

Foundations 2 Assignment: Part 2

24. Februar 2013

1 Part-2

`part-2.c` is the main file for the second part of the assignment. The *JSON* input file was used with the cJSON (<http://sourceforge.net/projects/cjson/>) C library. *cJSON* was used as it uses a tree structure for the JSON making parsing simpler. The main methods of `part-2.c` are:

- `parse_operator`: This method is the outermost part of the parser. It takes the root node and then loops through the array of variable objects. While doing this, it builds a global *Variable* array.
- `parse_equal_op` The first equal operation is used as allocator. It allocates the values in `t`
- `parse_tuple_op`, `parse_set_op` build the basic structures used, the Set and the Pair from the underlying JSON.
- `parse_equality_op`: Compares two values, sets containing the same elements are equal. Pairs are equal if the elements are equal and in the same order.
- `parse_member_op`: Checks if a value is contained in a set. Returns 1 if the element is contained within the set and 0 if otherwise.
- `parse_base_type` Used to check the types of operators and execute the method related to that type of operation, e.g. building a set for a set operation.

Added a *Variable* type for handling the variables in a reasonable manner. Store the name, e.g. `x1` and the value that the name corresponds to.

2 Changes

There were several changes made to the structures from Part 1. Here is a brief summary:

- Changed print methods so that they can optionally be printed out to a file, rather than always standard out
- Made some new create statements in *Pair* and *Value* to assist in a few pointer issues.
- Fixed the Set Equality method to work with sets not in sequence.