

## CS101 Final Exam (07-05-2015)

> Note that: Exam papers of the set 2 have MCQ as their second problem II.

### Solution and grading scheme for Problem II (set 2)

- There is 15 questions: Q.II.1), Q.II.2) to Q.II.15),
- The order is different between the 2 sets of papers,
- Each question have only one correct answer (given below),
- Two marks for each question, « binary » mode: 2 marks if the answer is the good one, zero mark otherwise.

### List of correct answers (and their justification)

Note: justification were **not asked** during the exam.

1. **a)** (True, False, False, because `d` has a key named 'age', but no key named 'Name' (everything in Python is case sensitive!) neither a key named 'job'),
2. **c)** (Tuples are immutable while values in lists can be changed, the three others choices are wrong),
3. **a)** (`[2015, 'Anil', 2015, 'Anil']` is the only correct answer, indeed, for a list, the right multiplication with an integer is concatenating the list with itself :  $l * k = l + l + \dots + l$  ( $k$  times)),
4. **c)** (user-defined data type is what a class in Python is),
5. **c)** (a new file object `f`, with name 'myfile.txt' is correct, because the request mode 'w' is write mode, so it does not care about the fact that `myfile.txt` does not exist),
6. **b)** (these constants only create confusion is wrong, the three others are correct, as seen in class),
7. **c)** (`raise` is used to force a specific exception, `force` does not exist, `try` is to start a block ended with by the `except` keyword),
8. **a)** (`del d['john']` is the only correct syntax, as seen in class and in lab).
9. **c)** (`TypeError` unsupported operand type(s) for `/`: 'int' and 'str' will be raised, because before actually performing the operation, Python checks if he can do it, and here an integer (5) and a string ('0') are incompatible for the `/` operator),
10. **b)** (Modularity is not a specific characteristic of Object Oriented Programming (OOP), but of modules/-packages programming, while the others are),
11. **b)** (List inside the function: `[1, 2, 3, 4]` and List outside the function: `[1, 2, 3, 4]` is the only correct choice, indeed the function will modify the list (on line 2), so inside the function the list is `[1, 2, 3, 4]` but also afterward),
12. **b)** (from one step to the next, the size of the problem is reduced by half, so there are totally  $\log_2(n)$  such steps is correct, the three others are not relevant),
13. **c)** (`a = 4` and `b = 5` because the while loop will conserve the fact that  $a \leq b$ , and therefore stops as soon as `b = 5`),
14. **b)** (3, because the power operator has a higher priority that the multiplicative operator, so `3*1**3 = 3 * (1**3) = 3 * 1 = 3`),
15. **c)** (the looping variable `i` should not be changed inside the for loop, the three other issues are not happening here),