

The manual to Ducque

Jérôme Rihon^{1,*}

¹KU Leuven, Rega Institute for Medical Research, Medicinal Chemistry, Herestraat 49 - Box 1041, 3000
Leuven, Belgium

**Corresponding author, maintainer*

Contents

1	Introduction	3
1.1	Modules	3
2	Usage	4
2.1	Build	4
2.2	Transmute	5
2.3	Randomise	6
2.4	XYZ to PDB	7
2.5	Graphic User Interface (GUI)	8
	References	9

1 Introduction

There are no mistakes, only happy little accidents.

Bob Ross

A software for the purpose of building native and synthetic nucleic acid duplexes.

Ducque (phonetically IPA : /dʌk/) stands for :

D Acronyms

U Are

C Rather

Q

U

E Tedious

The name was inspired much in the same way that the ORCA Quantum Mechanics package^{1,2} was named. At the time of development, I had different name for Ducque. During one of my evaluation moments, one of the members of my jury pointed out that with that name, I would probably not get enough traction, since Google searches gave millions of results. Given that fact, I was trying to come up with a new name for the model builder.

Weeks later, as I was working, my hat that was resting on my desk in front of me caught my eye. You see, my hat has a tiny duck embroidered on the front. With this idea, I "*researched*" ways to rewrite the word and get as little search engine results as possible, without degenerating the word itself and keeping it phonetically somewhat correct. Thusly, Ducque was born (a second time).

1.1 Modules

Ducque has five modules the user can access.

1. The **build** module, which is the primary module for to use. This will build us the duplex structures we want to use for further
2. The **transmute** will be used, together with an elaborate input, to convert the 'pdb' file to a suitable input for Ducque to use as a building block. The appropriate format has been settled to be the very simple 'json' format.
3. For testing purposes or just to be able to generate randomised sequences, there is a **randomise** module, which generates a randomised duplex structure based on a set of given inputs.
4. To be able to convert 'xyz' formatted molecule structures to a 'pdb' format, I have included a **xyz_pdb** module. Since I am an avid ORCA user and the 'xyz' format is often outputted, I wanted to make it more managable to format a 'pdb' file for myself during development.
5. To have clickable objects for the standard non-terminal user, there is a simple **gui** that can be used to employ the following modules : build, transmute and randomise. The reason that the xyz_pdb is not included is because I did not find a nice way to design a simple gui. Secondly, other QM programs might have different outputs of molecule structures, the 'xyz' format can be rather niche and therefor not worth to effort to design a good gui for. Chimera and PyMol can definitely convert from one format to the other if one cannot work through the CLI.

2 Usage

2.1 Build

Testing a reference.³

2.2 Transmute

2.3 Randomise

2.4 XYZ to PDB

2.5 Graphic User Interface (GUI)

References

- [1] Frank Neese, Frank Wennmohs, Ute Becker, and Christoph Riplinger. The ORCA quantum chemistry program package. *J. Chem. Phys.*, 152(22):224108, jun 2020.
- [2] Frank Neese. Software update: The ORCA program system—version 5.0. *WIREs Computational Molecular Science*, 12(5), mar 2022.
- [3] IUPAC convention of 1983. Symbols for specifying the conformation of polysaccharide chains. recommendations 1981. *Eur J Biochem*, 131(1):5–7, mar 1983.