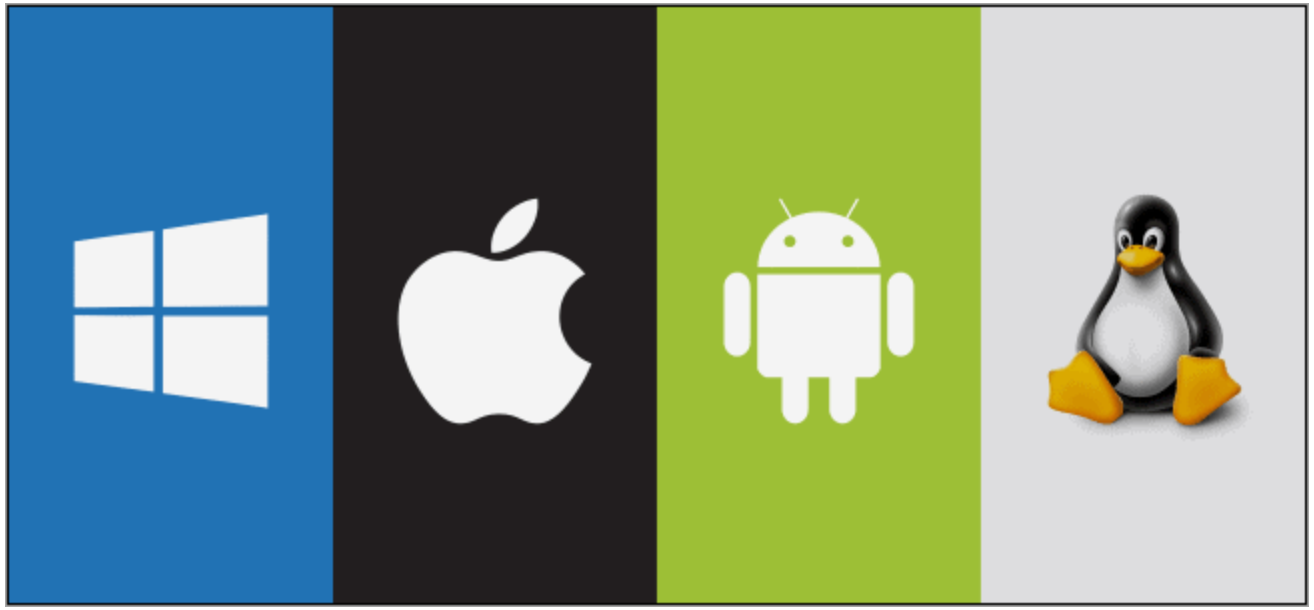
Why We Are Teaching This Class

The Gap in Traditional CS Education

- Advanced Topics: Operating Systems

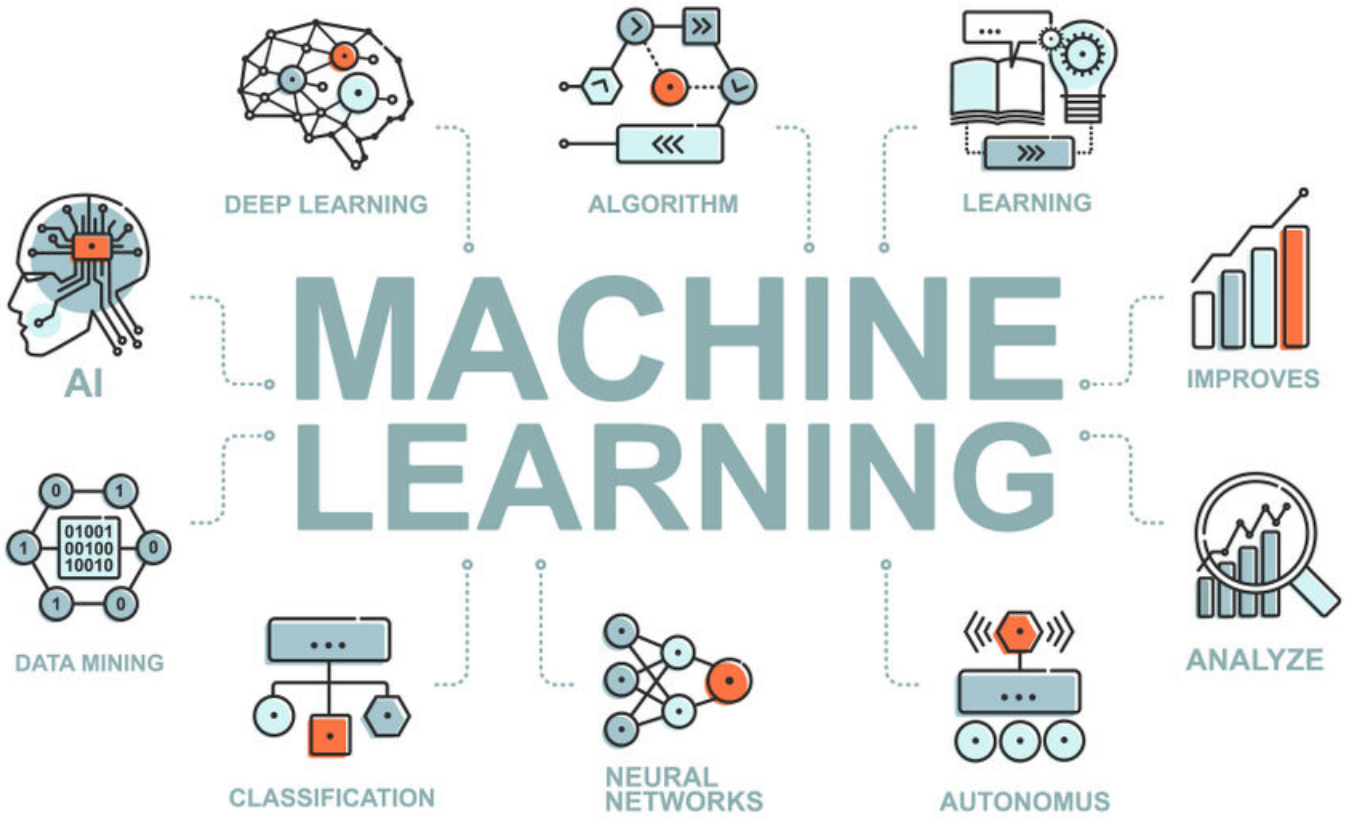


The Gap in Traditional CS Education

