stepik.org

Code Grading at Scale

Nikolay Vyahhi
Founder @ Stepik
vyahhi@stepik.org

May 23, 2017
Open edX Conference

Outline

1. What is Stepik
2. Basic code grading
3. How to use Stepik with Open edX
4. Understanding the basics
5. Multiple correct solutions
6. Adaptive feedback
7. Restrict available languages
8. Templates
9. Other graders on Stepik
10. What's inside

1/10
What is Stepik

Stepik is…

Grading tool for programming assignments (SaaS):

- Initially created to grade assignments for UCSD Bioinformatics MOOC on Coursera (on-air since 2013).
- Evolved into a full-featured **MOOC platform & LMS**.
- Grades **15-20K code submissions per day** (so far **7M total submissions**, **6K total assignments**).
- Available on https://stepik.org.

Written in **Python 3** (Django).

Stepik is…

Social enterprise supported by **JetBrains**
(you may know it for IntelliJ or PyCharm)

Stepik is **free** for open content (CC BY-SA), but **paid** if you want to make your content private (almost like the GitHub).

Though it’s free for all use cases in this presentation,
even if you make them private.

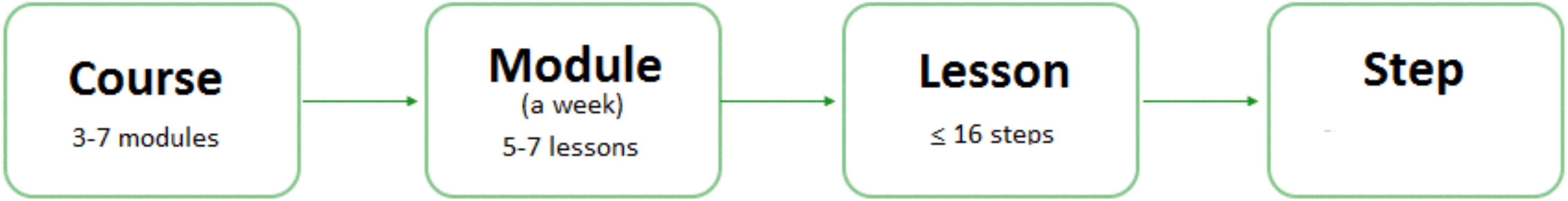
Content structure

Step = single assignment (problem)

Lesson = list of steps

Lesson can be embedded as a **component** in edX

We also have **modules** (list of lessons) and **courses** (list of modules), but it doesn’t matter today.



Course
3-7 modules

Module
(a week)
5-7 lessons

Lesson
≤ 16 steps

Step

2/10
Basic code grading

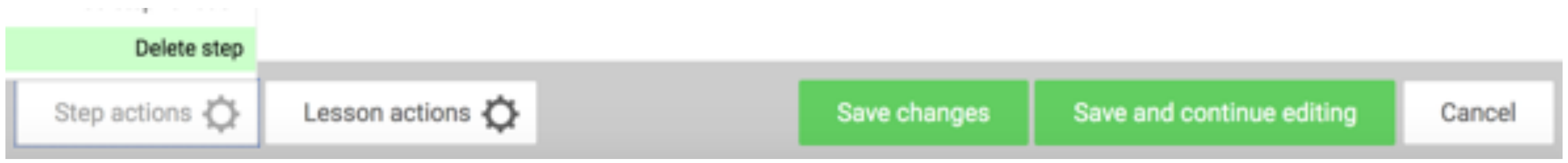
Demo Lesson

stepik.org/lesson/46775

A+B (sum of ints)

Let’s create A+B problem!

