Student Number _____

The University of Melbourne
Department of Computing and Information Systems

Final Examination, Semester 2, 2015
COMP10001 Foundations of Computing

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

This paper has 19 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 10 questions in the exam worth a total of 120 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. Excessively long answers or irrelevant information may be penalised.
- 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.

Examiners' use only										
1	2	3	4	5	6	7	8	9	10	Total

Part 1: Algorithmic Thinking

Question 1

[12 marks]

Evaluate the following expressions, and provide the output in each case.

(a) `len([1,2,3,(4,5),[]])`

(b) `sorted({'cat': 4, 'human': 2}.keys())`

(c) `True and not(False or True)`

(d) `[21,10,2015][-1:3] + [25,12,2016][2:3]`

(e) `[i**2 for i in range(0,10,3) if not i%2]`

(f) `','.join('='.join(t) for t in [('a','1'), ('b','2')])`

Question 2

[10 marks]

Rewrite the following function, replacing the `while` loop with a `for` loop, but preserving the remainder of the original code structure:

```
def fact(n):
    result = 1
    ni = 1
    while not ni > n:
        result *= ni
        ni += 1
    return result
```

Question 3

[10 marks]

The following code is intended to be a function to create and return a string that contains the valid HTML code for a table. The entries in the table come from the values in the given list of lists, where each of those lists corresponds to a row in the table.

As presented, the lines of the function are out of order. Put the line numbers in the correct order and introduce appropriate indentation (indent the line numbers to show how the corresponding lines would be indented in your code).

```
1  out += "<tr>"
2  out += "</td>"
3  return out
4  for element in array[i]:
5  def array2table(array):
6  for i in range(len(array)):
7  out += "</tr>"
8  out += "</table>"
9  out += "{value}".format(value=element)
10 out = "<table>"
11 out += "<td>"
```

Question 4

[12 marks]

The following function `findOutliers(values, nSigma)` takes a list argument `values` (containing a sequence of one or more real-valued numbers) and a positive real-valued argument `nSigma` (containing the sensitivity of the test) and returns the list of indexes of any outlier values in the given list of numbers. This is done by calculating the upper and lower control limits for values, between which most values are expected to occur.

The method we use to calculate the upper and lower control limits for a given list of numbers is called a Shewart control chart for individuals. We first calculate the mean μ of the given set of N values. For example, for a list of values `[4.0, 4.2, 4.1, 4.4, 4.3]`, the mean of the values is:

$$\begin{aligned}\mu &= \frac{1}{N}\left(\sum_{i=1}^{N} x_i\right) \\ &= \frac{4.0 + 4.2 + 4.1 + 4.4 + 4.3}{5}\end{aligned}$$

We then calculate the mean of the absolute differences between successive values, which we denote r. For the list of values above, the mean of the absolute differences of the values is:

$$\begin{aligned}r &= \frac{1}{N-1}\left(\sum_{i=2}^{N} |x_i - x_{i-1}|\right) \\ &= \frac{|4.2 - 4.0| + |4.1 - 4.2| + |4.4 - 4.1| + |4.3 - 4.4|}{4}\end{aligned}$$

where $|x|$ denotes the absolute value of x.

We can then calculate the upper control limit UCL and lower control limit LCL as follows:

$$\text{UCL} = \mu + \frac{n \times r}{1.128} \qquad \text{LCL} = \mu - \frac{n \times r}{1.128}$$

where n is the sensitivity of the test (typically $n = 3$) and 1.128 is a statistical parameter. Any value above the upper control limit or below the lower control limit is considered an anomaly.

```
1  def findOutliers(values, nSigma):
2      NORM = 1.128
3      if len(values) < 2: return []
4      sumValues = values[0]
5      sumDifferences = 0
6      for i in range(len(values)):
7          sumValues += values[i]
8          sumDifferences += (values[i] - values[i-1])
9      upperCL = sumValues/len(values) + (nSigma/NORM)*sumDifferences/(len(values)-1)
10     lowerCL = sumValues/len(values) - (nSigma/NORM)*sumDifferences/(len(values)-1)
11     outliers = []
12     for i in range(len(values)):
13         if values[i] > upperCL or < lowerCL:
14             outliers.append(i)
15     return outliers
```

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: If no outlier values are detected or there is an error in the input, the function should return the empty list []. You can assume that the list argument `values` contains one or more values, and that `nSigma` is positive. If `values` contains only one value, then there are no outliers.

Part 2: Constructing Programs

Question 5

[12 marks]

The function `assignRooms(nRooms)` takes a given number of rooms as a positive integer, and assigns lectures to rooms. A *lecture* is represented by a tuple containing the *subject name* and the *lecture name*, e.g., the lecture `('COMP10001','L1')` is the first lecture `'L1'` for the subject `'COMP10001'`. The function contains two predefiend lists:

```
subjectNames = ['COMP10001','COMP10002']
lectureNames = ['L1','L2','L3']
```

The product of these subject names and lecture names define the six lectures that need to be assigned to rooms by the function `assignRooms`:

```
('COMP10001', 'L1'), ('COMP10001', 'L2'), ('COMP10001', 'L3'),
('COMP10002', 'L1'), ('COMP10002', 'L2'), ('COMP10002', 'L3')
```

Each room is represented as a list, where the list will contain the lectures that have been assigned to that room. As much as possible, we try to share lectures for a subject between different rooms.

