

Student Number

--

The University of Melbourne
Department of Computing and Information Systems

SAMPLE Final Examination, Semester 1, 2015 COMP10001 Foundations of Computing

Reading Time: 15 minutes. **Writing Time:** 2 hours.

This paper has 15 pages including this cover page.

Instructions to Invigilators:

Students must write all of their answers on this examination paper. Students may not remove any part of the examination paper from the examination room.

Instructions to Students:

There are 8 questions in the paper worth a total of 60 marks, making up 50% of the total assessment for the subject.

- All questions should be answered by writing a brief response or explanation in the lined spaces provided on the examination paper.
- It is not a requirement that all the lined spaces be completely filled; answers should be kept concise.
- Only material written in the lined spaces provided will be marked.
- The reverse side of any page may be used for notes or draft answers.
- Your writing should be clear; illegible answers will not be marked.
- Extra space is provided at the end of the paper for overflow answers. Please indicate in the question you are answering if you use the extra space.
- Your answers should be based on Python 3 (the version that Grok uses), and can use any of the standard Python libraries.

Authorised Materials: No materials are authorised.

Calculators: Calculators are not permitted.

Library: This paper may NOT be held by the Baillieu Library.

<i>Examiners' use only</i>									
1	2	3	4	5	6	7	8	9	Total

Part 1: Algorithmic Thinking

Question 1

[8 marks]

Evaluate each of the following expressions; what is the “return value” in each case?

(a) `"2,4".split(",")`

(b) `bool(2 or "")`

(c) `[i%2 for i in range(3)]`

(d) `sorted(({(1,1): "racehorse", (2,2): "one too"}.items())[0][1][1:4])`

Question 2**[6 marks]**

Consider the following code fragment:

```
x = [("andrew", True), ("kim", False),  
      ("sandy", True), ("tim", True)]  
limit = 2  
i = 0  
j = len(x) - 1  
y = []  
while i < limit:  
    if x[j][1]:  
        y.append(x[j][0])  
        i += 1  
    j -= 1
```

List the values of the variables `i` and `j` after each iteration of the loop.

What is the final value of `y`?

Part 2: Constructing Programs

Question 4

[9 marks]

An *acrostic* is a piece of text in which the first letter, or the last letter, of every line, can be joined together to construct a word or a message.

The function `get_message()` accepts a string argument that contains the name of a text file and a number, which is either 0 or 1. The function is intended to produce the message in uppercase letters from the lines in the text file. If `which` is equal to 0, then the message should be constructed from the first letter of each line in the text. If `which` is equal to 1, then the message should be constructed from the last letter of each line in the text.

Calling `get_message("poe.txt", 0)` should return "ELIZABETH", given the following text file `poe.txt`:

```
Elizabeth it is in vain you say
Love not - thou sayest it in so sweet a way
In vain those words from thee or LEL
Zantippe's talents had enforced so well
Ah! if that language from thy heart arise
Breath it less gently forth - and veil thine eyes
Endymion, recollect, when Luna tried
To cure his love - was cured of all beside
His follie - pride - and passion - for he died
```

Calling `get_message("compl.txt", 1)` should return "CSNDNMW", given the following text file `compl.txt`:

```
qc
jess
madison
david
shaanan
tim
andrew
```

The initial definition of the function `get_message()` is given below.

```
1 def get_message(filename, which):
2     text = open("filename").read()
3     lines = text.split(',')
4     message = ''
5     for line in lines:
6         if which == 1:
7             index = len(line)
8         else:
9             index = 0
10        message += line(index)
11    return message
```

However, there are several errors in the given function definition. Identify exactly three (3) errors and specify: (a) the line number where the error occurs; (b) the type of error, as *syntax*, *logic* or

runtime; and (c) how you would fix each error, in the form of corrected code.

Note: Assume that the first and the last letters in every line are always Latin characters (in the range 'a', 'b', ..., 'z' or 'A', 'B', ..., 'Z') and that there are no blank lines in the text.

