# Computational Assignment System Design

No/minimal packages/downloads

**Working groups**

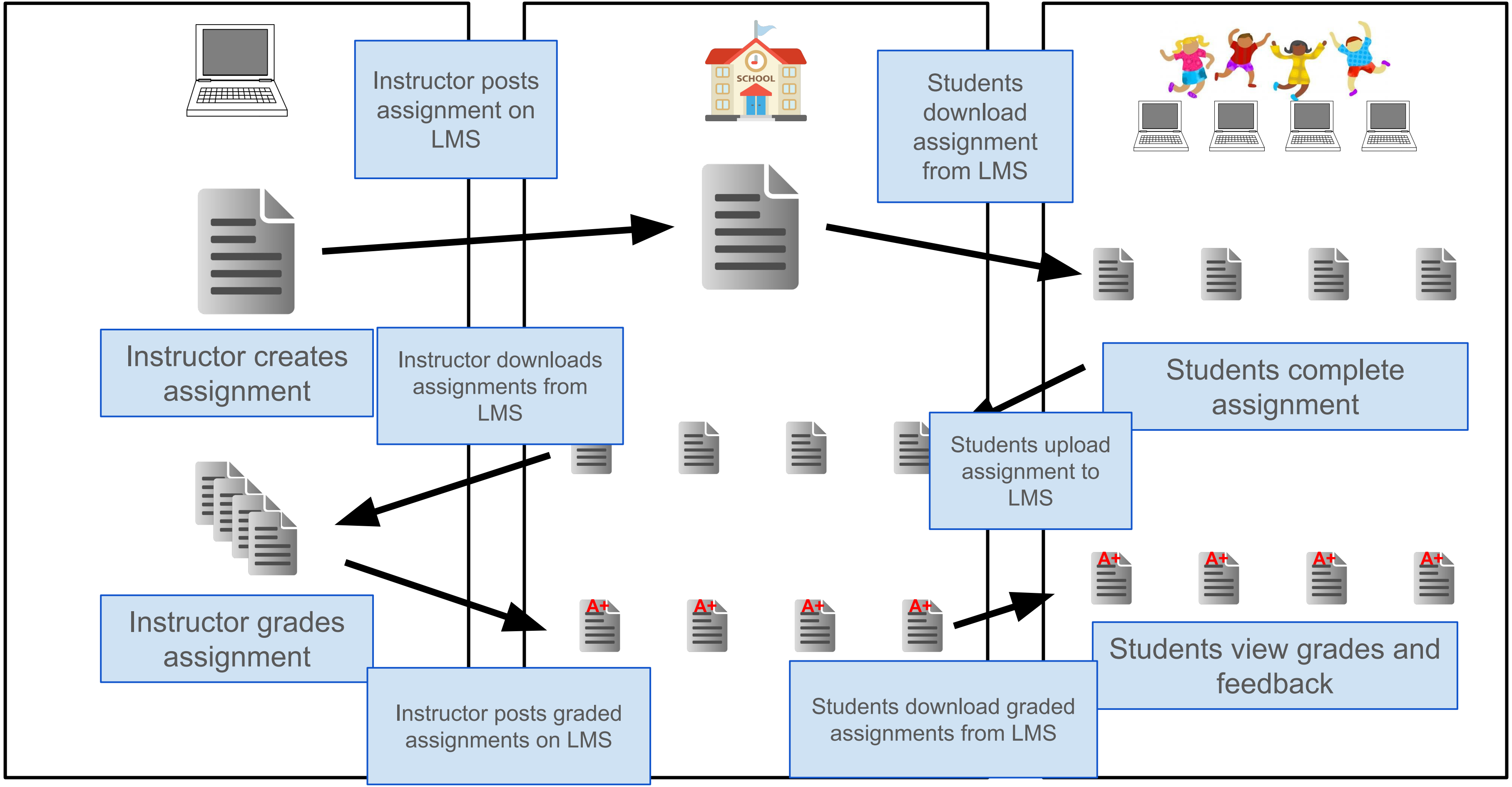
Subject groups

LMS groups

# Agenda

1. What goes into a computational assignment system?
2. Group work: fill in the details of your own workflow
3. Grading and feedback processes
4. Group work: practice grading

