

Cloud Applications - Homework 1

Include your name and student ID to the header of all pages of your solutions. This will guarantee that we cannot mix up your solutions with solutions of other students.

Exercise 1: Review problems (15 points)

1.a (5 points) List two commercial examples for each cloud service model: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Besides the name of the offering, you should add a link to the page where the offering is described. You may want to review [Lec1](#) and read [NIST's cloud definitions](#) before you start your search.

1.b (5 points) Associate each item below with the cloud service model that best matches the item.

- a) Virtual Machines b) System administrators c) Webmail d) API e) Containers
f) Developers g) Storage h) Network i) End customers
j) Highest level of abstractions provided k) Lowest level of abstractions provided
l) Highest flexibility for development m) Lowest flexibility for development

IaaS: <enter here the letters of the items above that best match IaaS>

PaaS: <enter here the letters of the items above that best match PaaS>

SaaS: <enter here the letters of the items above that best match SaaS>

1.c (5 points) What is the major advantage of containers over virtual machines? What is the major advantage of virtual machines over containers?

Exercise 2: Google Cloud registration (5 points)

We are going to deploy the cloud applications developed as part of this course on Google Cloud. Google has generously provided us with coupons for each student in our class to pay for the costs. Click on the link below to request your coupon. You will be asked for your name and school email address (i.e. either @tufts.edu, or @cs.tufts.edu, or @eecs.tufts.edu). An email will be sent to you to confirm these details before a coupon is sent.

[Student Coupon Retrieval Link](#)

Do not share the link above with students outside of our class and do not try to obtain more than one coupon. Otherwise, we may not have coupons for all your classmates and Google may decline future grant requests due to abuse.

Once you redeem your coupon, answer this exercise with your first "Project ID". You can obtain your project ID at <https://console.cloud.google.com/>.

Exercise 3: Your first cloud application (20 points)

This exercise depends on exercise 2.

3.a (10 points) Go through chapter 2 of our textbook, and follow all the steps to make your machine ready to develop applications using Google App Engine, and create your first cloud application. You should use the machine you intend to use to develop all other applications for this course, so you will save time on following assignments.

Once you complete chapter 2, modify your application to include your name and student ID at the top of all pages of your application. Answer this exercise with the URL to your application.

3.b (10 points) List everything you have not understood well or feel that is confusing in chapter 2. Explain what exactly you did not understand or what makes an item confusing. Please include page, section, and paragraph for each item. In case you fully understood the content of chapter 2, you can formulate and answer a problem about the content. The instructor might use your problem in a future homework or test.

Exercise 4: Sentiment extractor in Python (30 points)

In case you do not have Python working on your computer, go through exercise 3 first.

4.a (10 points) Write a Python script called `docstats.py` that receives the name of a text file and an integer N as parameters from the shell, computes how many times each word occurs in the file, and prints out the top N words. One should be able to run your script as follows:

```
python docstats.py text.txt 10
```

4.b (10 points) Write a script called `docstats2.py` that receives the names of two text files and an integer N as parameters from the shell. The second text file will list a stop word (e.g. "an", "the", "how") per line. Docstat2 must compute how many times each word occurs in the first file ignoring the stop words listed in the second file, and print out the top N words. One should be able to run your script as follows:

```
python docstats2.py text.txt stopwords.txt 10
```

4.c (10 points) Write a Python script called `docstats3.py` that receives the names of two text files, computes the sentiment of the text in the first file using the second file as the sentiment lexicon, and prints out the sentiment of the first file. In other words, Docstats3 is a sentiment extractor. One should be able to run your script as follows:

```
python docstats3.py text.txt lexicon.txt
```

Sentiment analysis. A sentiment lexicon is a list of words and their valence. Valence is an integer number in the range [-5, 5]. To compute the sentiment of a text, initialize an accumulator variable with zero, and for each word in the text, add the valence of the word to the accumulator. If a word is not present in the lexicon, its valence is zero. The final value in the accumulator is the sentiment of the text.

Before submitting your scripts, test them with the files in file `hw1.zip` available on Piazza. The file `HW1.zip` includes three other files: `text.txt`, `stopwords.txt`, and `lexicon.txt`. Your scripts should obtain the following results:

```
python docstats.py text.txt 10
and 12
the 10
cloud 10
of 8
to 7
a 5
such 4
computing 4
as 4
applications 4
```

```
python docstats2.py text.txt stopwords.txt 10
cloud 10
computing 4
applications 4
application 4
stores 3
services 3
course 3
value 2
relational 2
overview 2
```

```
python docstats3.py text.txt lexicon.txt
The sentiment of file text.txt is 4
```

Notice that your scripts will be graded using other files.

Credits: Our file `lexicon.txt` is a slightly modified version of the lexicon [AFINN-96](#) that prof. Finn Årup Nielsen created.

Exercise 5: Your own problem (10 points)

Formulate and answer a problem about the content of Lectures 1, 2, and 3. The instructor might use your problem in a future homework or test.