1. Sign up & create a new lesson: **stepik.org/new-lesson**
2. + → add **Code Challenge** step
3. Leave everything as default (default template is A+B)
4. Delete other steps (in "Step actions” menu), don’t need them now
5. Save the lesson



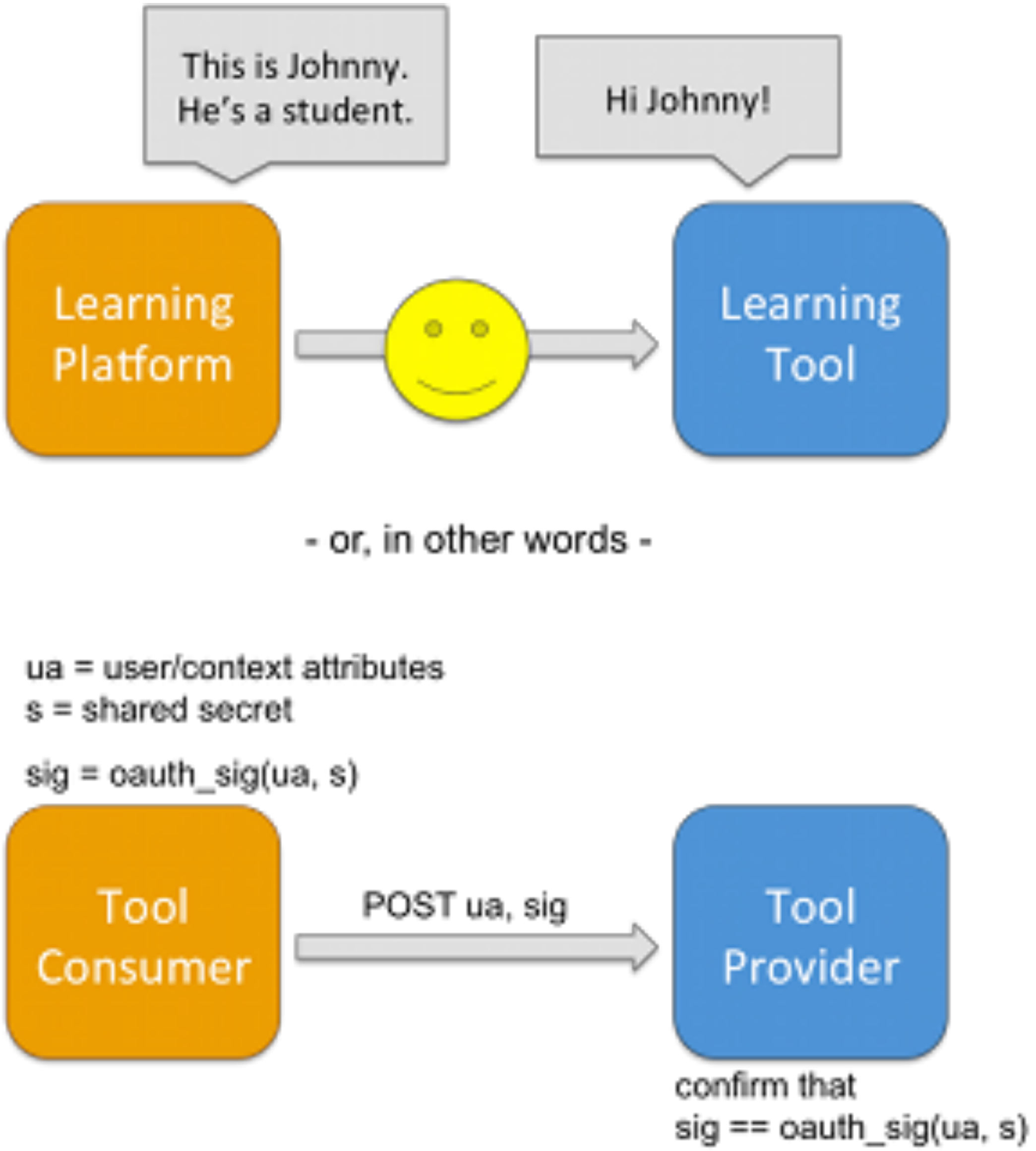
That’s it. Let’s solve it to verify as a student.

