HW 1

CPSC424

Jake Brawer

February 1, 2017

1 Building and Running the Code

1.1 Software and Dev. Environments

All the programming for this assignment was done in vim. This document, including the figures, were made using emacs (and gnuplot). The only module loaded used in this assignment is Langs/Intel/15.

1.2 Env Output

MKLROOT=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/mkl
MANPATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/man/en$_{\text{US}}$:/home/apps/fas
GDB$_{\text{HOST}}$=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/debugger/gdb/intel64$_{\text{mic}}$/bi
ia-mic HOSTNAME=login-0-0.local IPPROOT=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2}}$
INTEL$_{\text{LICENSEFILE}}$=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/licenses:/opt/intel/
TERM=xterm SHELL=/bin/bash HISTSIZE=1000 GDBSERVER$_{\text{MIC}}$=/home/apps/fas/Langs/Intel/2
SSH$_{\text{CLIENT}}$=172.27.41.66 41162 22 LIBRARY$_{\text{PATH}}$=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/compose
PERL5LIB=/opt/moab/lib/perl5 FPATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.}}$
QTDIR=/usr/lib64/qt-3.3 QTINC=/usr/lib64/qt-3.3/include MIC$_{\text{LDLIBRARYPATH}}$=/home/apps/fas/I
linux-release/lib:/opt/intel/mic/myo/lib:/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$
SSH$_{\text{TTY}}$=/dev/pts/63 ANT$_{\text{HOME}}$=/opt/rocks USER=jnb37 LD$_{\text{LIBRARYPATH}}$=/home/apps/fas/Langs/
linux-release/lib:/opt/intel/mic/myo/lib:/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$
MIC$_{\text{LIBRARYPATH}}$=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/compiler/lib/mic:/h
ROCKS$_{\text{ROOT}}$=/opt/rocks CPATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/ipp/
YHPC$_{\text{COMPILER}}$=Intel NLSPATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/com
MAIL=/var/spool/mail/jnb37 PATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/b
3.3/bin:/opt/moab/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/sbin:/usr/java/latest/b
YHPC$_{\text{COMPILERMINOR}}$=164 mposer$_{\text{xe2015.2.164}}$/debugger/gdb/intel64$_{\text{mic}}$/share/locale/%l_%t/%N:/hor

MAIL=/var/spool/mail/jnb37 PATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/b
3.3/bin:/opt/moab/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/sbin:/usr/java/latest/b
YHPC$_{\text{COMPILERMINOR}}$=164 TBBROOT=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.16}}$
F90=ifort PWD=/home/fas/cpsc424/jnb37/scratch/HW1/Pr1 _LMFILES_=/home/apps/fas/Modul
YHPC$_{\text{COMPILERMAJOR}}$=2 JAVA$_{\text{HOME}}$=/usr/java/latest GDB$_{\text{CROSS}}$=/home/apps/fas/Langs/Intel/201
mic DOMAIN=omega LANG=en$_{\text{US.iso885915}}$ MODULEPATH=/home/apps/fas/Modules
MOABHOMEDIR=/opt/moab YHPC$_{\text{COMPILERRELEASE}}$=2015 LOADEDMODULES=Base/yale$_{\text{hpc}}$:La
F77=ifort MPM$_{\text{LAUNCHER}}$=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/debugger/m
CXX=icpc SSH$_{\text{ASKPASS}}$=/usr/libexec/openssh/gnome-ssh-askpass HISTCONTROL=ignoredups
INTEL$_{\text{PYTHONHOME}}$=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.16}}$
SHLVL=1 HOME=/home/fas/cpsc424/jnb37 FC=ifort LOGNAME=jnb37
QTLIB=/usr/lib64/qt-3.3/lib CVS$_{\text{RSH}}$=ssh SSH$_{\text{CONNECTION}}$=172.27.41.66
41162 172.18.89.8 22 MODULESHOME=/usr/share/Modules LESSOPEN=||/usr/bin/lesspipe.sh
%s arch=intel64 INFOPATH=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/debugger/
CC=icc INCLUDE=/home/apps/fas/Langs/Intel/2015$_{\text{update2}}$/composer$_{\text{xe2015.2.164}}$/mkl/include
G$_{\text{BROKENFILENAMES}}$=1 BASH$_{\text{FUNCmodule}}$()=() { eval ‘/usr/bin/modulecmd
bash \$*‘ } _=/bin/env OLDPWD=/home/fas/cpsc424/jnb37/scratch/HW1k

1.3 Running Code

The code for this assignment is separated amongst two directories Pr1 and Pr2. To compile all the code at once, simply run setup.sh located in the toplevel directory. To run the code on Omega for problem 1, navigate into Pr1 and run pr1.sh. Running this file (`qsub pr1.sh`) will output the text file out.txt. This file contains measurements regarding the timing of the integration function, estimates of divide operation, and the estimated value of pi.

Similarly, running pr2.sh located in Pr2 outputs a file output.txt. This file contains data relating the size of N to MFLOPS.

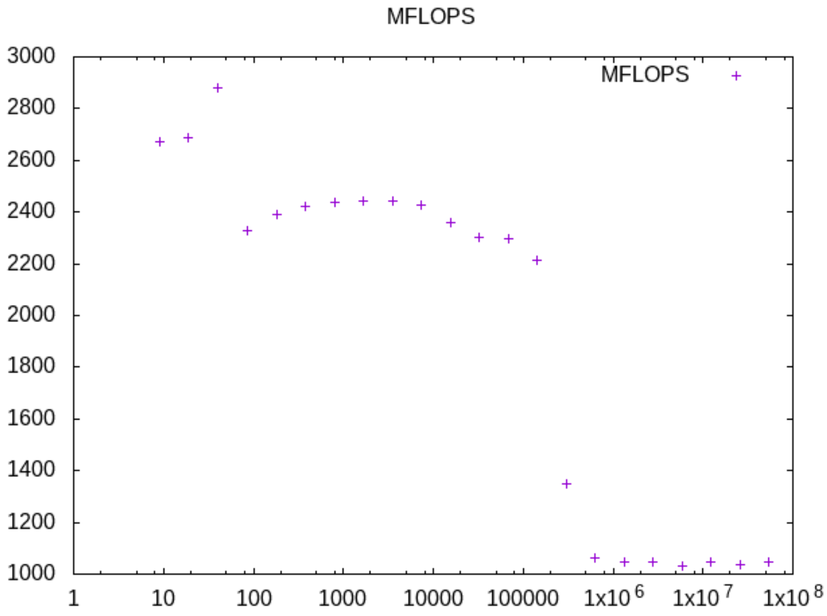
2 Pr 1

Interestingly there was almost no variation in speed for all 4 compiler flag options. For each of the flags the code ran for approximately 7.18 seconds. This likely means that the compiler was unable to optimize my code in a meaningful way. As such MFLOPS stayed constant across compiler options (~834 MFLOPS).

Although it's not completely true that no optimizations were taking place. In order to calculate the latency for divides, I created a timed two toy functions. In theory these functions only differed by the presence/absence of a single floating point divide. Taking the difference between the two functions should have in theory been useful for calculating the latency. However this difference differed wildly between compiler flag options. The difference was very large for for the first combination of flags, but very small for the rest. These later combinations employed techniques like loop unrolling and parallelization, which could explain the speed up.

The value of π calculate was correct to the 7th decimal place. This allowed my to estimate cos(pi) correctly to 1o decimal places (-1) and sin(pi) to 8 (0).

3 Pr 2



Above is a graph comparing array length to MFLOPS. Clearly there is a precipitous drop off in performance as N gets large. This is likely due to the increased number of cache misses