Introduction to Bioinformatics

Introduction to Computation

Janyl Jumadinova
January 28, 2016
What is computation?

- A quote from a famous computer scientist: “Computer Science is no more about computers than astronomy is about telescopes” Edsger Dijkstra.
What is computation?

- A quote from a famous computer scientist: “Computer Science is no more about computers than astronomy is about telescopes” Edsger Dijkstra.

Computation is

a sequence of well-defined operations that lead from an initial starting point to a desired final outcome.
What is computation?

▶ A quote from a famous computer scientist: “Computer Science is no more about computers than astronomy is about telescopes” Edsger Dijkstra.

Computation is

a sequence of well-defined operations that lead from an initial starting point to a desired final outcome.

▶ Computer science is the study of computation.
The layers of abstraction in a computation

1. **problem**
2. **solve**
3. **algorithm**
4. **implement**
5. **high-level program**
6. **execute or “run”**
7. **computation**
Computer is a machine that carries out the computation
Hardware/Software

- Software/hardware relationship:
  - Hardware is controlled by software.
  - Software is the collection of instructions that you issue to the computer to perform actions and make decisions.
Simple Structure
Everything is bits

- Binary number system has two digits (0 and 1)
Everything is bits

- Binary number system has two digits (0 and 1)
- Bit - single binary digit
Everything is bits

- Binary number system has two digits (0 and 1)
- **Bit** - single binary digit
- Binary number system is base 2
Everything is bits

- Binary number system has two digits (0 and 1)
- Bit - single binary digit
- Binary number system is base 2

\[ N \text{ bits can represent } 2^N \text{ unique items} \]
What is Computer Programming?

Programming is the act of writing usable and useful software. A program is a set of instructions.
What is Computer Programming?

- Programming is the act of writing usable and useful software.
- A program is a set of instructions.
Algorithms
Any problem can be solved by an algorithm
Algorithms

Any problem can be solved by an algorithm

- **Algorithm** is a procedure for solving a problem in terms of the *actions* to be executed and the *order* in which those actions are to be executed.
Algorithms

Any problem can be solved by an algorithm

- **Algorithm** is a procedure for solving a problem in terms of the *actions* to be executed and the *order* in which those actions are to be executed.
- Both the actions and their order are important.
Algorithms

Any problem can be solved by an algorithm

- **Algorithm** is a procedure for solving a problem in terms of the *actions* to be executed and the *order* in which those actions are to be executed.

- Both the actions and their order are important.

- Computer programs implement algorithms to solve problems.
Programming Language

- We will use Python programming language in this class
- Invented in the Netherlands, early 90s by Guido van Rossum
- Named after Monty Python
- Open source language from the beginning, Python Software Foundation
- Considered a scripting language, but is much more
Why Python?

- Easy to learn, yet powerful
- Emphasizes readability
- Great as both a scripting/glue language and for full-blown application development
- “Scales with the ability of the programmer”
A high-level language

- Built-in high level data structures
- Object oriented
Python

A high-level language
- Built-in high level data structures
- Object oriented

An interpreted language
- You don’t compile your programs
- Automatic memory management
Python

A high-level language
- Built-in high level data structures
- Object oriented

An interpreted language
- You don’t compile your programs
- Automatic memory management

Huge standard library with all sorts of functionality
Python

A high-level language
- Built-in high level data structures
- Object oriented

An interpreted language
- You don’t compile your programs
- Automatic memory management

Huge standard library with all sorts of functionality
Cross platform, and free
Non-interactively

- Write a script (a text file) and run it at the command line with `python MyProgram.py`, maybe adding arguments and other options.
- This is where we’ll eventually end up.
Running Python

Interactively

- Useful for succinct tests, debugging, and for demonstrations.
- Run the python program with no arguments and end up at something like this:

```
jjumadinova@aldenv113:~/repos/2015-2016$ python
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>```
Using an IDE (Interactive Development Environment)

- A hybrid of the above save work in script files, but maintain an interactive session for running them and debugging.
- Python comes with its own IDE named *idle*.
- Others are: PyCharm, SubLime Text, etc.