Submit in Python: `print(sum(map(int, input().split())))`

Docs: https://stepik.org/lesson/Step-Code-9173/

3/10
How to use Stepik with Open edX

Learning Tool Interoperability (IMS LTI)



LTI standard describes:

1. authentication
   (Platform → Tool)

2. passing grades
   (Tool → Platform)

LTI v1.1 is supported by [almost] any LMS & MOOC platform.

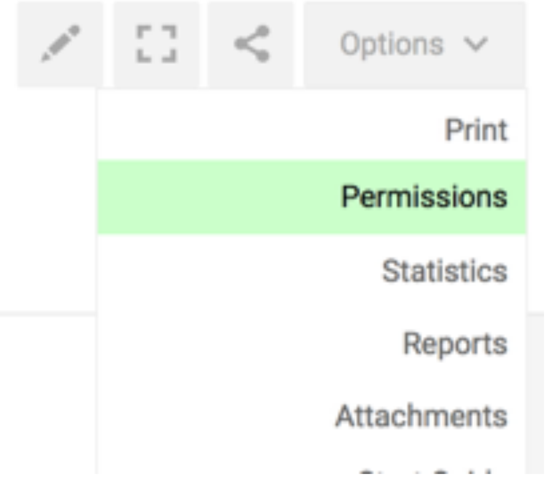
https://www.eduappcenter.com/docs/basics/index

LTI vs XBlock

XBlock should be deployed to your Open edX server.

LTI Tool can be deployed somewhere else (and by someone else).

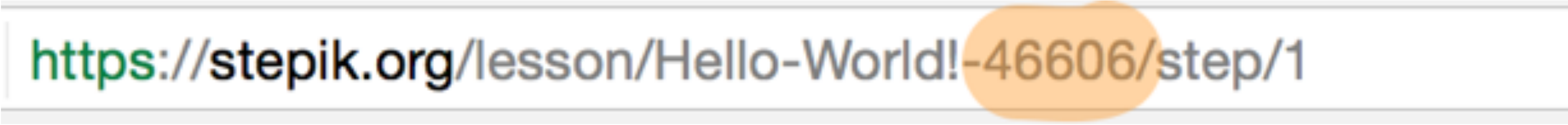
Stepik LTI



1. Lesson’s Options → Permissions →
   → LTI settings (bottom of the page)

2. Put **ANY** consumer & secret keys; then Save.

3. Remember your lesson's ID (number in the URL).



https://stepik.org/lesson/Hello-World!-46606/step/1

More info on http://stepik.help/en/support/solutions/articles/19000015504.

Open edX Installation

My playground for this presentation:

- edX: https://stepik-learning.tahoe.appsembler.com/
- Studio: https://studio.tahoe.appsembler.com/course/course-v1:stepik-learning+ST101+2017

You can use your local/web Open edX, or start a trial with Appsembler.

