Student Number [   ]

The University of Melbourne
School of Computing and Information Systems

Final Examination, Semester 1, 2017
COMP10001 Foundations of Computing

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

This paper has 35 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 11 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.
- 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 be held by the Baillieu Library.

Examiners' use only											
1	2	3	4	5	6	7	8	9	10	11	Total

Part 1: Code Interpretation

Question 1   [10 marks]

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

(a) `3 / 1`

(b) `'executrix'[3:6] + 'zippiest'[-3]`

(c) `'completely different' and not('to be seen')`

(d) `sorted({'human': ['arthur', 'trillian'], 'robot': ['marvin']}.values())[1][-1]`

(e) `[str(i) + j for i in range(8, 10) for j in 'HD']`

Question 2

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

```python
def same_colour(suits):
    SUIT_COLOURS = {'S': 1, 'H': 2, 'D': 2, 'C': 1}
    colour = ''
    for suit in suits:
        if not colour:
            colour = SUIT_COLOURS[suit]
        elif SUIT_COLOURS[suit] != colour:
            return False
    return True
```

Question 3

The function `animalise` is intended to take a string, identify each occurrence of a word which is also an animal name — as determined based on an input file with a single animal name per line — and replace each such word with `'animal'` (by default). For example, assuming the file `"animal-names.txt"` has the following content:

```
bear
bug
mouse
python
tick
worm
```

the function would produce the following output:

```
>>> orig = "bear with me as I fix the bug in my python code"
>>> animalise(orig, "animal-names.txt")
'animal with me as I fix the animal in my animal code'
```

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   for line in f:
2   out += word + " "
3   adict = {}
4   out += newword + " "
5   out = ''
6   else:
7   if word in adict:
8   with open(afile) as f:
9   def animalise(text, afile, newword="animal"):
10  adict[line.strip()] = True
11  return out[:-1]
12  for word in text.split():
```

(provide your answer on the next page)

Question 4

The following function is meant to take a list `lst` and return the maximum value, based on recursion. The following are example outputs the code should produce:

```
>>> print(rec_max([]))
None
>>> print(rec_max([0, 4, -1, 2]))
4
>>> print(rec_max([0, 1, -1, 2, 0, 2]))
2
```

Identify exactly three (3) errors in the code (using the provided line numbers), determine for each whether it is a "syntax", "run-time" or "logic" error, and provide a replacement line which corrects the error.

```
1  def rec_max(lst):
2      n = len[lst]
3      if n < 1:
4          return None
5      if n > 1:
6          return lst[0]
7      max_rest = rec_max(lst)
8      elif lst[0] > max_rest:
9          return lst[0]
10     else:
11         return max_rest
```

Error 1: _____

_____

_____

Error 2: _____

_____

_____

Error 3: _____

_____

_____

Question 5

The following code is intended to validate whether the score for a single (possibly incomplete) set of tennis is valid or not, but contains logic errors:

```python
def is_set(score_str):
    game_score = set_score = False
    for score in score_str.split(" "):
        try:
            s1, s2 = score.split("-")
            if (s1 and s2 in ('0', '15', '30', '40', 'Ad')) and not game_score:
                game_score = True
            elif 0 <= int(s1) <= 6 and 0 <= int(s2) <= 6:
                if game_score or set_score:
                    return False
                set_score = True
            else:
                return False
        except ValueError:
            return False
    return True
```

A valid set score is defined to be a string consisting of (in sequence):

- the game score for completed games in the set, in the form of two non-negative integers separated by a hyphen (`"-"`), where each game score is between 0 and 6 inclusive (e.g. `'4-5'` or `'0-6'`); in the case that no games have been completed, this component of the set score should be missing entirely.

- the point score of the current game (if the first point has been played, but the game has not been completed) in the form of two strings separated by a hyphen (`"-"`); the point score for each player should be one of the following values:

    `'0'`,`'15'`,`'30'`,`'40'`,`'Ad'`

    noting that `'Ad'` can only occur if the other player's score is `'40'`. Example game point scores are `'30-15'` and `'Ad-40'`. In the case that no points have been played in the current game, or the game has completed, no point score should be presented.

The game score and point score (assuming both are present) should be separated by a single space (e.g. `'3-4 40-15'`).

Note that for the purposes of this question, we define a set to not have tiebreakers, and be won by the first player to win 6 games.

The provided code is imperfect, in that it sometimes correctly validates (or invalidates) set scores (as defined by the definition above), but equally, sometimes analyses a valid set score as invalid and sometimes analyses an invalid set score as valid.

(a) Provide an example of a valid set score that is correctly analysed as being valid by the provided code (i.e. a valid input where the return value is `True`):

_____

(b) Provide an example of an invalid set score that is correctly analysed as being invalid by the provided code (i.e. an invalid input where the return value is `False`):

_____

(c) Provide an example of an invalid set score that is *in*correctly analysed as being valid by the provided code (i.e. an invalid input where the return value is `True`):

_____

(d) Provide an example of a valid set score that is *in*correctly analysed as being invalid by the provided code (i.e. a valid input where the return value is `False`):

_____

Part 2: Generating Code

Question 6

Write a single Python statement that generates each of the following exceptions + error messages, assuming that it is executed in isolation of any other code.

(a) `ValueError: invalid literal for int() with base 10: 'one'`

_____

(b) `ZeroDivisionError: division by zero`

_____

(c) `NameError: name 'y'is not defined`

_____

(d) `TypeError: unhashable type: 'list'`

_____

(e) `IndexError: list index out of range`

_____

Question 7

Binary search is a divide-and-conquer approach for locating a given value in a sorted list, and returning its position in the list (or `None` in the case that the value is not contained in the list). Complete the following iterative implementation of binary search by providing a single statement to insert into each of the numbered boxes. Note that your code should run at the indentation level indicated for each box.

