Name & <email>:

2DV605 Parallel Programming (H19)

Project Description

Aim: *Practical experience with parallel programming*

Begin Date: *Sep 30*

Deadline: *Oct 14*

Submission: *send your presentation slides and your report (PDF) to sabri.pllana@lnu.se*

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: Oct 14, room D1167V, 10:15 – 12:00h

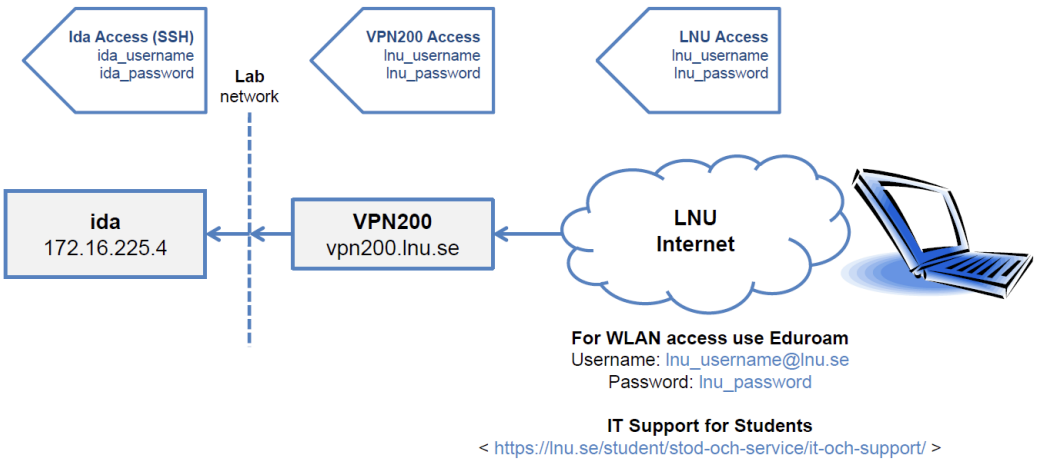
Final possibility: Oct 21, room D1167V, 10:15 – 12:00h

To pass the course students must complete Task 1. For a better grade, one needs to complete also the Task 2.

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 you need support for VPN access
  Email: stefan.petersson@lnu.se
- Ida (GPU)
  NVCC compiler version 8.0
  GCC compiler version 5.4.0

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



Name & <email>:

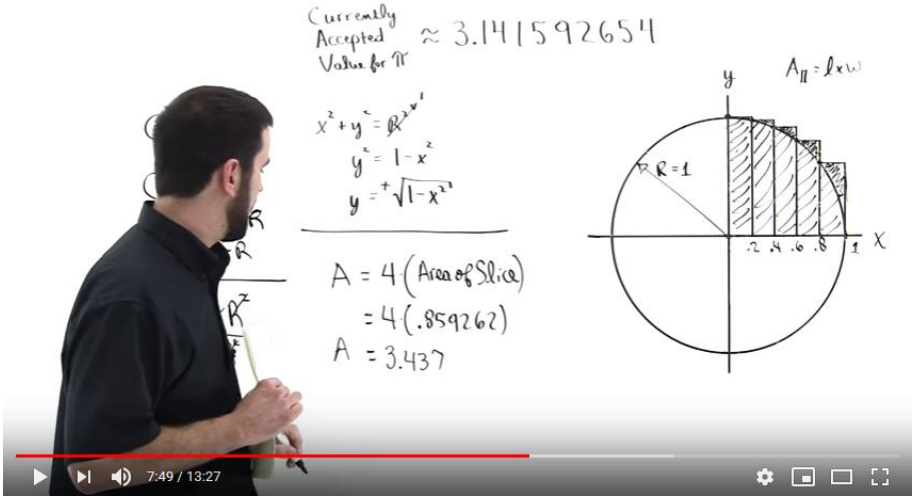
Problem Definition

PI may be approximately 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));
}
```

$$\int_0^1 \frac{4}{(1+n*n)} dn = \pi$$

$$\sum_{i=0}^{iterations} \frac{4}{(1+n_i*n_i)} \Delta n \approx \pi$$



An explanation of PI calculation is provided by Saul Rémi Hernandez at,
https://www.youtube.com/watch?v=uK2OQMUAUDQ

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

- Parallelize the code using OpenMP
- Measure and visualize (for instance, using Excel charts) execution time and speedup 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 GPU on Ida

- Parallelize the code using CUDA
- Measure and visualize (for instance, using Excel charts) execution time and speedup.

Name & <email>:

Solution

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

…

Task 1: Two 12 – core CPUs (Ida)

Execution times for various problem sizes and threads

	1	6	12	24	48
24000000					
48000000					
96000000					

…

Task 2: GPU (Ida)

Execution times for various problem sizes

iterations	**Execution time**
24000000	
48000000	
96000000	

Instructions for running and testing the program

Provide here any information that is relevant for running and testing your program, including the path (/home/username/foldername/filename) to the code that is ready for assessment.