THE UNIVERSITY OF
MELBOURNE

Lecture 10 (Week 11): Subject Review and Exam Prep

Dr Simon D'Alfonso

School of Computing and Information Systems
Melbourne School of Engineering

Lecture 9 Challenges

- Write a function file_compare(file1, file2), which receives the path names of two files and checks whether they have identical content. If they do not have identical content, then append the content of file2 to file1.
- Write a *recursive* function sum_nums(n) that accepts an integer *n* and returns the sum of integers from 1 to n.
- Write a *recursive* function reverse(string), that accepts a string input and returns that string in reverse order.

Final 2 Weeks

1. A2 now released
2. Final 2 weeks are revision and exam prep
3. Final tutorial next week will be revision too. Bring any questions or discussion points you might have for tutors

Today

1. Review questions covering topics from earlier weeks:
   1. Variables and expressions
   2. Conditionals
   3. Sequences
   4. Functions
2. Go over some student question requests

Concepts

- Python is:
  - a **high-level** programming language (closer to natural language than machine code)
  - an **interpreted language** as implementations execute instructions without previously **compiling** a program into machine-**language** instructions

Concepts

- Python programs perform calculations using expressions
- Functions are mini-programs, and Python programs are built from many functions
- Errors occur when a Python program isn't written correctly (syntax errors), or when it is written correctly but the instructions can't be completed (runtime errors)
- Even when syntax/runtime error-free, Python programs may not produce the correct output (logic errors); in either case, we can use systematic strategies for finding and removing bugs from our Python programs

Concepts

Variables and data types

int, float, bool, string, list, tuple, set, dictionary

Common issues: mutability, methods and functions, when and how to use

Numerical expressions

Order of operation, math library

Concepts

Conditionals

Common errors: Precedence and program flow

Sequences

Common errors: complex indexing and understanding of methods

Functions

Common errors: arguments, calling, namespace and returning

Iteration

Common errors: nesting and incrementing

Revision Questions 1

Declare and assign the following to a variable named my_var:

1. List of fruits: apple, pear, orange, lemon
2. String to store 54
3. List of tuples: apple, $4.40; pear, $4.20; orange, $3.95; lemon, $3.80

4. When calculating this expression explain the precedence order by evaluating in order each operator one at a time: 6 + 8 * 2 ** 3
5. An expression that returns the string ‘spree’ using only the string s = ‘spam and green eggs’

Revision Questions 1 - Solutions

1. `my_var = ['apple', 'pear', 'orange', 'lemon']`
2. `my_var = '54'`
3. `my_var = [('apple', 4.40), ('pear', 4.20), ('orange', 3.95), ('lemon’, 3.80)]`
4. `6 + 8 * 2 ** 3`
    1. `6 + 8 * 8`
    2. `6 + 64`
    3. `70`
5. `s[0:2] + s[-9:-6]`

Revision Questions 2

Evaluate the following , where a = True and b = False

1. (a and b) or not a

2. ('pi' in 'sky') or ('3.14'.isdigit())

3. Write a function to decide if someone is accepted to a given course based on the following conditions:
    1. ATAR over 85; or
    2. ATAR over 75 plus 2 years or more work experience; or
    3. Over 25 years old and 5 years or more work experience.

Revision Questions 2 - Solutions

```
1.  (a and b) or not a - False
2. ('pi' in 'sky') or ('3.14'.isdigit()) - False
```

```
3.
def accepted(score, work, age):
    if score > 85:
        return True
    elif score > 75 and work >= 2:
        return True
    elif age > 25 and work >= 5:
        return True
    else:
        return False
```

Revision Questions 3

```
#Find the errors in this program:
words = ['rabbit', 'orange', 'cat', 'mat', 'bookkeeper']
lst = []
count = 0
for word in len(words):
    for char in word:
        if char in 'aeiou':
            count += 1

    #if the word has more than 2 vowels, append
    if count > 2
        lst.append(word)

print(lst)
```

Revision Questions 3 - Solutions

Errors:

- Syntax – if count > 2:
- Runtime – for word in words: - len(words) is not an iterable.
- Logic – count is not assigned to zero inside the outer *for* loop, so it will be accumulating without being reset for each word.

Revision Questions 4

Write a function that takes as input a list of integers and prints the next four even numbers for each item in the list.

Revision Questions 4 - Solutions

```
def next_4_even(num_lst):

    for num in num_lst:
        print('The next 4 even numbers of ' + 
str(num) + ' are:')

        if num % 2:
            num += 1
        else:
            num += 2

        for x in range(num, num+8, 2):
            print(x)

        print()
```

Iterators

Definition: an **iterator** is an *object* that keeps track of the *traversal* of a *container*.

object: something you can manipulate
traverse: walk through/across
container: an object representing a collection of other objects (e.g., list, set, etc.)

An **iterable object** will return an iterator object when you pass it to the built-in Python function **iter()**

This happens automatically with *for* loops

Iterator Objects

- Iterators have a **next()** method that will return the next thing in the iteration and update their state/memory of where they are up to.
- Iterators raise a **StopIteration** exception when the container is exhausted and **next()** is called again.
- iterators.py

Iterators

https://groklearning.com/learn/unimelb-comp90059-2022-s1/w12/0/

https://groklearning.com/learn/unimelb-comp90059-2022-s1/w15/18/

Lambda

```
my_tuples = [(6, 'a'), (3, 'b'), (1, 'z'), (5, 'k’)]
#sorts by first element of tuples
print(sorted(my_tuples))

#how to sort by second element of tuples?
print(sorted(my_tuples, key = lambda x: x[1]))
```

Lecture 7 (Week 8)

Copying lists (or other collections)

copying_lists.py

Lecture 8 (Week 9)

- Nested Lists
- Tracking frequencies with using dictionaries

Worksheet 18 – Single Recursive Call

https://groklearning.com/learn/unimelb-comp90059-2022-s1/w18/11/solution/

Lecture Identification and Acknowledgement

Coordinator / Lecturer: Simon D’Alfonso

Semester: Semester 1, 2022

These slides include materials from 2020 - 2021 instances of COMP90059 run by Kylie McColl or Wally Smith