

Class 5 - Notes

From [President Obama's Weekly Address](#), 30 January 2016:

Today's auto mechanics aren't just sliding under cars to change the oil; they're working on machines that run on as many as 100 million lines of code. That's 100 times more than the Space Shuttle. Nurses are analyzing data and managing electronic health records. Machinists are writing computer programs. And workers of all kinds need to be able to [figure out how to break a big problem into smaller pieces and identify the right steps to solve it](#).

In the new economy, computer science isn't an optional skill – it's a basic skill, right along with the three "Rs."

Communications in cs1120

Course Calendar. Please incorporate the course calendar into your calendar. If you are using Google calendar, this is easy: (1) Visit <https://goo.gl/bI7zZd>, (2) click the "+" link at the bottom right to add to your calendar.

Slack. Please use the Slack group for asking questions that don't need to be private. Your classmates and the course staff will be able to see the questions there, and answer them for everyone. Nine of you still have names like **rtq4ma** in Slack. It is hard to have a good community when people have unpronounceable names. Please set your name to something else (doesn't have to be your real name), following the directions on [Class 4 Notes](#); pictures are optional but also appreciated!).

Notes and Questions

Rules of Evaluation are precise rules for determining the meaning of *every* surface form in the language. Given that the Python language has infinitely many different surface forms, how can we define the meaning of every one?

What is the meaning of `False`?

Here is the Python grammar for an integer literal:

(slightly simplified from https://docs.python.org/3/reference/lexical_analysis.html#literals)

DecimalInteger ::= *NonZeroDigit Digits*

Digits ::= *Digits Digit*

Digits ::= ϵ

Digit ::= **0** | *NonZeroDigit*

NonZeroDigit ::= **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**

What evaluation rules do we need to know the meaning of every *DecimalInteger*?

How would the evaluation rules be different if we change the *Digits* rule to be:

Digits ::= *Digits Digit*

What is the meaning of `abs`?

What is the meaning of `abs(-2)`? (What rules are needed to obtain it?)