

Peer-review of assignment 4 for *INF3331-evgeniag*

Reviewer 1, morthand morthand@uio.no

Reviewer 2, sindrech, sindrech1@uio.no

Reviewer 3, peterdevoogd@live.nl, youremail@uio.no

October 10, 2017

1 Introduction

1.1 Goal

The review should provide feedback on the solution to the student. The main goal is to *give constructive feedback and advice* on how to improve the solution. You, the peer-review team, can decide how you organise the peer-review work between you.

1.2 Guidelines

For each (coding) exercise, one should review the following points:

- Is the code **working as expected**? For non-internal functions (in particular for scripts that are run from the command-line), does the program handle invalid inputs sensibly?
- Is the code **well documented**? Are there docstrings and are they useful?
- Is the code written in **Pythonic way**¹? Is the code easy to read? Are the variable/class/function names sensible? Do you find overuse of classes or not sufficient use of functions or classes? Are there parts of the program that are hard to understand?
- Can you find **unnecessarily complicated parts** of the program? If so, suggest an improved implementation.
- List the programming parts that are not answered.

Use (shortened) code snippets where appropriate to show how to improve the solution.

1.3 Points

The review is completed by pushing the review Latex source file (.tex files) and the PDF files to each of the reviewed repositories. The name of the files should be *feedback.tex* and *feedback.pdf* in the students assignment4 directory.

You will get up to 10 points for delivering the peer-reviews. Each of you should contribute to the review roughly equivalently - your team will get the same number of points².

2 Review - *to be filled out*

Specify the system (Python version, operating system, ...) that was used for the review.

Python 3.6.1, macOS Sierra

General feedback

Use this section to give general feedback about the solution such as advice for improved programming or documentation style.

Assignment 4.1

Add a review based on section 1.2. Do the tests have a meaningful name?

Both the test-file and the tests have the correct name according to the specification in the assignment, thereby meaningful. Both the constant and linear integral tests are passed. The code is not documented, but since we know what the assignment is about the code is self-explanatory.

¹<https://www.python.org/dev/peps/pep-0020/>

²In case a team-member does not contribute, please email simon@simula.no

Assignment 4.2

Add a review based on section 1.2.

The plot is excellent! It shows how precise the integration is when N gets very large. The integration functions are well defined, short and precise and does what they are supposed to do. It is easy to understand since the variables have meaningful names. There is not much need for improvements except maybe adding some comments.

Assignment 4.3

Add a review based on section 1.2. In addition, review the following assignment specific items:

- Is numpy being used effectively (e.i. vectorization where possible)?

The tests added to test-integrator works. The student has commented out a line where he tried to vectorize the function, since it makes the function slower. But due to my limited understanding of numpy and vectorization I am unsure why it makes it slower. But the solution presented with numpy linspace is the same as mine. And I would say that this works very well, and I guess a very good vectorization! The code could have been commented a bit to explain the $+2$ endpoints.

The report3 shows the integration and numpy integration speed comparison and explains the speedup using numpy.

Assignment 4.4

Add a review based on section 1.2.

@jit feels like magic. So when it does not work fast, like when the function is not hardcoded as the student explains in report4, it is difficult to understand why. The functions in the numba.py file is easy to understand since it's the same integration functions as in 4.2 except using @jit. The report4 explains the speedup from switching from normal python integration to using numba.

Assignment 4.5

Add a review based on section 1.2. If you are reviewing a INF3331 student, you can skip this review.

Assignment 4.6

Add a review based on section 1.2.

The 3 midpoint functions for integrator, numpy and numba seems to be correctly implemented. The results from running integrator-comparison was maybe a little hard to understand just from the print out to terminal, without looking at the code. It could maybe have printed out something like: "Finding the needed N to get a precision of 10^{-10} :" "Result for integrator: $N=x$ ", for this is just small stuff!

The student has found a N that gets a good enough precision. The assignment says nothing about choosing this number manually or creating a script that find this number automatic, so I see nothing wrong with just choosing a high enough number! In report5 the student explains why he has not tested numba midpoint. Even though it is the same as integrator, it could have been included, but I guess that it is not that important.

I would say that the student has delivered a very good solution to this assignment!

Assignment 4.7

Add a review based on section 1.2.

The setup files contains the 2 folders/packages, so I guess everything is correct! :)

Getting help from the TA to understand what and how module/packages work precisely was hard. I ran the setup file, and then I tried to run some files and got errors because it was unable to import stuff. I just moved some files and got it all working without any hassle. So I guess this is mostly due to my lack of understanding, and just choosing the easy way to get things working.

Assignment 4.8

Add a review based on section 1.2.

Don't think the student took part.

2.1 Useful Latex snippets

Here are some sample usage of Latex.

2.1.1 Sample code

```
1 import sys
2 print "This is a sample code"
3 sys.exit(0)
```

2.1.2 Mathematical equation

$$2\pi > 6 \tag{1}$$

References

[Gre93] George D. Greenwade. The Comprehensive Tex Archive Network (CTAN). *TUGBoat*, 14(3):342–351, 1993.