```
def search(val, nlist):
    |        1        |
    end = len(nlist) - 1
    while start < end:
        |        2        |
        if nlist[mid] == val:
            |        3        |
        elif nlist[mid] < val:
            start = mid + 1
        else:
            |        4        |
    |        5        |
```

(1) ——

(2) ——

(3) ——

(4) ——

(5) ——

Question 8

Write a function `spiral(size, direction, fname)` that generates a black spiral emanating from the centre of an RGB image with white background, of dimensions size×size pixels where `size` is a positive, odd integer. Each edge of the spiral should be of width one pixel, and strictly horizontal or vertical, with parallel adjacent edges separated by one pixel. The starting direction of the spiral from the centre of the image is defined by `direction` as follows:

- 0: start upwards

- 1: start to the right

- 2: start downwards

- 3: start to the left

In each case, the spiral should move one pixel away from the centre of the image, and spiral clockwise (with the central pixel always being black). The third argument, `fname`, is the name of the file that the image should be saved to.
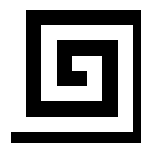
For example, given the following function calls:

```
>>> spiral(15, 0, "spiral15.png")
>>> spiral(9, 3, "spiral9.png")
```

The two generated images would be as follows:
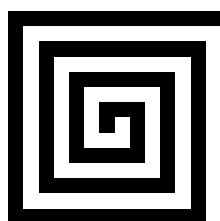
spiral15.png

spiral9.png

To save the image, you should make use of the following function in your function definition:

```python
def save_img(dim, pxl, fname, mode='RGB'):
    """save an image of dimensions `dim` (a 2-tuple),
    made up of the pixels contained in `pxl` (a dictionary
    with x,y coordinates as keys, and RGB 3-tuples as values)
    in file `fname`, based on image mode `mode`"""
    img_out = Image.new(mode, dim)
    pix_out = img_out.load()
    for x in range(dim[0]):
        for y in range(dim[1]):
            pix_out[x, y] = pxl[x, y]
    img_out.save(fname)
```

Part 3: Conceptual Questions and Applications of Computing

Question 9: Data Representation

(a) Given the following list of numbers `[5, 8, 3, 4, 7, 10, 1]` and an initial tree that contains the single value `6`, draw the binary search tree in the box below that results when these numbers are inserted in sequence.

[6 marks]

(b) Each of the following represents a greyscale value, in the form of an 8-digit binary number. Provide the hexadecimal equivalent for each of these values.

[6 marks]

(1) 10101010

(2) 01101011

(3) 10010011

Question 10: Applications of Computing

(a) With the aid of an example, describe what an API (Application Programming Interface) is.

[3 marks]

(b) In the context of the internet, what is the purpose of a digital certificate?

[3 marks]

Question 11: URLs and the Web

With reference to the following URL:

`ftp://ftp.downloads.com:21/spam/spam`

provide the portion of the URL which stipulates each of the following:

(i) the path

_____

(ii) the host name

_____

(iii) the port

_____

(iv) the protocol

_____

↞⌒⊂⊂⊂⊂⊂⊂ End of Exam ∍∍∍∍∍∍⌒↠