- **Advanced Topics**: Operating Systems, **Programming Languages**

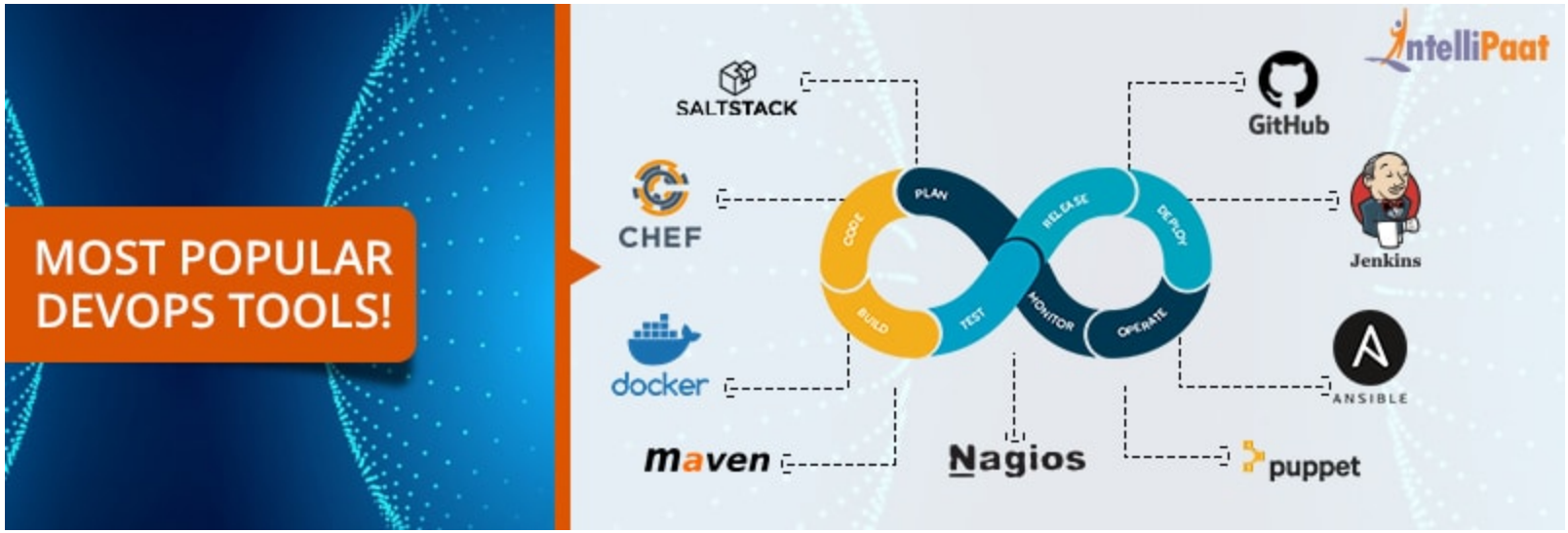
The Gap in Traditional CS Education

- **Advanced Topics**: Operating Systems, Programming Languages, **Machine Learning**