For example, the function call `assignRooms(2)` will return:

```
[[('COMP10001', 'L1'), ('COMP10001', 'L3'), ('COMP10002', 'L2')],
 [('COMP10001', 'L2'), ('COMP10002', 'L1'), ('COMP10002', 'L3')]]
```

In another example, the function call `assignRooms(3)` will return:

```
[[('COMP10001', 'L1'), ('COMP10002', 'L1')],
 [('COMP10001', 'L2'), ('COMP10002', 'L2')],
 [('COMP10001', 'L3'), ('COMP10002', 'L3')]]
```

Provide code to insert into each of the numbered boxes in the code on the next page to complete the function as described. Note that your code will be evaluated at the indentation level indicated for each box.

```
from itertools [      1      ]

def assignRooms(nRooms):
    subjectNames = ['COMP10001','COMP10002']
    lectureNames = ['L1','L2','L3']

    lectureIter = product(subjectNames, lectureNames)
    lectures = [      2      ]

    rooms = [      3      ]
    roomIter = cycle(rooms)
    for lec in lectures:
        room = [      4      ]
        room.[      5      ]
    return [      6      ]
```

(1) _____

(2) _____

(3) _____

(4) _____

(5) _____

(6) _____

Question 6

[14 marks]

Write a function `findPrefixWords(inFileName, outFileName, testWord)` that takes three arguments: `inFileName` (a non-empty string containing the name of a file to be read) , `outFileName` (a non-empty string containing the name of a file to be created and written), and `testWord` (a non-empty alphabetic string).

The input file whose name is given by `inFileName` contains one or more words, which are separated by spaces or newlines. The function reads each word in the input file, and if a word in the input file begins with the given `testWord`, then that word is written to the output file. Each word written to the output file should be on a new line.

For example, `findPrefixWords('in.txt', 'out.txt', 'settle')` would read words from the file `'in.txt'`, and any word in `'in.txt'` that begins with the string `'settle'` would be written to `'out.txt'`. Then if the words `'settler'`, `'settles'`, `'settle'` and `'settled'` appear in the input file, they would each be written to the output file. Whereas if the words `'set'` or `'settling'` appear in the input file, they would not be written to the output file.

The function `findPrefixWords` should return the number of words that were found that started with the given `testWord`. Note that this return value is also the number of words that have been written to the output file.

Question 7

[16 marks]

You are given a square table of integers. Each row of the table is stored as a list, and the entire table is stored as a list of lists. For example, the following table:

0	1	2	3
4	5	6	7
8	7	6	5
4	3	2	1

would be stored as the following list of lists in Python:

```
table = [[0,1,2,3], \
         [4,5,6,7], \
         [8,7,6,5], \
         [4,3,2,1]]
```

We define a square subtable of a given table by (1) the row and column of the top left cell in the table (we refer to this as the offset), and (2) the size of the subtable in terms of the number of rows or columns. For example, given the table above, the subtable at row 0 and column 1 of size 2 is:

1	2
5	6

Write a function `rotateSubTable` that rotates a specified subtable in a given table by 90 degrees clockwise. Your function `rotateSubTable(inTable, offset, size)` should take a square table as the argument `inTable`, as well as the arguments `offset` and `size` of the subtable of interest in `inTable`, where `offset` is a tuple of two integers corresponding to the (row,column) of the top left cell in the subtable, and `size` is the number of rows or columns in the subtable as an integer. Your function should return a copy of the original table with the subtable rotated clockwise by 90 degrees, and the other elements of the table unchanged.

For example, using the table above, the function call `rotateSubTable(table,(0,1), 2)` would result in the following table:

0	5	1	3
4	6	2	7
8	7	6	5
4	3	2	1

or in Python:

```
>>> rotateSubTable(table,(0,1), 2)
 [[0,5,1,3], [4,6,2,7], [8,7,6,5], [4,3,2,1]]
>>> rotateSubTable(table,(1,1), 3)
 [[0,1,2,3], [4,3,7,5], [8,2,6,6], [4,1,5,7]]
>>> rotateSubTable(table,(0,0), 4)
 [[4,8,4,0], [3,7,5,1], [2,6,6,2], [1,5,7,3]]
```

You can assume that the table is square (i.e., length of each sublist in `inTable` is equal to the number of sublists in `inTable`), that the `offset` lies within the table, and that `size` is between 1 and the number of rows or columns in `inTable`. You can also assume that the subtable to be rotated lies completely within the given table.

Part 3: Conceptual Questions

Question 8: Algorithmic Problem Solving [12 marks]

(a) Briefly explain the concept of "memoisation" and why it is important.

[6 marks]

(b) Briefly outline the "brute force" strategy of algorithmic problem solving, with the aid of an example.

[6 marks]

Question 9: Applications of Computing [12 marks]

(a) Briefly explain the roles of the *public key* and the *private key* in public-key cryptography.

[6 marks]

(b) List three (3) examples of possible applications of wireless sensor networks.

[6 marks]

Question 10: HTML and the Internet [10 marks]

(a) Briefly explain the differences between *binary* and *grayscale* methods for representing raster images.

[5 marks]

(b) What is the relationship between a "port" and a "hostname" on the Internet?

[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.

— End of Exam —