2DV605 Parallel Programming (H17)

**Project Description**

**Aim**: *Practical experience with parallel programming*

**Begin Date**: *October 2, 2017*

**Deadline**: *October 16, 2017 (final possibility for presentation: October 23, 2017)*

**Submission**: *an archive file (zip or tar) that includes documentation (PDF or DOCX format) together with all files relevant to execute and test your program should be sent before the deadline via e-mail to* [*sabri.pllana@lnu.se*](mailto:sabri.pllana@lnu.se)*; use this word document to document your solution; if the submitted program does not run (execute) using your instructions provided in the documentation, your solution is not accepted as successful.*

**Instructions**: *you are allowed and encouraged to use the literature recommended in the course; use of other literature is also recommended; assignment must be completed* ***independently*** *and without the help of other colleagues or the teacher.*

**Evaluation**: Presentation of project results (10 – 15 minutes, 7 – 10 slides),

First possibility: October 16, 2017, room B3033V, 10:15 – 12:00h

Final possibility: October 23, 2017, room B3033V, 10:15 – 12:00h

To pass the course students must complete Task 1.

For a better grade, one needs to complete also the Task 2 and/or Task 3.

**Additional Information**

* UNIX Tutorial for Beginners

<http://www.ee.surrey.ac.uk/Teaching/Unix/>

Example: use *passwd* to reset your default login password

* OpenSSH

http://www.openssh.com/

* Contact FTK IT in case of problems with VPN

Email: [ftk.it@lnu.se](mailto:ftk.it@lnu.se)

* Emil (Xeon Phi)

To enable the use the ICC compiler, execute the following command on Emil,

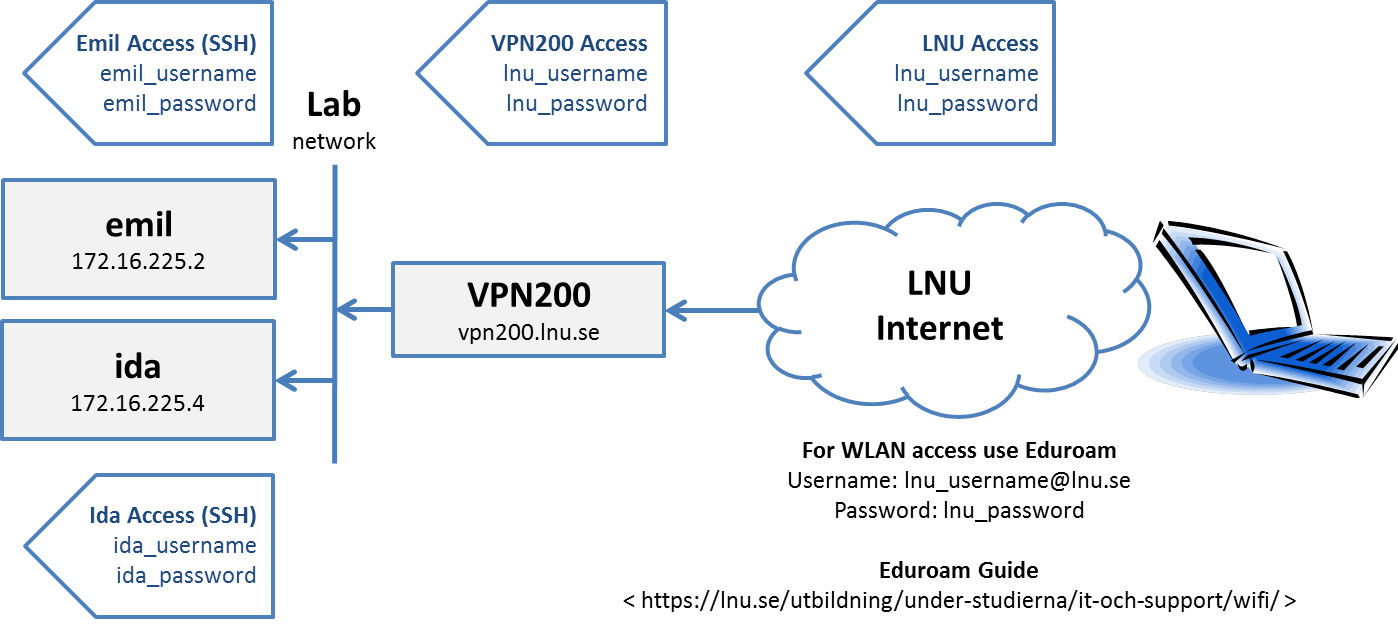
source /opt/intel/bin/compilervars.sh intel64

* Ida (GPU)

NVCC compiler version 8.0

GCC compiler version 5.4.0

For experimentation, you will use our parallel computers Emil & Ida. Your account will be removed after the end of this learning period (October 31.)

****

**Problem Definition**

PI may be calculated using the following sequential C-code,

#include <stdio.h>

#include <stdlib.h>

#define PI 3.14159265358979323846264

int main (int argc, char \*argv[]){

double m, ni, mypi = 0.0;

int i, iterations;

iterations = 24000000;

m = 1.0 / (double)iterations;

**for (i = 0; i < iterations; i++){**

**ni = ((double)i + 0.5) \* m;**

**mypi += 4.0 / (1.0 + ni \* ni);**

**}**

mypi \*= m;

printf(" MyPI = %.20lf\n", mypi);

printf("MyPI - PI = %.20lf\n", (mypi - PI));

}

**Task 1: Calculate PI in parallel using two 12 –core CPUs on Emil**

* Parallelize the code using OpenMP
* Measure and visualize (for instance, using an Excel chart) execution time for 1, 6, 12, 24, and 48 threads, for each of the following numbers of iterations: 24000000, 48000000, 96000000

**Task 2: Calculate PI in parallel using the Intel Xeon Phi co-processor on Emil**

* Parallelize the code using OpenMP for Intel Xeon Phi
* Measure and visualize (for instance, using an Excel chart) execution time for 1, 4, 8, 16, 30, 60, 120, 180, and 240 threads, for each of the following numbers of iterations: 24000000, 48000000, 96000000

**Task 3 (optional task): Calculate PI in parallel using the GPU on Ida**

* Parallelize the code using CUDA
* Measure and visualize (for instance, using an Excel chart) execution time for various number of threads and number of iterations

**Solution**

*Use up to four pages to describe your solution. Code excerpts may be used for illustration.*

…

**Task 1: Two 12 –core CPUs (Emil)**

Execution times for various problem sizes and threads

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 6 | 12 | 24 | 48 |
| 24000000 |  |  |  |  |  |
| 48000000 |  |  |  |  |  |
| 96000000 |  |  |  |  |  |

…

**Task 2: Intel Xeon Phi (Emil)**

Execution times for various problem sizes and processes

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 4 | 8 | 16 | 30 | 60 | 120 | 180 | 240 |
| 24000000 |  |  |  |  |  |  |  |  |  |
| 48000000 |  |  |  |  |  |  |  |  |  |
| 96000000 |  |  |  |  |  |  |  |  |  |

…

**Task 3: GPU (Ida)**

Execution times for various problem sizes and processes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Instructions for running and testing the program**

*Provide here any information that is relevant for running and testing your program.*