Open edX LTI

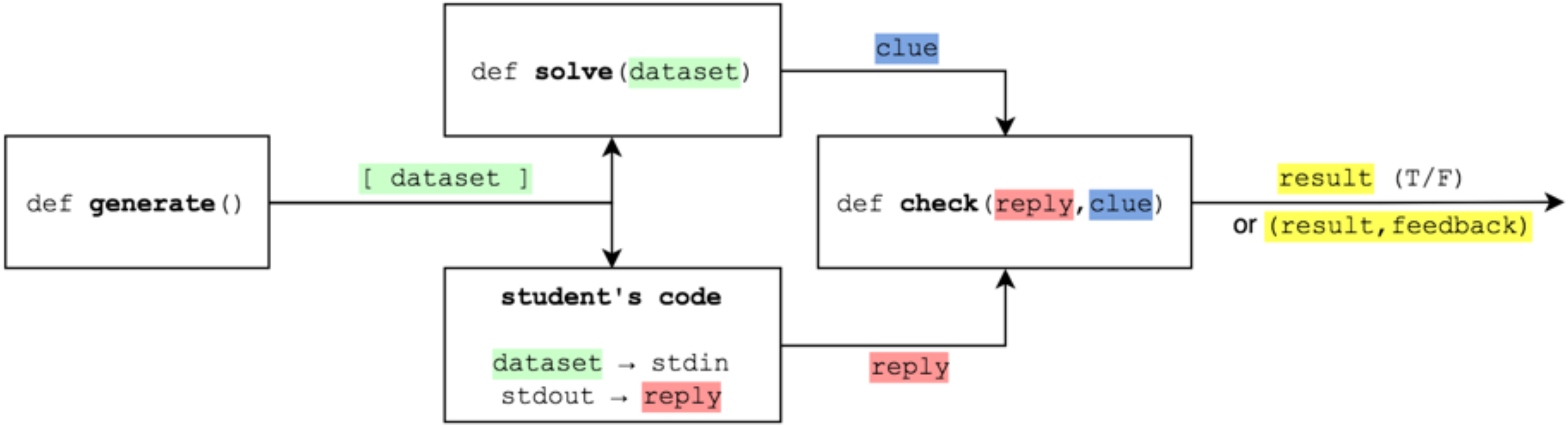
1. Advanced Settings:
   1.1. **Advanced Module List**: append **"lti_consumer"** (if not yet)
   1.2. **LTI Passports**: append **"somename:consumer:secret"** (same keys as in Stepik LTI settings)

2. Add **“LTI Consumer” component** to a unit:
   2.1. **LTI ID:** same as you used in step 1.2 above (**"somename"**)
   2.2. **LTI URL**: **https://stepik.org/lti/**
   2.3. **Custom Parameters**: **["lesson=ID"]** (lesson’s ID from Stepik URL)
   2.4. Scored: True (default=False)
   2.5. Accept grades past deadline: False (default=True)
   2.6. **Request user's username & email**: **Trues** (defaults=Falses)

More info on http://edx.readthedocs.io/projects/edx-partner-course-staff/en/latest/exercises_tools/lti_component.html

4/10
Understanding the basics

Understanding A+B



def **generate**(): [dataset] or [(dataset, clue)]

def **solve**(dataset): clue

dataset → **student's code** → reply

def **check**(reply, clue): result or (result, feedback)

