```
1  /*-----------------------------------------------------------------
2  Copyright (c) 2014 Author: Jagadeesh Vasudevamurthy
3  file: dstack.h
4  -----------------------------------------------------------------*/
5
6  /*-----------------------------------------------------------------
7  This file has dstack class declaration
8  -----------------------------------------------------------------*/
9
10 /*-----------------------------------------------------------------
11 All includes here
12 -----------------------------------------------------------------*/
13 #ifndef dstack_H
14 #define dstack_H
15
16 #include "../util/util.h"
17 #include "../darray/darray.h"
18
19 /*-----------------------------------------------------------
20 static definition - only once at the start
21 Change to false, if you don't need verbose
22 -----------------------------------------------------------*/
23 template <typename T>
24 bool darray<T>::_display = false;
25
26 /*-----------------------------------------------------------------
27 Declaration of dstack class
28 -----------------------------------------------------------------*/
29 template <typename T>
30 class dstack {
31 public:
32   dstack(int size = 50);
33   ~dstack();
34   int num_elements() const;
35   bool isempty() const;
36   bool isfull() const;
37   void push(const T& b); // Stack copies b and holds. Now stack is the owner of b
38   T& top(); // user can get top by alias. He can change its contents also. See explanation in
       implementation
39   void pop(); // Remove top element from the stack. Nothing returned
40
41   void for_each_element_of_stack_from_top_to_bottom(void(*pf) (T& c));
42   bool display()const { return _display; }
43   static void set_display(bool x) {
44     darray<T>::set_display(x);
45     _display = x;
46   }
47   /* no body will copies or equal stack */
48   dstack(const dstack<T>& s) = delete;
49   dstack<T>& operator=(const dstack<T>& rhs) = delete;
50 private:
51   int _sp;
52   darray<T> _stack;
53   static bool _display;
54 };
55
56 #include "dstack.hpp"
57
58 #endif
59 //EOF
60
61
```