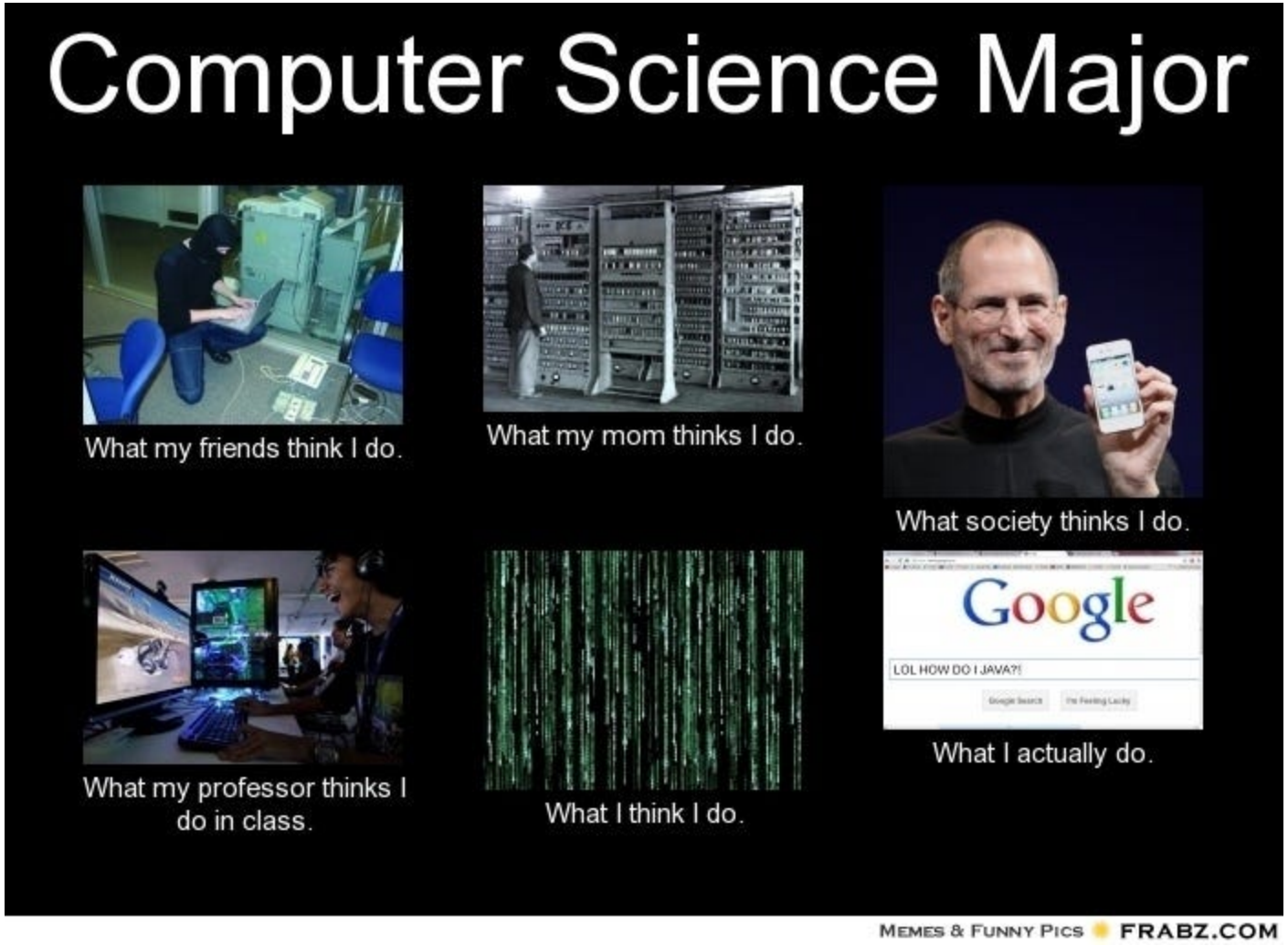
The Gap in Traditional CS Education

- **Advanced Topics**: Operating Systems, Programming Languages, Machine Learning
- **Rarely Covered**: **Computing ecosystem literacy**



