CECS 326 Assignment #1 - Threads

Due Date: 22 SEP 2016

20 points

Read the documentation for `pthread_cancel` by typing `man pthread_cancel` on the Linux command line. Using this information and the model provided below, write a program where the initial (main) thread creates a second thread. The main thread should read input from the keyboard, waiting until the user presses the Enter key. At that point, it should kill off the second thread and print out a message reporting that it has done so. Meanwhile, the second thread should be in an infinite loop, each time around sleeping five seconds and then printing out a message.

This program should be coded in C, using the gcc C compiler on POSIX (i.e. Linux or Mac OS) systems. Note: you may have to compile my sample code using the gcc flag `-lpthread` in order for it to work on some systems.

Answer the following questions in a text file and submit along with your code:

1. Can the sleeping thread print its periodic messages while the main thread is waiting for keyboard input? Explain your answer.

2. Can the main thread read input, kill the sleeping thread, and print a message while the sleeping thread is in the early part of one of its five-second sleeps? Explain your answer.

```c
1  #include <pthread.h>
2  #include <unistd.h>
3  #include <stdio.h>
4
5  static void *child(void *ignored){
6     sleep(3);
7     printf("Child is done sleeping 3 seconds.\n");
8     return NULL;
9  }
10
11 int main(int argc, char *argv[]){
12    pthread_t child_thread;
13    int code;
14
15    code = pthread_create(&child_thread, NULL, child, NULL);
16    if(code){
17       fprintf(stderr, "pthread_create failed with code %d\n", code);
18    }
19    sleep(5);
20    printf("Parent is done sleeping 5 seconds.\n");
```

```
21    return 0;
22 }
```

Deliverables

Demonstrate your code for the instructor. Submit a copy of your code, the answers to the two questions above, and a screenshot of your executed code to **Beachboard Dropbox**.