# The Life Cycle of an Assignment



Instructor creates assignment

Instructor posts assignment on LMS

Students download assignment from LMS

Students complete assignment

Students upload assignment to LMS

Instructor downloads assignments from LMS

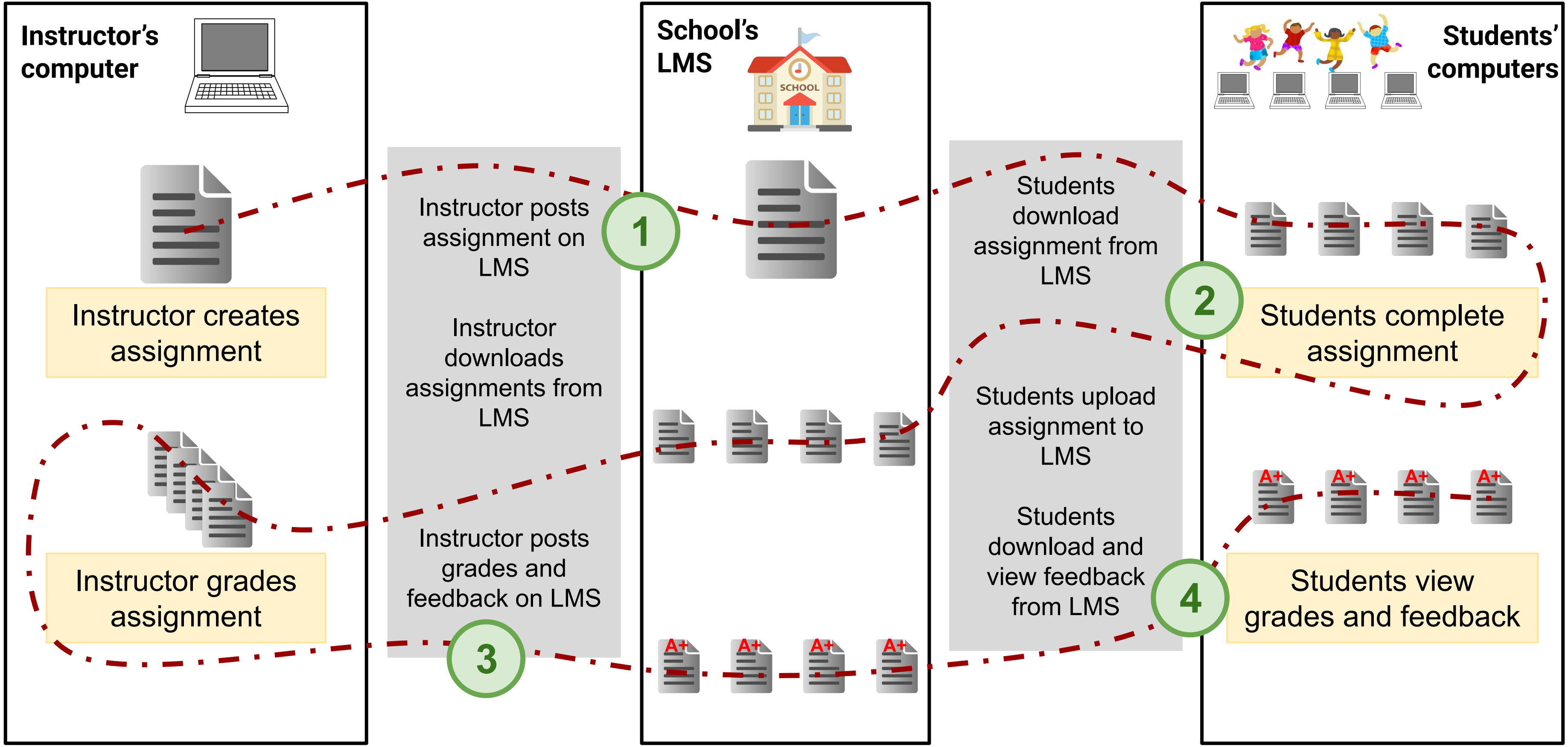
Instructor grades assignment

Instructor posts graded assignments on LMS

Students download graded assignments from LMS

Students view grades and feedback

# The Path of an Assignment



Instructor's computer

School's LMS

Students' computers

Instructor creates assignment

1

Instructor posts assignment on LMS

Students download assignment from LMS

2

Students complete assignment

Students upload assignment to LMS

Instructor downloads assignments from LMS

Instructor grades assignment