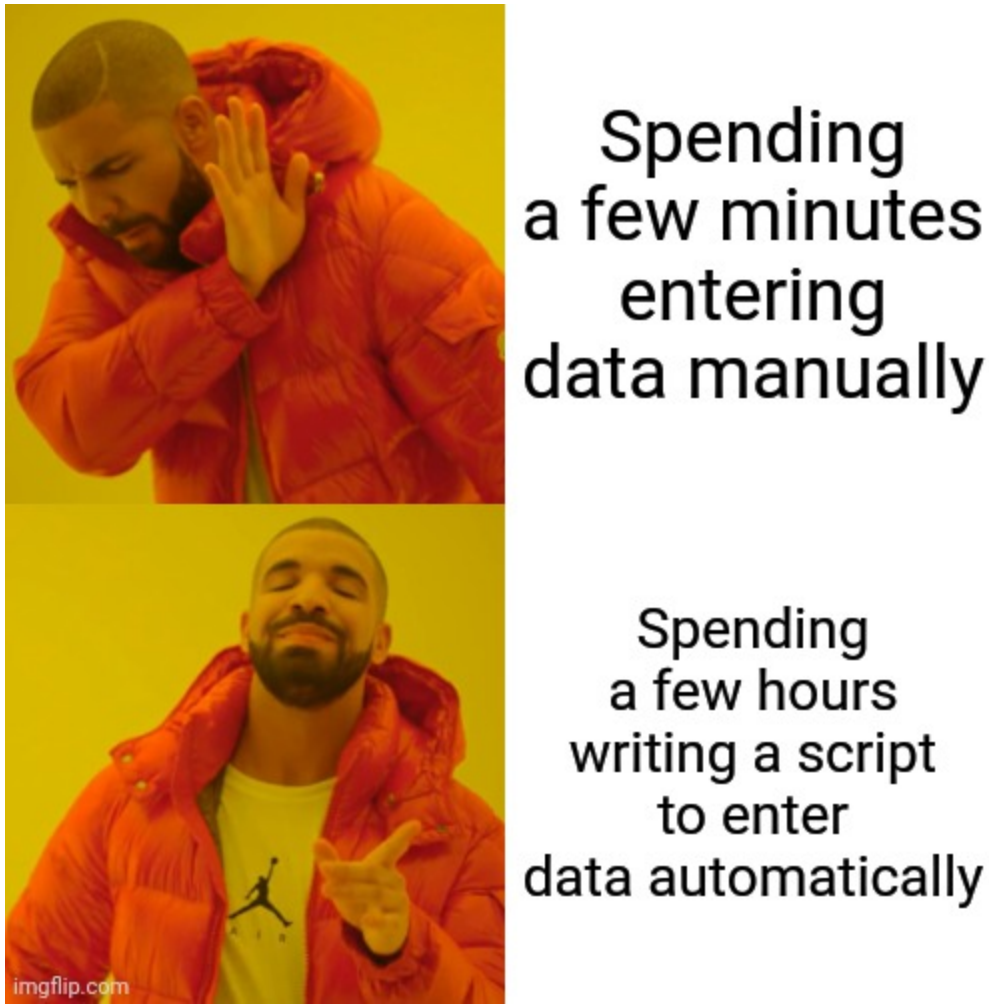
The Problem

- **Observation at SUSTech**: Students have limited knowledge of essential tools.



The Problem

- **Observation at SUSTech**: Students have limited knowledge of essential tools.
- **Computers' Purpose**: Automate manual tasks.