More tests

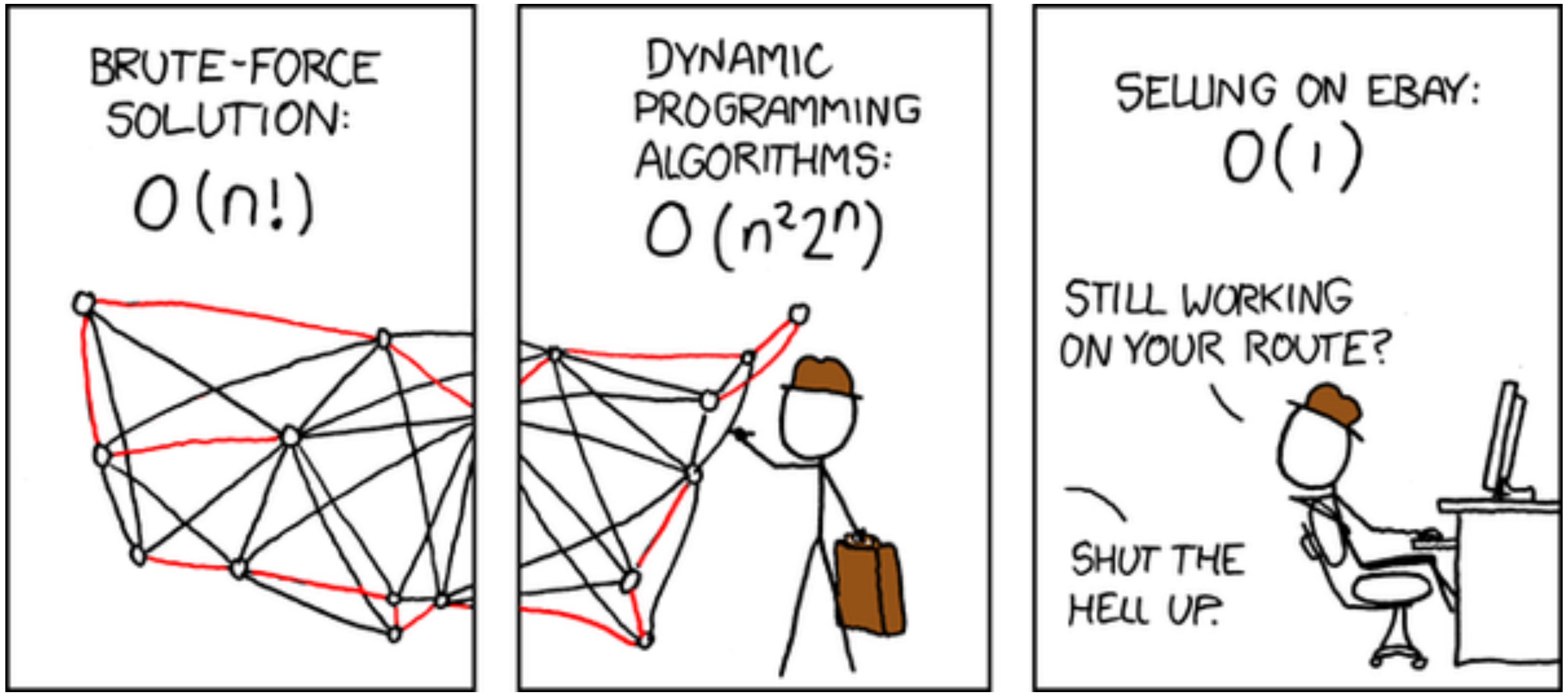
1. Change **generate**() to return more.

2. Upload new tests: zip with N & N.clue files.

3. Number of sample tests shown:
    - 1+ is good,
    - sometimes you want 0, later about it.

Limits

1. Time Limit (CPU time)
2. Memory Limit
3. Can be different for different languages



https://xkcd.com/399/

Errors

Students see **stdout** if submission is wrong, but only on sample tests.

Students always see **compilation errors & stderr (including runtime errors)**!

- student can print a test into stderr and will see it!
- but only one test, because such a solution will fail on it.

5/10
Multiple correct solutions

Multiple correct solutions

Let’s also accept A-B solutions. So the problem now is “find A+B or A-B”.

1. We need a different **clue**.

2. Clue may even be the **dataset**.

Multiple correct solutions

More problem examples:

1. Shortest path in a graph

2. Edit distance between strings

6/10
Adaptive feedback

Adaptive feedback

Let’s do not accept A*B wisely, i.e. give feedback to students on what’s wrong.

- return feedback from the **check** function iff reply == A*B

Adaptive feedback

Problem example: find all substrings in a text.

```
[m.start() for m in re.finditer('aba', ‘ababa')]

[m.start() for m in re.finditer('(?=aba)', 'ababa')]
```

7/10
Restrict available languages

Restrict available languages

1. `::python`

2. `::java`, `::kotlin`, etc.

Huh… but where is the default template?

8/10
Templates

Students Templates

Let’s add our own template after the `::language` block

Hidden Templates

1. `::header` – hidden from student (prepended)
2. `::code` – shown to student
3. `::footer` – hidden from student (appended)

Gives you ability to write graders in your preferred language!

Warning: not secure for interpreted languages:

- `print(open(__file__).read())`

And here you may want to show 0 sample tests.

