

NAME:

**Open book. No calculators, no computer.
Write all answers on these sheets. Do not ask questions!**

question	1	2	3	4	5	6	7	total
points								
maximum	20	10	10	10	20	15	15	100

1. A manager wants an automatic birthday announcer for the employees. Each employee receives a congratulatory email on the birthday date. The manager is sent an email a week before the birthday date as a reminder to plan celebrations. The announcer program is scheduled to run automatically every day at 1 AM. Describe the modular design for this program. What is at the bottom of it? Draw the dependencies between the modules. For each module give a one line description of its functionality. Justify your design.

2. Explain the differences between static whitebox and dynamic blackbox testing.
Give examples of activities in both static whitebox and dynamic blackbox testing.

/10

3. Compare the use of open source versus proprietary software:

(a) Why might open source software be better than proprietary software?
Give two reasons, explaining each time what “better” means.

(b) Why might proprietary software be better than open source software?
Give two reasons, explaining each time what “better” means.

/10

4. What is the use of the `assert` statement in Python? Describe an example of a good use of an `assert`. Justify why it is good to use an `assert` in your example.

/10

5. The file `currency.txt` contains the following data:

	dollar	yen	euro
dollar	1.0	110.54	0.6838
yen	0.009046	1.0	0.006186
euro	1.4625	161.6647	1.0

From this table, we see that one euro is worth 1.4625 dollar and 161.6647 yen. In general, do not assume that the currencies are listed in the order as shown. Furthermore, there are more currencies on file than shown in the example.

Think of a program that will prompt the user for the name of a file (e.g.: `currency.txt`) and two names of currencies, say A and B. The program will then print that one A is worth x B, where the number x is retrieved from the file.

- (a) Describe in plain English (use complete sentences) the structure of the program. What data structures do you use? Which built-in functions do you apply?

- (b) Give Python code. Do not worry about wrong user input or incorrect data.

6. Consider an account management program for a utility company. Customers can login, check the status of their account, make a payment, and logout. After logging in, employees use the program to email bills to customers. Use UML to draw a use case diagram for an object oriented design of the program. For each object in the diagram, write one line to define what the object represents.

/15

7. Consider the search in a phone directory. We search the directory in both ways. Either we provide a name and then the program shows the corresponding phone number, or we give a phone number and then the program shows the corresponding name. Draw the layout of the GUI for this search program. Indicate on your drawing what widgets you will use.

/15