The Problem

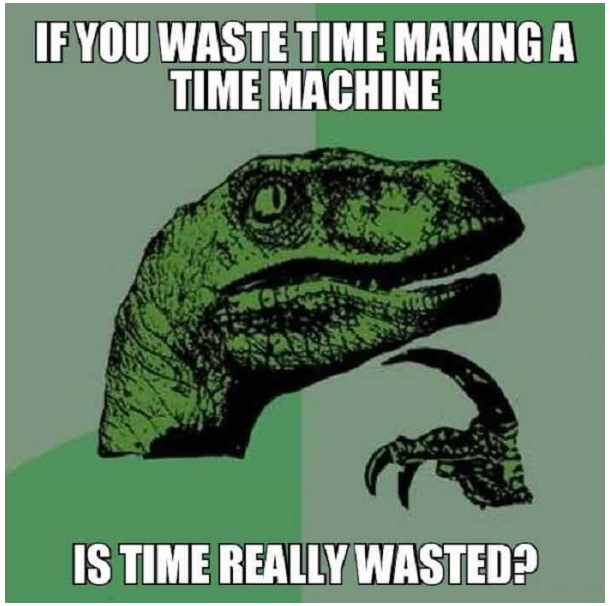
- **Observation at SUSTech**: Students have limited knowledge of essential tools.
- **Computers' Purpose**: Automate manual tasks.
- **Reality**: Students perform repetitive tasks manually.

The Problem

- **Observation at SUSTech**: Students have limited knowledge of essential tools.
- **Computers' Purpose**: Automate manual tasks.
- **Reality**: Students perform repetitive tasks manually.
- **Consequences**:
  - Inefficiencies and wasted time.

The Problem

- **Observation at SUSTech**: Students have limited knowledge of essential tools.
- **Computers' Purpose**: Automate manual tasks.
- **Reality**: Students perform repetitive tasks manually.
- **Consequences**:
  - Inefficiencies and wasted time.
  - Severe issues like data loss.



What's Missing

- **University Curriculum**: Lacks teaching on powerful tools like version control and text editors.
- **Result**: Time and effort wasted on simple tasks.
- **Critical Topics Omitted**: Computing ecosystem essentials for an easier CS life.

The Missing Semester of Your CS Education

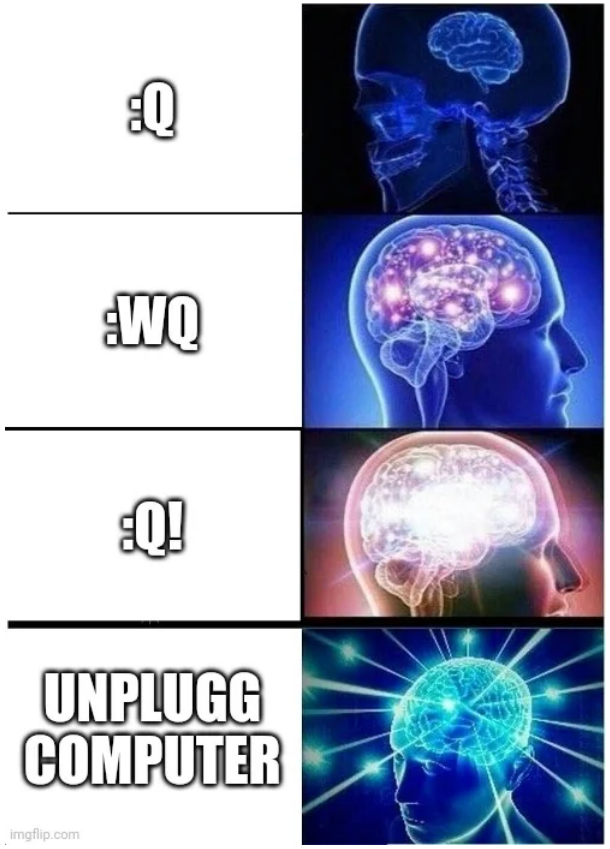
- **Our Solution**: A class covering crucial practical skills.
- **Hands-On**: Immediate application of tools and techniques.
- **Original By**: MIT's Independent Activities Period, January 2020. [LINK]
- **Accessibility**: Materials and lectures available to the public.

Concrete Examples of Class Content

- **Concrete Examples**:
  - **Version Control**: How to use `git` efficiently.

