

stkdriver.c

```
1 /*****
2 File name:  stkdriver.c
3 Author:     Computer Science, Pacific University
4 Date:      10.11.17
5 Class:     CS300
6 Assignment: Static Generic Stack
7 Purpose:   Test driver for a static stack of generic elements
8 *****/
9
10 #include <stdio.h>
11 #include <stdlib.h>
12
13 #include "../include/stk.h"
14
15 /*****
16 Function:   success
17
18 Description: print a success message
19
20 Parameters: pszStr - the message to print
21
22 Returned:   none
23 *****/
24 static void success (char *pszStr)
25 {
26     printf ("SUCCESS: %s\n", pszStr);
27 }
28
29 /*****
30 Function:   failure
31
32 Description: print a failure message
33
34 Parameters: pszStr - the message to print
35
36 Returned:   none
37 *****/
38 static void failure (char *pszStr)
39 {
40     printf ("FAILURE: %s\n", pszStr);
41 }
42
43 /*****/
```

stkdriver.c

```
44 Function:    assert
45
46 Description: if the expression is true, assert success; otherwise, assert
47 failure
48
49 Parameters:  szStr - the message to print
50
51 Returned:    none
52 *****/
53 static void assert (bool bExpression, char *pTrue, char *pFalse)
54 {
55     if (bExpression)
56     {
57         success (pTrue);
58     }
59     else
60     {
61         failure (pFalse);
62     }
63 }
64
65 *****/
66 Function:    main
67
68 Description: test all the functionality of the stack
69
70 Parameters:  none
71
72 Returned:    Exit Status
73 *****/
74
75 int main ()
76 {
77     Stack sTheStack;
78
79     puts ("Program Start\n");
80
81     puts ("SUCCESS TESTS:");
82
83     stkLoadErrorMessage ();
84     success ("Loaded Error Messages");
85
86     stkCreate (&sTheStack);
```

stkdriver.c

```
87  success ("Stack Created");
88
89  assert (stkSize (&sTheStack) == 0, "Stack size is 0",
90      "Stack size is NOT 0");
91
92  stkTerminate (&sTheStack);
93  success ("Stack Terminated");
94
95  puts ("\nProgram End");
96
97  return EXIT_SUCCESS;
98 }
99
```