# Assignment 1
## System Setup Instructions

**CSE 522- Real Time Embedded Systems**
**Spring 2014**

## A. How to setup the system to do the assignment :

Following system setup options are available for doing the first assignment.

# define Preferred_Operating_System  Ubuntu 12.04 LTS

If (Ubuntu 12.04 LTS already present on Personal Computer -PC)
{

    1. Do_system_setup_for_KernelTracing();

}
Else if (Plan on using PC && want to install Ubuntu on VM-Player)
{

    1. Download VMPlayer from below link and install :
        https://my.vmware.com/web/vmware/free#desktop_end_user_computing/vmware_player/6_0

    2. Download Ubuntu 12.04 (.iso) LTS from :
        http://www.ubuntu.com/download/desktop

    3. Open VMPlayer and create a new virtual machine and browse the .iso file from step2.
    It will detect OS and version automatically.

    4. Create user login and provide other details and let Ubuntu boot on VMplayer.

    5. Open terminal and do the installation of updates:
        # sudo  apt-get update

    6. Do_system_setup_for_KernelTracing();

}
Else if (Plan on using BYENG-217/215 Lab Machines)
{

    While (Every time you login to Lab Machines)
    {
      1.  Run "VMware->VMware Player" as Administrator on Windows

      2.  Click "Open a Virtual Machine" and select C:\Dr_Lee_217\VM2\Light_Ubuntu_Full_1\
         Light_Ubuntu_Full_1"

      3.  Click "Play virtual machine"

      4.  Enter the following log in details
         ID: esp
         Password: asu1234

      5.  If you wish to collect kernel trace data then do Do_system_setup_for_KernelTracing();
    }

}

void <mark>Do_system_setup_for_KernelTracing()</mark>
{

        1. Open the terminal in Ubuntu 12.04 LTS and do the following pre-requisite installation:
                # sudo apt-get install asciidoc
                (Let it complete installation)

                # sudo apt-get install swig
                (Let it complete installation)

        2. Install the GUI tool KERNELSHARK for viewing the kernel trace.
                # sudo apt-get install kernelshark
                (Let it complete installation)

}

**<mark>NOTE:</mark> Since every time you log off from the lab machines all your saved data is lost. The data/programs as well installations done during each log in would also be lost after you log out.**

**/***
**We are also looking into providing access to <mark>online students (primarily)</mark> with a server hosted Ubuntu Image. It is however up to the online students if they want to use this facility or wish to work on their own personal machines.**

**The instructions for access and log in would be provided to you in a couple of days. Details of this would be posted on Blackboard and sign in details would also be provided.**
**\*/**


**B. Useful API's for doing the assignment:**

Following are <mark>**some**</mark> **of the API's** that would be used for doing the assignment. It is recommended that you refer to the Linux manual pages for each of these API's to better understand their usgae.

pthread_setschedparam(pthread_self(), policy, &param);
pthread_getschedparam(pthread_self(), &policy, &param);


pthread_attr_setinheritsched(&attr, inheritsched);
pthread_attr_getinheritsched(&attr, &inheritsched);

pthread_attr_setschedpolicy(&attr, policy);
pthread_attr_getschedpolicy(&attr, &policy);

pthread_attr_getschedparam(&attr, &param);
pthread_attr_setschedparam(&attr, &param);

pthread_attr_init

pthread_attr_destroy(&attr);

pthread_mutex_init(&mutex, &my_mutex_attr);

clock_gettime();
clock_nanosleep();


## C. Compilation of the program (tasks)

# gcc your_program.c -o app1 **-lpthread** **-lrt** **-D_GNU_SOURCE**

## D. Running the program and gathering trace data+analysis

After the installations have been completed the trace tool can be used with the following command:

# **trace-cmd record -e sched_switch** ./app1

For more details on using the trace-cmd tool (Section: Tracing with Trace-cmd)
http://omappedia.org/wiki/Installing_and_Using_Ftrace#Tracing_with_Trace-cmd