Instructor posts grades and feedback on LMS

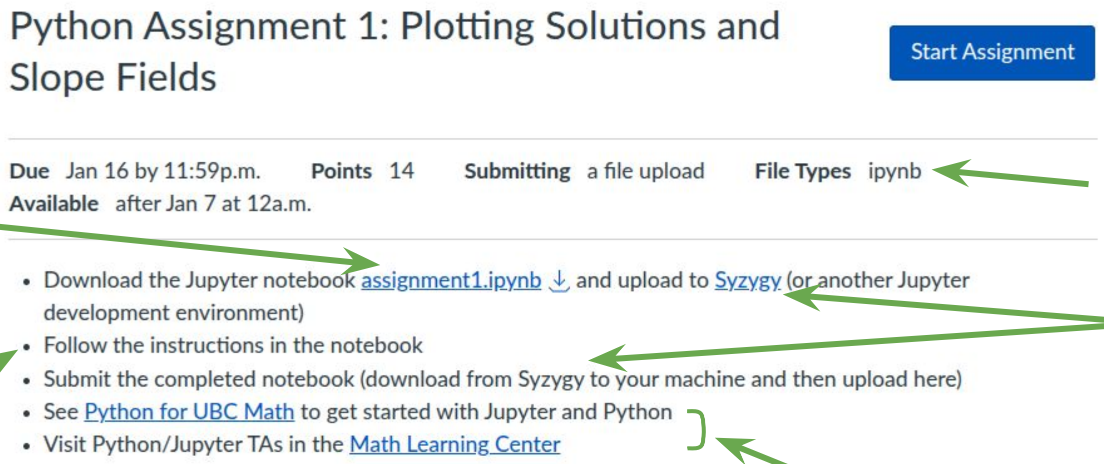
3

Students download and view feedback from LMS

4

Students view grades and feedback

# 1. Posting the assignment



## Python Assignment 1: Plotting Solutions and Slope Fields

**Due** Jan 16 by 11:59p.m. **Points** 14 **Submitting** a file upload **File Types** ipynb
**Available** after Jan 7 at 12a.m.

- Download the Jupyter notebook assignment1.ipynb and upload to Syzygy (or another Jupyter development environment)
- Follow the instructions in the notebook
- Submit the completed notebook (download from Syzygy to your machine and then upload here)
- See Python for UBC Math to get started with Jupyter and Python
- Visit Python/Jupyter TAs in the Math Learning Center

