

# Introduction to Python: Basic Programming

Visualize ComplexCity  
Chair of Information Architecture  
ETH Zürich

February 28, 2014

Please hand in all your Python files in a zip file until the next lecture. You have to solve all but one of the problems of exercise 1 and the full exercise 2 to complete this exercise.

## 1 Loops

In this exercise you have to print the following patterns using *for* and/or *while* loops. You should not use more than one *print* statement per exercise and only print one line at a time. The usage of hard coded lists is not allowed.

- |                             |                                    |                                    |
|-----------------------------|------------------------------------|------------------------------------|
| a) #####<br>#####<br>#####  | c) #<br>##<br>###<br>####<br>##### | e) #0#0#0#<br>0#0#0#0<br>#0#0#0#   |
| b) ###<br>000<br>###<br>000 | d) #0#0#0#<br>#0#0#0#<br>#0#0#0#   | f) #<br>0#<br>0#0<br>#0#0<br>#0#0# |

## 2 Data Structures

We will work with Tuples and Lists in this exercise. You will have to write small pieces of code which mutates one to the other. Please *print* the variable you changed after every exercise.

- a) Tuple to List

```
# Convert the following tuple
foo = (1,2,3,4,5,6,7,8,9)
# to this list
bar = [1,2,3,4,5,6,7,8,9]
```

- b) Append an element to *bar*.

```
# Printing bar should give the following
print(bar)
[1, 2, 3, 4, 5, 6, 7, 8, 9, 0]
```

c) Remove the second element of *bar*.

```
# Printing bar should give the following
print(bar)
[1, 3, 4, 5, 6, 7, 8, 9, 0]
```

d) Reverse the list and store it to *foo*.

```
# They should afterwards look the following
print(bar)
[1, 3, 4, 5, 6, 7, 8, 9, 0]
print(foo)
[0, 9, 8, 7, 6, 5, 4, 3, 1]
```

e) Put the elements of *bar* into a dictionary called *bardict*. The key of the dictionary should be their index in *foo*.

```
# Printing bardict should yield the following
print(bardict)
{0: 0, 1: 9, 2: 8, 3: 7, 4: 6, 5: 5, 6: 4, 7: 3, 8: 1}
```