**Lab 1 Threads**

**Exercise 1:**

You are required to design and develop a multi-threaded Java program containing 2 classes:

(a) Busy class: Objects of this class should be able to be run as separate threads. When run they should display a message to indicate they are busy (e.g. "Busy...") 10 times before terminating.

(b) Test class: should create and start a new thread which runs a Busy object and terminates.

Both threads should indicate when they have started and finished.

**Exercise 2:**

Modify the solution to exercise 1 to do the following:

(a) the busy thread should sleep for 10 milliseconds after displaying each busy message, if interrupted while asleep the thread should terminate.

(b) after the main thread starts the busy thread it should sleep for 70 milliseconds before interrupting the busy thread.

**Exercise 3:**

Modify the solution to exercise 1 to do the following:

The main thread should not terminate until the busy thread has finished. To do so it needs to "join" the busy thread. Have a look at the join method in the Java API. The busy thread should run with maximum priority.

**Exercise 4:**

Modify the solution to exercise 1 to do the following:

(a) The Busy class should store the name of the person who is busy. Code an appropriate constructor for the class. The thread should display a message indicating the person is busy (e.g. "Harry is busy..."), sleep for 10 milliseconds, and repeat the sequence 10 times before terminating.

(b) The main thread should start 2 busy threads for different Busy objects (e.g. Harry and Sally) and wait for both to terminate before terminating itself.