**Final Exam Name \_\_Justin Minsk\_\_\_\_\_\_\_\_**

**DATA 520**

Dec 13, 2017

This is an open book, open notes exam. Simply edit this document.

**Work on this exam on your own, alone. Do not discuss with anyone else.**

**No internet use is allowed except to email me this document when you have completed it.**

**The completed exam is due BEFORE 6:30 PM TODAY. Late paper will be penalized 5 points per minute late.**

**First, take a deep breath. You have plenty of time to complete this!**

**Part A: data types (30 points)**

1. Explain the differences between Lists, Sets, Tuples, and Dictionaries, in how each is designated, how each is different, and how each is especially useful for certain things.

Lists store data sequentially and is created using [] brackets. Lists are good for data you need to sort and mutate. Sets have no certain order and do not contain duplicates and are created with {} braces. Sets are good for checklists and/or catalogs and are faster than lists. Tuples are like lists but cannot be modified after assignment and are created with () parenthesis. These are good when you need a static list. They also are useful for changing variables. Dictionaries store items with a string index rather than a numerical index. Dictionaries are created using {} and : to separate key’s from values. Dictionaries are good when you have a key to value pair relationship.

**Part B**: **Short answer (40 points). Use a maximum of about five (5) sentences. Be concise.**

1. What is an algorithm?

An algorithm set of steps that accomplishes a task. In programing this will often be done by a top-down design with the algorithm being written in English then written in Python. Since English does not always translate to Python well debugging and testing is need to make sure that the algorithm accomplishes the task.

2. Are some algorithms better than others? What different ways should we judge which algorithm to use?

Yes, there are subjectively better algorithms to solve certain problems. The best algorithm has the best mix of performance and simplicity so that the algorithm is easy enough to use, but is fastest and most efficient or how much time and memory it uses.

3. Explain in numerical terms how fast a binary search is, compared to the length of the search list.

The numerical value for binary is log2 N steps or logarithmic growth and linear would be N steps or linear growth.

4. What are the advantages in debugging using classes that inherit from unittest.TestCase?

It allows you to make sure your program or algorithm can pass all test cases. This is accomplished by making sure your unittest.TestCase covers all test coverage or size, dichotomies, boundaries, and order. It also separates your test code from your actual code so that you can test independent of where main code. It also documents each test case so that you know what bugs you are running into.

5. What are the advantages and disadvantages to using SQLite in Python?

Large scale database need to be stored in a relational database like an SQL database. Databases take the data and turn them into tables with rows or records and columns which are stored with a name and type they also can contain NULL values or values with no ‘real’ values. SQL can be used in command-line tools and in GUI’s and changes in the database do not happen until they are committed allowing multiple programs to use the same database and keep it in a consistent state. SQL is its own unique langue that requires you know two programing langue also if the committing is done incorrectly you can interrupt a second program working on the database.

**Part C: Coding and code (30 points). You can leave out the function design recipe elements.**

1. Provide code to open e:\text editor\newtext.txt in three different modes. (one line each)

a. with open(' e:\text editor\newtext.txt', 'r') as example\_file:

b. with open(' e:\text editor\newtext.txt', 'w') as example\_file:

c. with open(' e:\text editor\newtext.txt', 'a') as example\_file:

2. Write a function to return only the last line in a text file (filename is the argument) in the default folder.

def last\_line(filename):

with open(filename, 'r') as example\_file:

lines = example\_file.readlines()

last\_line = lines[-1]

return last\_line

3. A file in the default folder, planets.txt, has these five lines for text:

Mercury

Venus

Jupiter

Earth

Neptune

Write code to open the file and return (AND print) the lines in alphabetical order.

def alph\_order(filename):

with open(filename, 'r') as example\_file:

lines = example\_file.readlines()

alph\_lines = sorted(lines)

return alph\_lines

if \_\_name\_\_ == '\_\_main\_\_':

for planet in alph\_order('planets.txt'):

print(planet.strip())