Docs: https://stepik.org/lesson/9173/step/5

http://knowyourmeme.com/memes/wat

9/10
Other graders
on Stepik

Data Challenge

Like a code challenge, but client-side and with a single test. Students download this test (dataset), process it locally and upload result. Can be randomized for each student & attempt.

Good for machine learning and similar courses.

Docs: https://stepik.org/lesson/Step-Data-9172/

SQL Challenge

Students need to write SQL query.

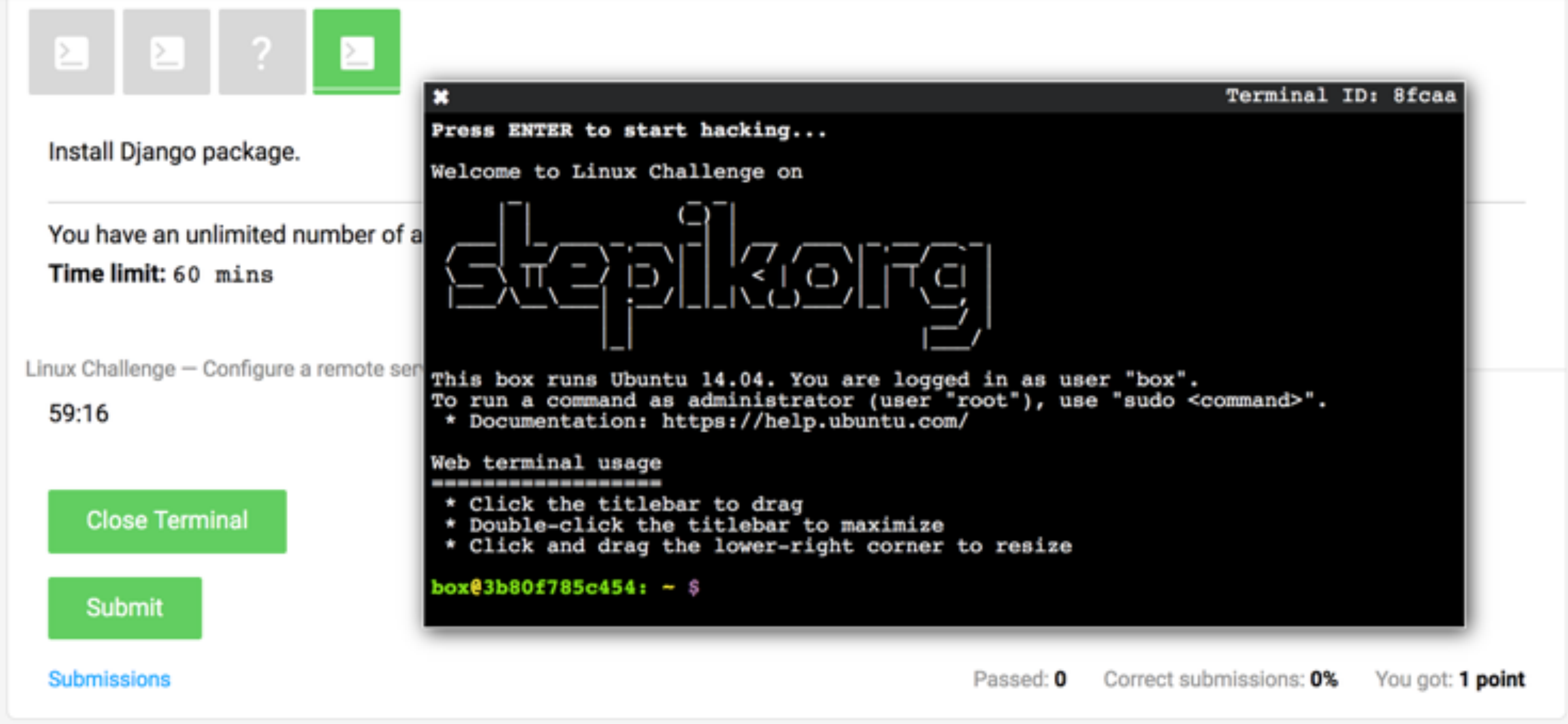
Instructors write init SQL script, Python checker.

