Assignment: basic programming
Numerical integration

The purpose of this assignment is for you to

- get started back in writing C or C++ code
- get familiar with running code on mamba
- write a simple program that will be reused in future assignments

As usual all time measurements are to be performed on the cluster.

1 Preliminary: running anything on mamba

Question: Write a code that simply prints the name of the current machine. You can obtain that name using function `gethostname` (check man page for details). Write that code in file `prelim.cpp` or `prelim.c`. You can test this works by compiling it with `make prelim`.

Question: Run that code on a mamba compute node using `./queue_prelim.sh`. This will start a PBS job. Once this job has completed, a file `preliminary_answer` will be created containing the name of the compute node the command has run on.

2 Numerical Integration

Numerical integration is often used when one wants to compute $\int_a^b f(x)dx$ but one does not know how to find a primitive of f. You can use the definition of integration to obtain a simple approximation by computing $\frac{b-a}{n}\sum_{i=0}^{n-1} f\left(a+(i+.5)*\frac{b-a}{n}\right)$. n is often called the number of point in the approximation. (This is the numerical integration using the rectangle rule. You can learn more at `https://en.wikipedia.org/wiki/Numerical_integration`.)

Note that you do not need to understand numerical integration. The problem is just to evaluate $\frac{b-a}{n}\sum_{i=0}^{n-1} f\left(a+(i+.5)*\frac{b-a}{n}\right)$ for a particular combination of a, b, n, and f.

The provided package contains multiple functions to integrate in `libfunctions.a`. The functions are named `f1`, `f2`, `f3`, `f4`, and take two parameters: the first one is a floating point number x where the function is computed, and the second one is *intensity* an operation intensity. The second parameter is used to make the function take more time.

The code you should write should take 5 command line parameters:

- `functionid`, an integer to know which function to integrate. If `functionid` is 1, integrate `f1`
- `a`, the lower bound of the integral
- `b`, the upper bound of the integral
- `n`, an integer which is the number of points to compute the approximation of the integral
- `intensity`, an integer which is the second parameter to give the the function to integrate

The code should compute the integral and output the value of the integral on `stdout` (and nothing else). The code should also measure the time it took to compute the integral and write that time (in seconds) to `stderr`.

Question: Write the described code. You can use the provided archive as a template. It contain a template code and makefile to help you write the code. You should only need to complete `main.cpp`. You should be able to test if your code is correct using `make test`.

Question: Report the time it takes on the cluster to integrate `f1` using different number of points (from 10^1 to 10^8) and with different operation intensity (from 1 to 10^4). To help you in that task, you should be able to run `make bench` which should run the benchmark in a PBS job. Once that job is completed, you can draw charts using `make plot` which reports time in a pdf file `plot/time_plots.pdf`.

Make sure you keep this code around as it is your base for comparisons in future assignments. Also note that a run with 10^8 points and an operation intensity of $1,000$ could take an hour to run.