Assignment file link to download

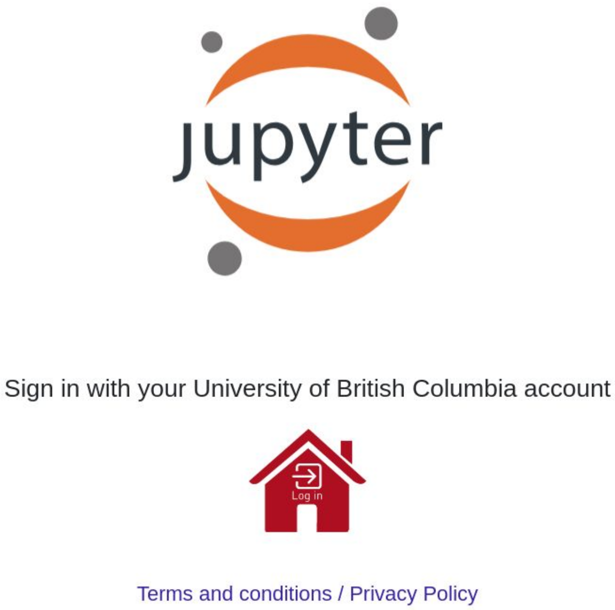
Must upload a jupyter notebook

Reminder of where to work

Assignment is self-contained

Where to go for support

# 2. Students complete the assignment



Sign in with your University of British Columbia account

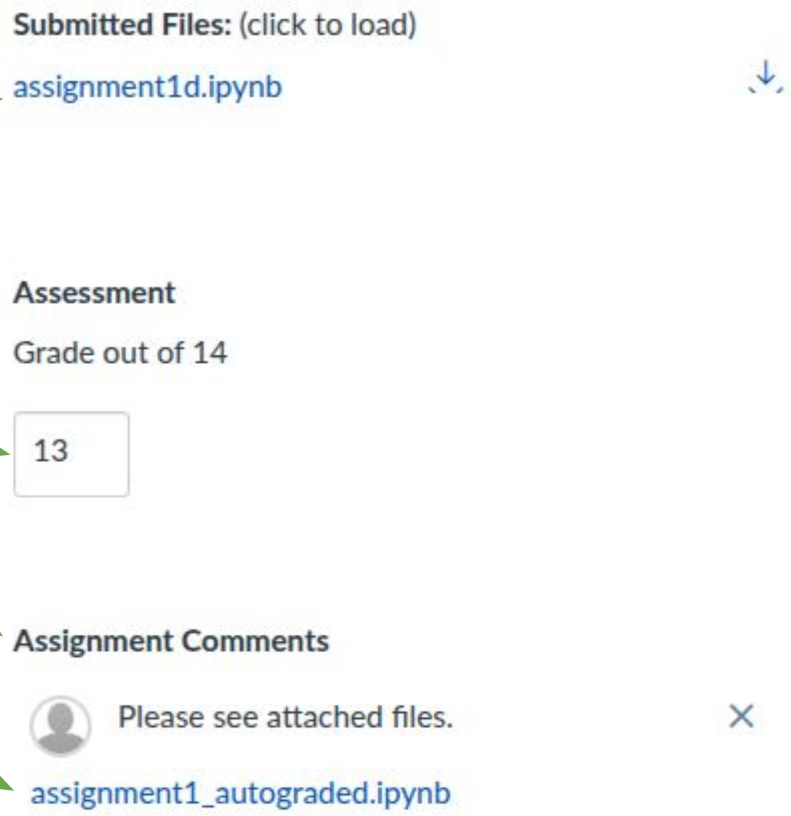
Terms and conditions / Privacy Policy

- Students download the blank assignment from LMS to their computer
- Working locally:
  - Open the blank assignment in the Jupyter notebook interface (or other editor)
  - Complete the assignment and save file
- Syzygy:
  - Upload the blank assignment to Syzygy
  - Complete the assignment and save file
  - Download the assignment to computer
- Students upload the complete assignment to LMS submission page

# 3. Grading the assignments

- Download student submission
- Run the notebook on your computer (with any additional tests pasted in)
- Input the grade in LMS/grader
- Feedback:
  - Enter written feedback in the comments
  - Use an LMS rubric to indicate which tests passed/failed, what needs to be improved
  - Save graded notebook and upload as feedback on submission
- Continue to next student

- [optional] post solutions or additional tests used or general comments for whole class



**Submitted Files:** (click to load)

assignment1d.ipynb

**Assessment**

Grade out of 14

13

**Assignment Comments**

Please see attached files.

