Introduction to Computer Programming with Python

Why Learn to Program?

Many of the tasks that a researcher performs with his or her computer are repetitive: Collect data from a Web page, convert files from one format to another, execute or interpret 10 or hundreds of BLAST results, first design, look for restriction enzymes, etc.

In many cases it is evident that these are tasks that can be performed with a computer, with less effort on our part and without the possibility of errors caused by tiredness or distractions.

What Is a Program?

A program is a set of ordered instructions designed to command the computer to do something. The word “ordered” is there because is not enough to declare what to do, but the actual order of the directions should also be stated.

Listing 1.1: Protocol for Lambda DNA digestion

```
Restriction Digestion of Lambda DNA

Materials

5.0 mcL    Lambda DNA (0.1 g/L)
2.5 mcL    10x buffer
16.5 mcL   H2O
1.0 mcL    EcoRI

Procedure

Incubate the reactions at 37°C for 1 hr.
Add 2.5 mcL loading dye and incubate for another 15 minutes.
Load 20 mcL of the digestion mixture onto a minigel
```

Computer Programs as Protocols

There are at least two components of a protocol: procedure and materials. A procedure provides specific order like incubate, add, mix, store, load and many others. The same goes for a computer program. The programmer gives specific order to the computer: print, read, write, add, multiply, assign, round, and others.

Data and Instructions

While protocol procedures correlate with program instructions, materials are the data. In protocols, procedures are applied to materials: Mix 2.5 µL of buffer with 5 µL of Lambda DNA and 16.5 µL of H2 0, load 20 µL onto a minigel.

In a program, instructions are applied to data: print the text string “Hello”, add two integer numbers, round a float number.

Simple program example

Listing 1.2: Sample Python Program

```
seq_1 = 'Hello,'
seq_2 = ' you!'
total = seq_1 + seq_2
seq_size = len(total)
print(seq_size)
```

This small program can be read as “Name the string ‘Hello’, as seq_1. Name the string ‘ you!’ as seq_2. Add the strings named seq_1 and seq_2 and call the result as total. Get the length of the string called total and name this value as seq size. Print the value of seq size.” This program prints 11.

Data types

As shown, there are different types of data (often called “data types” or just “types”). Numbers (integers or float), text string, and other data types are covered in Chapter 3. In print(seq size), the instruction is print and seq size is the name of the data. Data is often represented as variables. A variable is a name that stands for a value that may vary during program execution.

Variables

Data is often represented as variables. A variable is a name that stands for a value that may vary during program execution. A value is assigned to a variable by doing:

var = value

Note that this is not an equality as seen in mathematics. In an equality, terms can be interchanged, but in programming, the term of the right (value)
takes the name of the term of the left (var). For example,

seq_1 = ’Hello,’

After this assignment, the variable seq 1 can be used, like, len(seq_1)