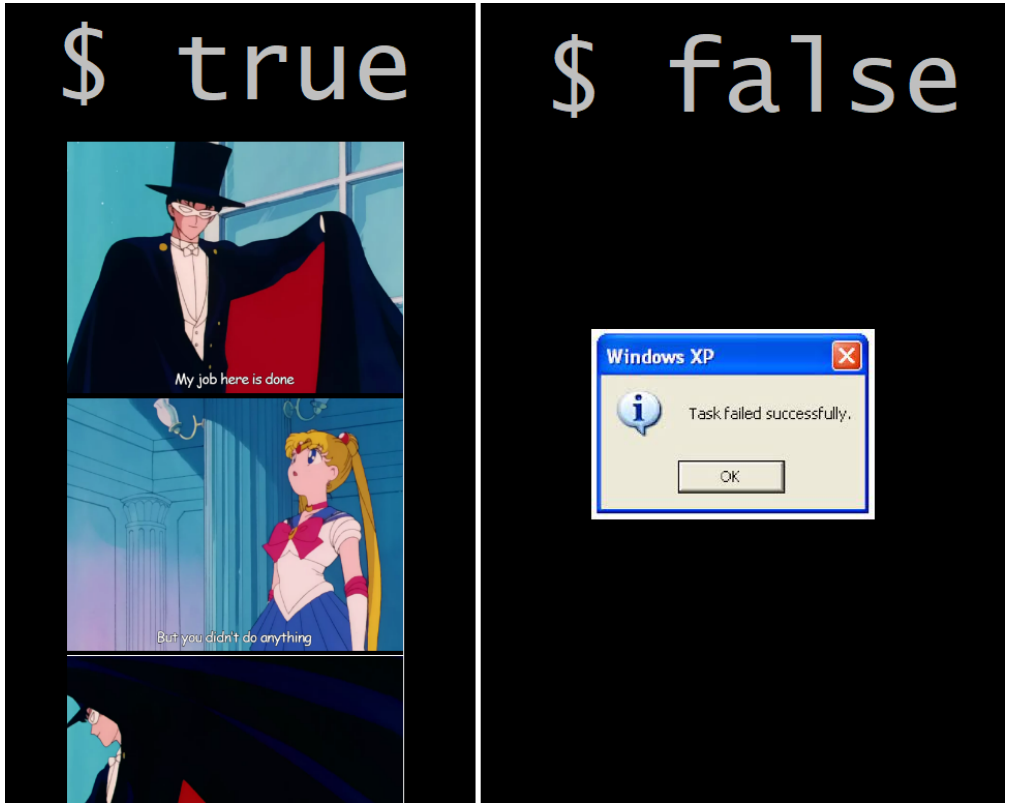
Concrete Examples of Class Content

- **Concrete Examples**:
  - **Version Control**: How to use `git` efficiently.
  - **Text Editors**: Mastery of `vim` or `emacs`.



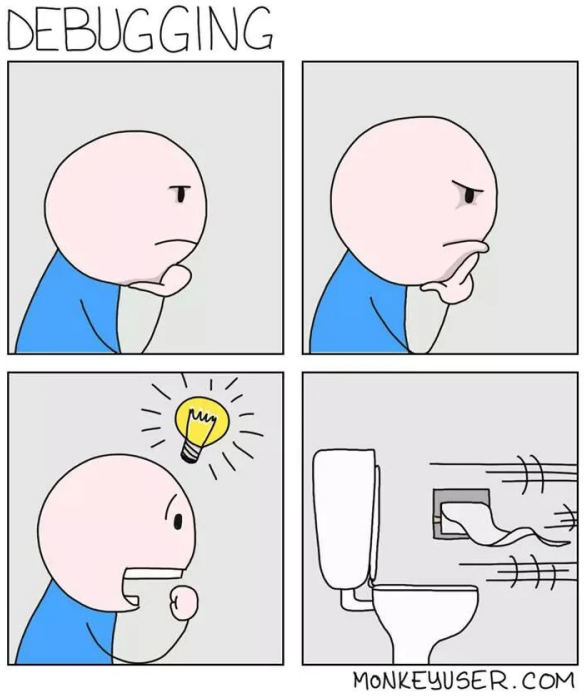
Concrete Examples of Class Content

- **Concrete Examples**:
  - **Version Control**: How to use `git` efficiently.
  - **Text Editors**: Mastery of `vim` or `emacs`.
  - **Command Line Usage**: Automate tasks with shell scripting.



Concrete Examples of Class Content

- **Concrete Examples**:
  - **Version Control**: How to use `git` efficiently.
  - **Text Editors**: Mastery of `vim` or `emacs`.
  - **Command Line Usage**: Automate tasks with shell scripting.
  - **And More**: Debugging, profiling, and more.



Conclusion

- **Empowerment**: Equip yourself with the knowledge to be more effective.
- **Efficiency**: Save time, avoid frustration, and focus on what truly matters in your CS education.
- **Fun**: Enjoy the power of computing.