MySQL inside.

Docs: https://stepik.org/lesson/Step-SQL-31182/

Linux Challenge

Students need to configure Linux machine (example: run Django).
Instructors write: bootstrap bash script, Python checker (machine state).
Dockers inside.



Install Django package.

You have an unlimited number of a

Time limit: `60 mins`

Linux Challenge — Configure a remote ser

59:16

```
Terminal ID: 8fcaa
Press ENTER to start hacking...

Welcome to Linux Challenge on
```

```
This box runs Ubuntu 14.04. You are logged in as user "box".
To run a command as administrator (user "root"), use "sudo <command>".
 * Documentation: https://help.ubuntu.com/

Web terminal usage
==================
 * Click the titlebar to drag
 * Double-click the titlebar to maximize
 * Click and drag the lower-right corner to resize

box@3b80f785c454: ~ $
```

Passed: **0** Correct submissions: **0%** You got: **1 point**

Docs: https://stepik.org/lesson/Step-Linux-9180/

10/10
What’s inside

Sandboxing

1. CodeJail (AppArmor):
    - forked for Python 3 and other languages https://github.com/StepicOrg/codejail/
    - hard to differ Runtime Errors vs. Memory Limits
    - All languages should be installed to the same machine (grading node). It’s hard to configure (especially multiple versions & libs)

2. Epicbox (Docker swarm):

- Better Memory Limits
- Mimics CodeJail API
- Easier to configure (Docker images)
- 2x times slower (latency) because of HTTP API
- https://github.com/StepicOrg/epicbox

Queue

Stepik platform puts student’s submission into the queue (RabbitMQ).

Grader (Celery worker) picks the next submission, grades it (using stateless RPC service), then puts the grade back to the DB.

Quite similar to edX External Graders http://edx.readthedocs.io/projects/edx-partner-course-staff/en/latest/exercises_tools/external_graders.html

We run multiple grading nodes (identical) at the same time.

