Programming in Python[1]

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

Academic year 2023/24, I semester

Lecture IV: Repetitions

Repetitions

It is also useful to be able to repeat instructions: it is very convenient, but it also opens a deep Pandora's box...
There are two ways of looping in Python:

Repeat by iterating on the elements of a collection (similar to math notation $\sum_{i \in \{a,b,c\}} f(i)$)

```
for i in range(0, 5):
    # 0 1 2 3 4
    print(i)
```

Repeat while a (variable) condition is true

```
i = 0
while i < 5:
    print(i)
    i = i + 1
```

Euclid's GCD

Two unequal numbers being set out, and the less being continually subtracted in turn from the greater, if the number which is left never measures the one before it until an unit is left, the original numbers will be prime to one another. [...] But, if CD does not measure AB, then, the less of the numbers AB, CD being continually subtracted from the greater, some number will be left which will measure the one before it. [...]

[Euclid *"Elements"*, Book VII, Prop. I, II (c. 300 BC)]

```
a: int = 420
b: int = 240

while a != b:
  if a > b:
    a = a - b
  else:
    b = b - a

print(a)
```

Loops can be difficult to understand

When you have loops, understanding the code can be a difficult task and the only general strategy is to track the execution.

```
# This is known as Collatz's procedure
n = ...
while n > 1:
  if n % 2 == 0:
    # if the remainder of division by 2 is 0, i.e. n
    ↪  is even
    n = n / 2
  else:
    n = 3*n + 1
```

We know (by empirical evidence) that it ends for all $n < 2^{68} \approx 10^{20}$, nobody is able to predict the number of iterations given any `n`.
With loops it is also hard to exploit parallel execution.

Learn to write loops can be hard

When you write a loop, you should have in mind two related goals:

1. the loop must terminate: this is normally easy with `for` loops (when the finite collection ends, the loop ends also), but it can be tricky with `while`s (remember to change something in the condition);
2. the loop repeats something: the programmer should be able to write the "repeating thing" in a way that makes it equal in its form (but probably different in what it does).

The second part (technically known as loop invariant) is the hardest to learn, since it requires experience, creativity, and ingenuity.

Homework

- Create an account on https://github.com/ (if you don't have one) and send me the name.