



Meeting III: Recursion + tuples, lists, dictionaries

15.06.2017

© Andrey Tymofeiuk
(some rights reserved)

Structure



Key points of the Meeting III: English meeting

- Quick recap of Meeting II
- Recursion
- Tuples
- Lists
- Dictionaries
- Meeting V = Midterm!
- Meeting III 10 Words
- Office Hours II – this time collective struggle
- Preparation for Midterm!

Recap of Meeting I & Meeting II



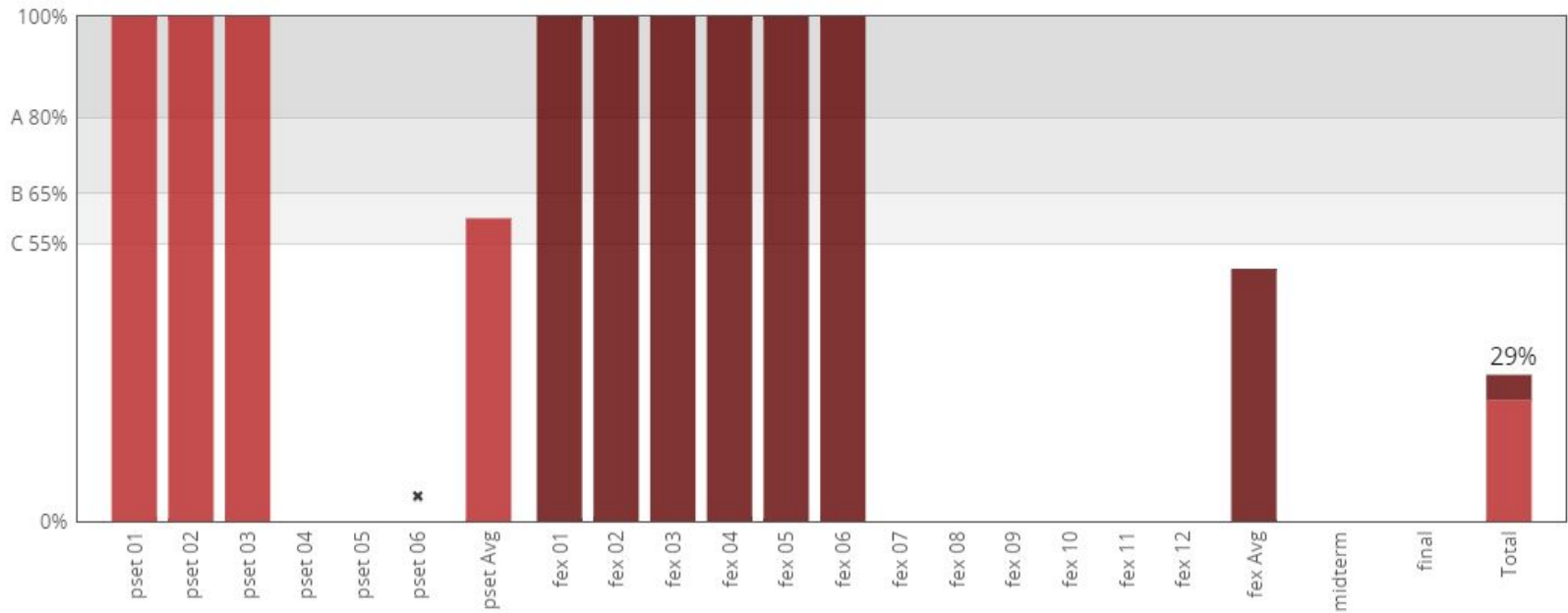
- Variables
- Types of variables
- For loops
- While loops
- Conditional statements
- Functions
- Functions: Definitions
- Methods
- Bisection search

Declarative and Imperative knowledge!

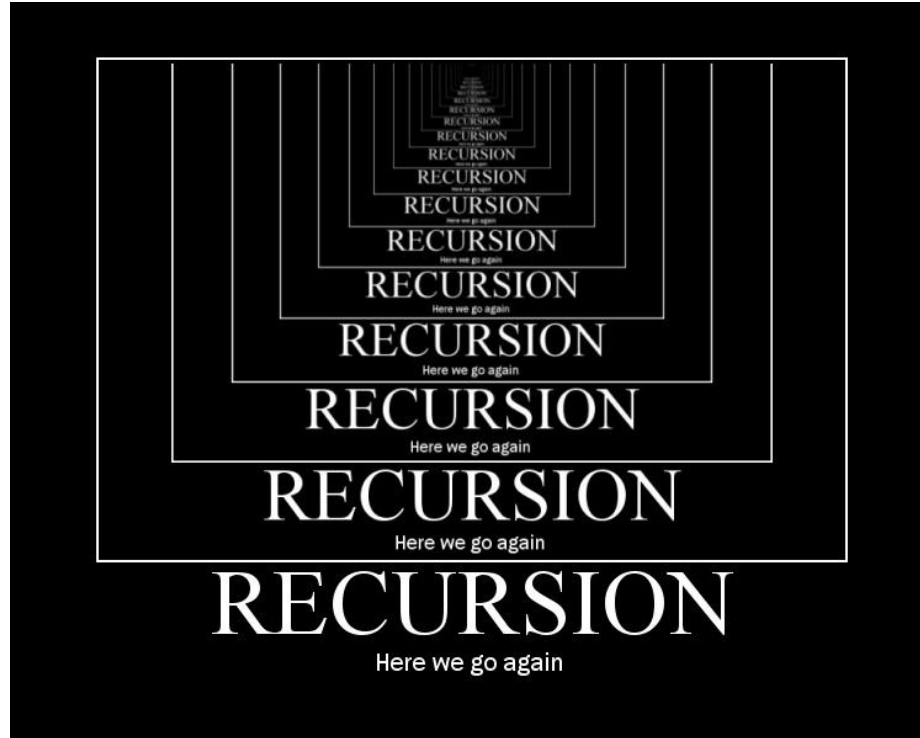
Breakthrough example



Course Progress for Student 'ATymofeiuk' (atymofeiuk@gmail.com)



Recursion



Recursion is a way of programming or coding a problem, in which a function calls itself one or more times in its body.