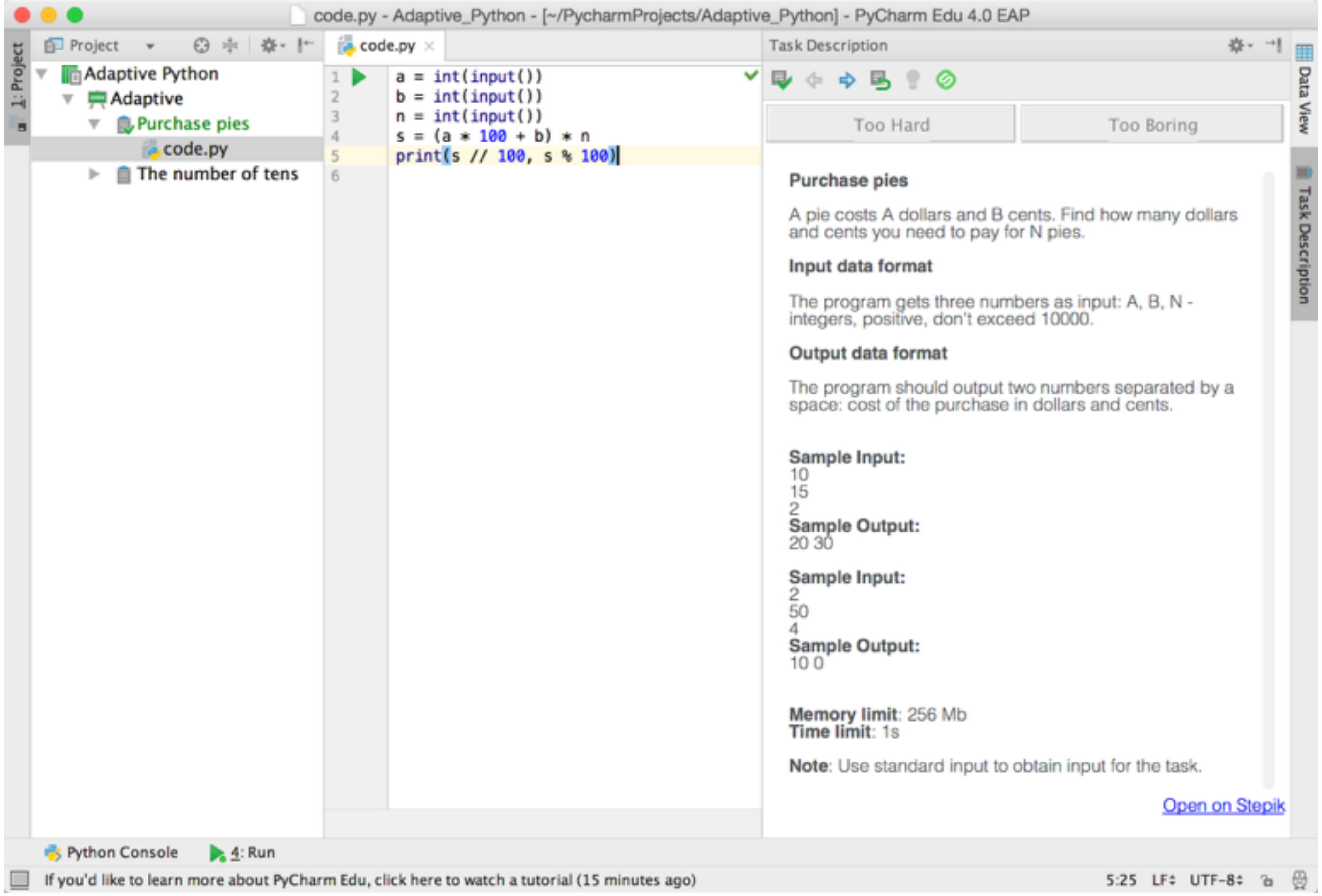
We can estimate grading time based on the current grading speed.

Stepik API

Just in case – Stepik has full REST API with OAuth:

https://github.com/StepicOrg/stepik-api

11/10
PyCharm Edu



```python
a = int(input())
b = int(input())
n = int(input())
s = (a * 100 + b) * n
print(s // 100, s % 100)
```

Purchase pies

A pie costs A dollars and B cents. Find how many dollars and cents you need to pay for N pies.

Input data format

The program gets three numbers as input: A, B, N - integers, positive, don't exceed 10000.

Output data format

The program should output two numbers separated by a space: cost of the purchase in dollars and cents.

Sample Input:
10
15
2
Sample Output:
20 30

Sample Input:
2
50
4
Sample Output:
10 0

Memory limit: 256 Mb
Time limit: 1s

Note: Use standard input to obtain input for the task.

Open on Stepik

If you'd like to learn more about PyCharm Edu, click here to watch a tutorial (15 minutes ago)

Stepik assignments can be solved using PyCharm Edu IDE. Current version is 3.5:

https://www.jetbrains.com/pycharm-edu/

https://blog.jetbrains.com/pycharm/2017/04/adaptive-python-course/

PyCharm Edu 4 EAP will be available on May 30, 2017

Links to others

https://github.com/coursera/programming-assignments-demo

https://github.com/blackav/ejudge (contest creation & grader)

https://github.com/laplab/carbon (sandboxing)

https://polygon.codeforces.com (contest creation platform)

http://www.inginious.org (code grading)

Coding problem sets platforms: https://www.hackerrank.com, https://www.topcoder.com, http://rosalind.info.

stepik.org

Thanks!

Nikolay Vyahhi
Founder @ Stepik
vyahhi@stepik.org

May 23, 2017
Open edX Conference