Data Structure Libraries

Data and File Structures Laboratory

http://www.isical.ac.in/~dfslab/2018/index.html

Options I

- Sglib `http://sglib.sourceforge.net/`, `http://freshmeat.sourceforge.net/projects/sglib`
  - doubly linked lists, hashed containers, RB trees
  - *only a single header file*
  - Need to use `-g3` ?
- GDSL `https://github.com/sjchao/GDSL`
  - lists, queues, stacks, hash-tables, BSTs, RB trees, 2D arrays, permutations, heaps, etc.
  - `sudo apt-get install gdsl`
- GNULIB data structures module `https://www.gnu.org/software/gnulib/MODULES.html#ansic_ext_container`
  - AVL trees, RB trees
  - `sudo apt-get install gnulib`

Options II

- GNU libavl `http://adtinfo.org/`
  - BSTs, AVL trees, RB trees
  - example of *literate programming*
  - `sudo apt-get install libavl1 libavl-dev`
- GLib (Gnome) `https://developer.gnome.org/glib/stable/`
  - overkill?
- BSD queues and trees `http://openbsd.su/src/sys/sys/tree.h`
  - RB trees, splay trees

Options III

- CDSA `https://github.com/MichaelJWelsh/cdsa`
- Collections-C `https://github.com/srdja/Collections-C`
- LibDS: A Generic Data Structures Library
  `http://libds.sourceforge.net/`
- Wayne's Little Data Structures and Algorithms Library
  `http://www.cs.toronto.edu/~wayne/libwayne/libwayne.html`

See `https://stackoverflow.com/questions/3039513/type-safe-generic-data-structures-in-plain-old-c` for some opinions / discussions.

Example: RB trees using GDSL

```
gdsl_rbtree_t  gdsl_rbtree_alloc ( const char * NAME,
                                   gdsl_alloc_func_t ALLOC_F,
                                   gdsl_free_func_t FREE_F,
                                   gdsl_compare_func_t COMP_F );

gdsl_element_t gdsl_rbtree_insert ( gdsl_rbtree_t T, void * VALUE,
                                    int * RESULT );

gdsl_element_t gdsl_rbtree_remove ( gdsl_rbtree_t T, void * VALUE );

gdsl_element_t gdsl_rbtree_map_infix ( const gdsl_rbtree_t T,
                                       gdsl_map_func_t MAP F,
                                       void * USER_DATA );
```

Functions you have to provide

```
typedef gdsl_element_t( * gdsl_alloc_func_t )(void * USER_DATA);

typedef void( * gdsl_free_func_t )(gdsl_element_t E);

typedef long int( * gdsl_compare_func_t )(const gdsl_element_t E,
                                          void * VALUE)

typedef int( * gdsl_map_func_t )(const gdsl_element_t E,
                                 gdsl_location_t LOCATION,
                                 void * USER_DATA)
```

How little I know

`http://en.wikipedia.org/wiki/List_of_data_structures`