Recursion: finding factorial



```
def iterative_factorial(n):  
    result = 1  
    for i in range (2, n+1):  
        result *= i  
    return result
```

Iteration

VS

```
def recursive_factorial(n):  
    if n == 1:  
        return 1  
    else:  
        return n*recursive_factorial(n-1)
```

Recursion

Recursion



```
(factorial 6)
(* 6 (factorial 5))
(* 6 (* 5 (factorial 4)))
(* 6 (* 5 (* 4 (factorial 3))))
(* 6 (* 5 (* 4 (* 3 (factorial 2))))))
(* 6 (* 5 (* 4 (* 3 (* 2 (factorial 1)))))))
(* 6 (* 5 (* 4 (* 3 (* 2 1))))))
(* 6 (* 5 (* 4 (* 3 2))))
(* 6 (* 5 (* 4 6)))
(* 6 (* 5 24))
(* 6 120)
720
```

A recursive function has to terminate to be used in a program. A recursive function terminates, if with every recursive call the solution of the problem is downsized and moves towards a base case. A base case is a case, where the problem can be solved without further recursion. A recursion can lead to an infinite loop, if the base case is not met in the calls.

Tuples, lists, dictionaries/dicts



Lists = []

Tuples = ()

Dict = {}



Tuple: ()



```
ninjaTurtlesTuple = ('Michelangelo', 'Leonardo', 'Donatello', 'Rafael')
```

- Empty tuple: ()
- Important property: It is ordered
- Can index: `ninjaTurtlesTuple [1] = 'Leonardo'`
- Immutable (we can't add 'Splinter' and we can't change 'Rafael' to 'Shrek')
- Mix of element types, it could be:

```
ninjaTurtlesTuple = ('Michelangelo', 'Leonardo', 'Donatello', 'Rafael', 3, 4.5)
```

- Funky stuff: `ninjaTurtlesTuple[1:2] = ('Leonardo',)`

List: []



```
ninjaTurtlesList = ['Michelangelo', 'Leonardo', 'Donatello', 'Rafael']
```

- Empty list: []
- Ordered
- Can index: `ninjaTurtlesList[1] = 'Leonardo'`
- Mutable! We CAN add 'Splinter'! We CAN change elements!
- Mix of element types, it could be:

```
ninjaTurtlesList = ['Michelangelo', 'Leonardo', 'Donatello', 'Rafael', 3, 2.66]
```

- No funky stuff: `ninjaTurtlesList[1:2] = ['Leonardo']`



Useful remarks

STRINGS, TUPLES, RANGES, LISTS

- Common operations
 - `seq[i]` → i^{th} element of sequence
 - `len(seq)` → length of sequence
 - `seq1 + seq2` → concatenation of sequences (not range)
 - `n*seq` → sequence that repeats `seq` n times (not range)
 - `seq[start:end]` → slice of sequence
 - `e in seq` → `True` if `e` contained in sequence
 - `e not in seq` → `True` if `e` contained in sequence
 - `for e in seq` → iterates over elements of sequence

Dictionary: {}



```
ninjaTurtlesDict = ['Michelangelo' : 'Leader', 'Leonardo' : 'Sniper', 'Donatello' :  
'Explosives', 'Rafael' : 'Medic']
```

- Empty dictionary: {}
- There is a key
- 'Michelangelo' – key0, 'leader' – val0
- Mutable! We CAN add 'Splinter' : 'Chief'!

A list

0	Elem 1
1	Elem 2
2	Elem 3
3	Elem 4
...	...

A dictionary

Key 1	Val 1
Key 2	Val 2
Key 3	Val 3
Key 4	Val 4
...	...

```
ninjaTurtlesDict['Splinter'] = 'Chief'
```

- *Quickly check:*
In: 'Givi' in uselessSeparatistsDict
Out: False

(!!!!!!!!!!!!HILARIOUS, 26.01.2017!!!!!!!!!!!!)

HILARIOUS!!! In January it was:



...and in 1 week it became! :-D



- *Quickly check:
In: 'Givi' in uselessSeparatistsDict
(!!!!!!!!!!!!!!!!HILARIOUS, 26.01.2017!!!!!!!!!!!!)*
Out: False

Meeting III: 10 words



Key points of the second meeting

- Quotes - лапки
- Curly braces – фігурні скобки
- Square brackets – квадратні скобки
- Alias – «псевдонім» (дзеркало)
- Tuple – кортеж
- To append - приєднати
- To extend – продовжити
- To split – розділити, розколювати
- Pointer – вказівник
- Tutor - репетитор

Homework till Meeting IV (09/02/17)



Homework for the following week

- PSet2 for “less comfortable” + listen Lecture 5 & Lecture 6
- Pset3 for “more comfortable”
- Create GitHub account for “more comfortable”
- START YOUR PREPARATION FOR MIDTERM!!!! 😊

Midterm: 29.06.2017



Key points:

- You can use laptop
- You can google. But everything can be solved just by your memory.
- Don't be afraid – minor mistakes will not count

Office Hours II



America House, Saturday, 17/06/2017, 10:00 – 14:00...but w/o me 😊



Ukraine, Mykoly Pymonenka St, 6, Київ

Don't forget Passport/Driving License!

Andrey: 050 44 33 585

Thank you!