DRAFT

Question 5**[8 marks]**

Consider the following sales profit data in `sales.csv`:

```
Product,Year,Salesman,Profit
TimTom,2009,Andrew,2000
Catbury,2010,Andrew,3000
Pufflova,2009,Andrew,500
TimTom,2009,Tim,1000
Catbury,2010,Tim,5000
KidCat,2010,Tim,700
```

Your task is to write a program that finds the *salesman that earns the maximum total profit over the year 2009 and 2010 and the corresponding maximum total profit*. Using the data above, the program should print the following output at the end of code execution:

```
Tim earns the highest profit of 6700.
```

Complete the code by providing a short code snippet to fill each of the boxes below, based on the above description of what the program does.

```
import 

data = open("sales.csv")
sales = list(csv.DictReader(data))

totalsales = 

for row in sales:
    salesman = row['Salesman']
    profit = 
    if salesman  totalsales:
        totalsales[salesman] = 0
     += profit

max_salesman = ''
max_profit = 
for salesman in totalsales:
    if max_profit  totalsales[salesman]:
        max_profit = totalsales[salesman]
        max_salesman = 

print("{0} earns the highest profit of {1}".format(max_salesman, max_profit))
```

1 _____

2 _____

3 _____

4 _____

5 _____

6 _____

7 _____

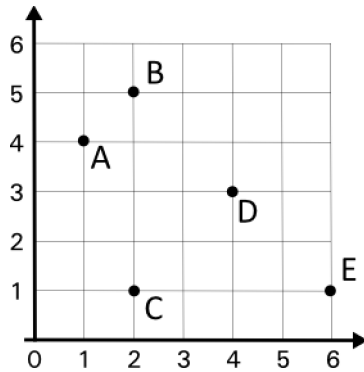
8 _____

DRAFT

Question 6

[9 marks]

Consider some hypothetical cities on a map:



The following data is available in the file `cities.csv`, and specifies the x and y coordinates of the cities on a map.

```
city,x,y
A,1.0,4.0
B,2.0,5.0
C,2.0,1.0
D,4.0,3.0
E,6.0,1.0
```

The *taxicab distance* between two cities is defined as the sum of the absolute differences of their coordinates (that is the distance travelled by a taxi ride that goes along the grid of the map, and never diagonally). The function `taxicab_dist()` to calculate *taxicab distance* has been defined for you. The first two arguments of the function are the x and y coordinates of the starting city and the next two are the coordinates of the destination city (see the code snippets below).

Complete a Python program that produces an HTML table containing the taxicab distances between any pair of cities. The table should be similar to the following figure.

	A	B	C	D	E
A	0.0	2.0	4.0	4.0	8.0
B	2.0	0.0	4.0	4.0	8.0
C	4.0	4.0	0.0	4.0	4.0
D	4.0	4.0	4.0	0.0	4.0
E	8.0	8.0	4.0	4.0	0.0

The program has been started for you; you need to write the code that generates the distance table.

```
import csv

def taxicab_dist(x1,y1,x2,y2):
    return abs(x2-x1) + abs(y2-y1)

data = csv.reader(open("cities.csv"))
header = next(data)
rows = list(data)
num_cities = len(rows)

print("<!DOCTYPE html><html><body>")
# write your code from here
```

continued on the following page

```
print("</body></html>")  
# end of program
```

DRAFT

Part 3: Conceptual Questions

Question 7: Algorithmic Problem Solving

[7 marks]

(a) What is an “algorithm” in the context of Computing?

[3 marks]

(b) Outline the “generate-and-test” strategy of algorithmic problem solving. With the aid of an example, explain what sort of problems it is commonly applied to.

[4 marks]

Question 8: Representation and Multimedia

[5 marks]

Consider a blank USB key with a capacity of 4 gigabytes (assume that 1 gigabyte = 1,000,000,000 bytes). A wedding photographer intends to use this 4GB key to distribute all of their works. The photographer takes each of the digital images as a RGB colour image *4000 pixels wide* and *3000 pixels high*. To save space, these images are compressed using JPEG compression. This will reduce the space required to store the images to *one-tenth* ($1/10$) of the original space.

Calculate the maximum number of images that can be stored on the USB key, assuming JPEG compression is used.

[5 marks]

This is blank space for further answers should you need it. Please ensure that you label the answers in this area carefully, and that you indicate on the corresponding question page that your answer can be found here.

DRAFT

⚡⚡⚡⚡⚡⚡⚡⚡ END OF EXAM ⚡⚡⚡⚡⚡⚡⚡⚡⚡