COMP500 / ENSE501: Week 11 – Exercise:

EXERCISE NAME: *Function Pointers*

The following program source code is incomplete:

```
1   #include <stdio.h>
2
3   // TODO: 1) Add the typedef here:
4
5   // Prototypes:
6   void repeat(int times);
7   void test(void);
8
9   // The main function definition:
10  int main(void)
11  {
12      // TODO: 4) Modify this call to pass in test:
13      repeat(10);
14
15      // TODO: 6) Add another call to your void function:
16
17      return 0;
18  }
19
20  // TODO: 2) Modify the parameter list of the repeat function to add
21  //          in the function pointer parameter for the function to be
22  //          called repeatedly. Also, do not forget to update the
23  //          function declaration prototype on line 6 to match...
24  void repeat(int times)
25  {
26      for (int k = 0; k < times; ++k)
27      {
28          // TODO: 3) Add the call to the function pointer here:
29      }
30  }
31
32  void test(void)
33  {
34      printf("Test!\n");
35  }
36
37  // TODO: 5) Add your own void function, and prototype:
```

To complete the program, add a **`typedef`** for a function pointer where the function returns nothing, and takes in no parameters (**`//TODO: 1`**).

Modify the **`repeat`** function to add a second parameter of the function pointer **`typedef`**, which is the address of the function to be repeatedly called **`times`** number of times upon calling **`repeat`** (**`//TODO: 2`** and **`//TODO: 3`**).

Next, modify the call to **`repeat`** in the **`main`** function to pass in the address of the **`test`** function (**`//TODO: 4`**).

Finally, add your own **`void`** function that can be used with the **`repeat`** function (**`//TODO: 5`**). In the **`main`** function, add a call to **`repeat`** using your new function as an argument (**`//TODO: 6`**). Ensure the whitespace of your source code is well formatted.