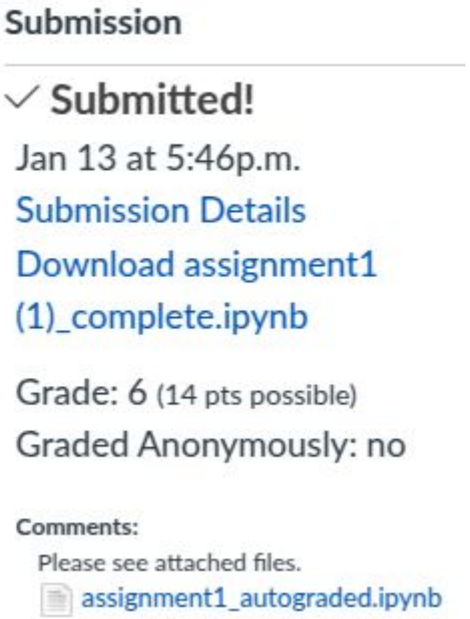
assignment1_autograded.ipynb

# 4. Students view grades and feedback

- Students see grade and download a file with their graded assignment

OR

- Students see grade and comments with no separate feedback file
  - Post grading scheme/tests used?



Submission

✓ Submitted!

Jan 13 at 5:46p.m.

Submission Details

Download assignment1 (1)_complete.ipynb

Grade: 6 (14 pts possible)

Graded Anonymously: no

Comments:

Please see attached files.

assignment1_autograded.ipynb

# Options for file exchanges

Post files in assignments in your LMS

- Can you upload individual graded notebooks as feedback on an assignment?
- Do you want to view assignments natively in your LMS?
- Can you grade assignments within the LMS?

Post files in a shared folder

# Manual Grading & Feedback

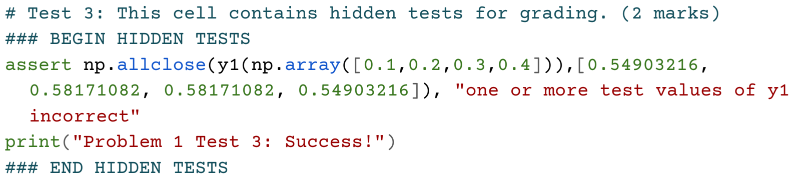
No packages means no automated grading

- Can replicate this with careful copy-paste

… also means more freeform and personalized feedback

# Feedback in Notebook

## Tests



```
# Test 3: This cell contains hidden tests for grading. (2 marks)
### BEGIN HIDDEN TESTS
assert np.allclose(y1(np.array([0.1,0.2,0.3,0.4])),[0.54903216,
  0.58171082, 0.58171082, 0.54903216]), "one or more test values of y1
  incorrect"
print("Problem 1 Test 3: Success!")
### END HIDDEN TESTS
```

- Tests check if an answer is correct
- Can include a custom message for success or failure of the test
- Insert tests, run notebook, add points for passed tests

## Markdown

- Write in any comments in a new markdown cell
- Free form customized feedback

## Uploading notebooks

- Leaving feedback in a notebook means the notebook has to get to the student
- Batch uploading generally requires API

# Feedback in LMS/grading tool

## Rubrics

| **Problem 1 _/5** | | | |
|---|---|---|---|
| Used lamda to write function<br>_/1 | Test 1: y1 should be callable<br>_/1 | Test 2: y1(0)=1<br>_/1 | Test 3: Check that y1 takes on correct values<br>_/2 |

- For each problem, list the tests and marks associated with it
- Check off passed/failed tests and any other rubric items as you go

## Written feedback

- Write in any comments in comments section of grading tool
- Free form customized feedback
- Paste in any hidden tests that failed

## Annotations

- If students also upload pdf or html version of notebook, your grading tool may allow drawing/writing directly on the file