**Processing of Internal Chemical Structure (ICS) via Normalization**

After the creation of Internal Chemical Structure (**ICS**) by “MakeInpAtomsFromMolFileData” method in source file “mol2atom.c” the next step is now to look into the ICS from the perspective of Normalization.

Normalization is basically the procedure applied to an ICS before generating the InChI to reduce it to its so-called core parent structure, so that we avoid generating different InChI's. This might include reordering formal charges, modifying bond orders, and even adding and eliminating protons.

To start up with Normalized representation of the ICS it is important to get a know-how of the source file and methods involved in Normalization. One of the files that deal majorly with processes involved around normalization is called “**ichi\_bns.c**”. This source file has various methods that gives us different functionalities of Normalization.

The method that deals with the ICS for further normalization based processing is called “**AllocateAndInitBnStruct**”. This method basically allocates and initializes the data structure “**BN\_STRUCT**” which is further used for bond perception and manipulation in chemical structures. This method takes several input parameters like input atoms array, number of input atoms, max. number of added atoms, max. number of additional edges (Bonds), max. number of alternative paths and a pointer to the number of changed bonds.

To check the outputs of this method with an input which in our case is “**Nitro\_Normalize.mol**” file we can use take the help of the “**pBNS**” pointer that points to the data in the BN\_STRUCT, already defined structure holding multiple variables. Figure 1.1 shows the atoms information of ICS and Figure 1.2 shows the bonds information of the ICS for input file.

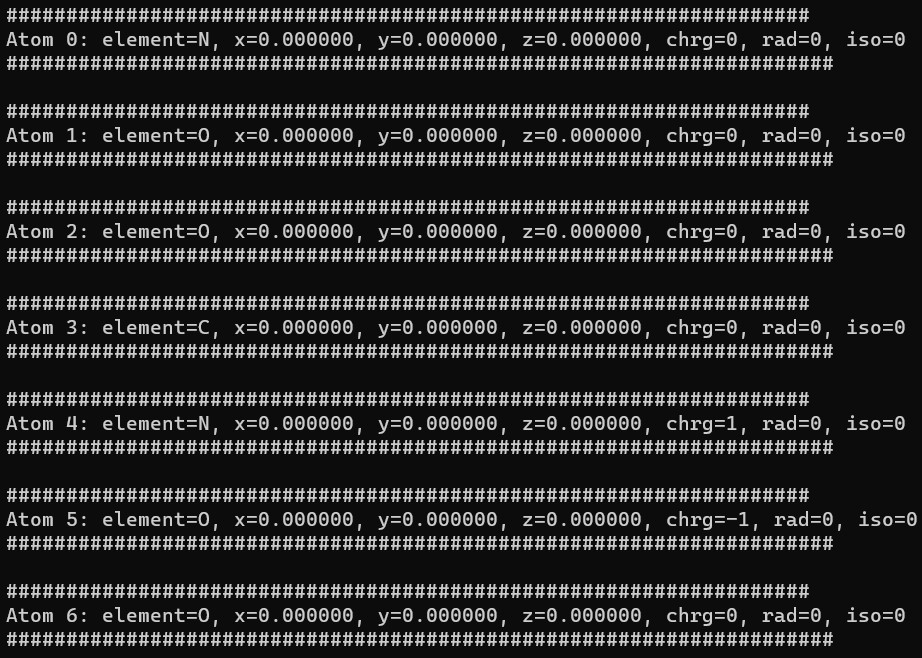


Figure 1.1: Atoms information of the ICS for the input mol file.

A picture containing text, screenshot, font, menu

Description automatically generated

Figure 1.2: Bonds information of the ICS for the input mol file.

Figure 2 shows the output of the normalization method for the data regarding atoms, vertices and bonds.

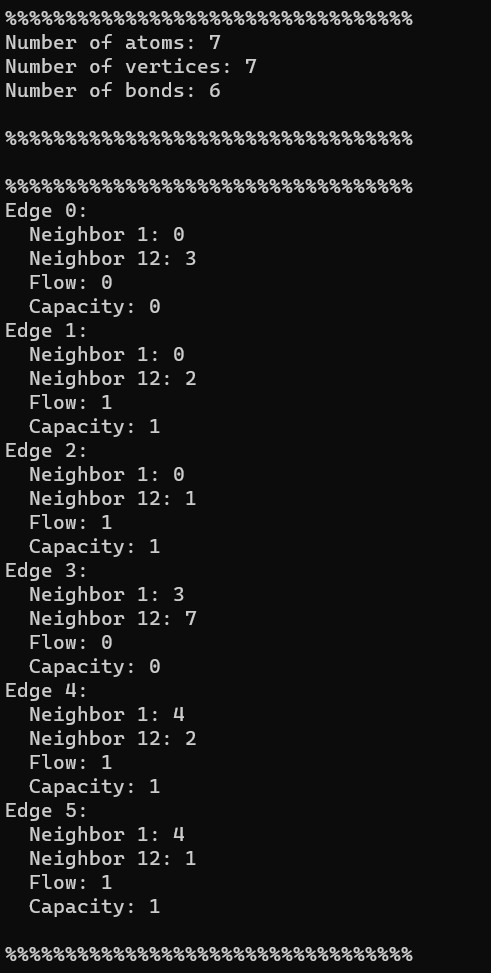


Figure 1.2 : ICS information with Normalization perspective.

Here, it is to note that in figure 1.2 we have “**Edge**”information which in accordance to the source code documentation refers to the bonds. So, as we have 6 bonds (zero indexed) so we have 6 edges. Neighbor 1 in the figure refers to the “**smaller neighbor**”and Neighbor12 refers to the “**neighbor1 ^ neighbor2**”.

So, it means that neighbor1 stores the index of smaller neighbor atom while neighbor12 stores the XOR (bitwise operation) combination of the two neighboring atoms.

Also, just for understanding purposes in the source header file “**ichi\_bns.h**” at line number 183 there is a defined structure named “**BnsEdge**” where the variables like neighbor1, neighbor12, flow and cap are initialized with comments from InChI programmers.