Using Killdevil and Kure - Basic Lab Exercises

These are a series of very simple lab questions to exercise basic functionality on Killdevil and Kure. These exercises mostly apply to either machine so feel free to try these out on the machine you intend to use or try them on both!

1. Start the X windows server, X-win32.
2. Login to Killdevil or Kure using a ssh client.
3. Use the `pwd` command to print your working directory. This is your home directory. Change directories to the scratch space (see notes) and list all the files there (use `ls` to list files). Is there a directory there with (the name of) your onyen? If not, create one using `mkdir`.
4. What queues are available on the cluster? How many job slots per user are there on the week queue? What about the int (kure) or the day (killdevil) queue?
5. What is the longest a job can run in the debug queue? How about on the week queue?
6. The `parallelpi_kure` and `parallelpi_killdevil` executables are in the `/netscr/markreed/courses` directory. Copy them to your scratch directory. Check that your environment has the `mvapich/intel` module loaded and switch to it if you don't. Try submitting the appropriate parallel MPI job, `parallelpi`, to run on three processors, for example

   ```
   bsub -n 3 mpirun parallelpi_killdevil
   ```

   Which queue is it running in? Use `bjobs` to view the job. Use `bpeek` to view standard output (stdout) and and standard error (stderr).
7. Now submit the `parallelpi` job to run in the hour (killdevil) or now (kure) queue. Check to see that it worked.
8. Now try submitting the job again. Ooops, I didn't mean to submit that job! Kill the job. Did it work?
9. Run the `bhosts` command and examine the output. Do the same with the `lshosts` command. Hint, to look at a page at a time, pipe this into `more`, e.g.

   ```
   lshosts | more
   ```
10. What modules do you already have in your environment? Add the `grace` package. How did your environment change? Grace is a 2D plotting package that is freely available. Bring up the Grace GUI by running `xmgrace`. Feel free to explore.
11. Use the `matlab` command to bring up the Matlab GUI interactively on a compute node.
12. If you are a SAS/Stata/Matlab user, you should try out the specialty scripts.
13. If you compile code then try picking your favorite C compiler and compiling the `parallelpi` code. The source is in `/netscr/markreed/parallelpi.c`. After compiling the job, run the job in batch.