Paradigms of Programming I:
Imperative Computation

COMP206 Course Outline

Imperative Program

- Step by step detailed instructions.
- Focus on **how** to get a task done.
- **why** the program works may need some explanation.

Part 2 of 3: Brewing Methods

1. Remove the lid and plunger.
2. Add the **coffee**. ...
3. Pour hot water to the halfway mark, wetting all the grounds.
4. After one minute, stir the floating grounds gently. ...
5. After three additional minutes, press slowly down until the plunger hits bottom. ...
6. Pour into a cup.

www.wikihow.com › ... › Drinks › Coffee

How to Make Perfect Coffee: 14 Steps (with Pictures) - wikiHow

Learning Outcomes

- Programming
- Data Structures
- Analyzing Codes (correctness, speed etc)
- Computational Thinking

Programming

- C
- Linux Shell
- Turning high level algorithms into code

Algorithmic Ideas

- How fast does a program run?
- Is an idea worth translating into a code?
- Data Structures aka organizing stuff.

Computational Thinking

- Approaches to solve a problem.
- Finding correct solutions.
- Making correct solutions faster.

Grading

- 4 written Assignments, 25%
- weekly programming assignments, 20%
- timed quiz/tests, 20%
- class participation, 5%
- Final exams, 30%

About Assignments

- Late submission: 1 day late 50%, 2 days late zero.
- Plagiarism: Zero on that assignment.
- Not following submission instructions: not graded, will be treated as 1 day late submission.

Online resources

- Keep watching the canvas page.

- Principles of Imperative Computation at CMU
  http://www.cs.cmu.edu/~15122/