Compilers & Optimized Librairies

- Modules
- Environment
  - .bashrc
  - env
  - $PATH
  - ...
- Compilers : GNU, Intel, Portland
- Memory considerations : size, top, ulimit
- Hello world ! exercice

modules

```
[bvr@hmem00 ~]$ module avail

----------- /usr/share/Modules/modulefiles ------
Armadillo/2.4.4-goolf-1.4.10-Python-2.7.3
ATLAS/3.8.4-gompi-1.1.0-no-OFED-LAPACK-3.4.0
Autoconf/2.69-GCC-4.8.2
Autoconf/2.69-goolf-1.4.10
Automake/1.13.4-goolf-1.4.10
Automake/1.14-GCC-4.8.2
Bert/0.9.0-82
Bert/0.9.1
BLACS/1.1-gompi-1.1.0-no-OFED
blas/gcc/3.2.1
[...]
```

modules

```
[bvr@hmem00 ~]$ module avail GCC

----------- /usr/share/Modules/modulefiles ------
GCC/4.6.3 GCC/4.7.2 GCC/4.8.1 GCC/4.8.2

[bvr@hmem00 ~]$ module avail intel

----------- /usr/share/Modules/modulefiles ------
intel/clusterstudio/2013.0.028
intel/clusterstudio/advisor/2013.1
intel/clusterstudio/amplifier/2013.5
[...]
[bvr@hmem00 ~]$ module avail pgi

----------- /usr/share/Modules/modulefiles ------
pgi/11.2-1
```

modules

```
[bvr@hmem00 ~]$ module --help

 Modules Release 3.2.10 2012-12-21 (Copyright GNU GPL v2 1991):

   Usage: module [ switches ] [ subcommand ] [subcommand-args ]

Switches:
[...]       add|load
            rm|unload
            list
            purge
            whatis
            ...
```

environment

```
[bvr@hmem00 ~]$ more .bash_profile
 # .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin

export PATH
```

environment

```
[bvr@hmem00 ~]$ more .bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific aliases and functions

module load intel/compilerpro

[bvr@hmem00 ~]$ alias
```

environment

```
[bvr@hmem00 ~]$ env
MKLROOT=/usr/local/intel/ics_2013.0.028/composer_xe_2013
.1.117/mkl
MANPATH=/usr/local/intel/ics_2013.0.028/composer_xe_2013
.1.117/man/en_US::/usr/share/man
GLOBALSCRATCH=/globalfs/bvr
HOSTNAME=hmem00.cism.ucl.ac.be
IPPROOT=/usr/local/intel/ics_2013.0.028/composer_xe_2013
.1.117/ipp
INTEL_LICENSE_FILE=/opt/flexlm
SHELL=/bin/bash
TERM=xterm-256color
HISTSIZE=1000
TMPDIR=/scratch
[...]
```

environment

```
[bvr@hmem00 ~]$ echo $PATH
                which ifort
                module purge
                which ifort
[bvr@hmem00 ~]$ echo $MANPATH
                man -k gcc
```


```
[bvr@hmem00 ~]$ icc -help

                Intel(R) C++ Compiler Help
                 ==========================

 Intel(R) Compiler includes compiler options that optimize for instruction
sets that are available in both Intel(R) and non-Intel microprocessors, but
may perform additional optimizations for Intel microprocessors than for
non-Intel microprocessors.  In addition, certain compiler options for
Intel(R) Compiler are reserved for Intel microprocessors.  For a detailed
description of these compiler options, including the instructions they
implicate, please refer to "Intel(R) Compiler User and Reference Guides >
Compiler Options."

usage: icc [options] file1 [file2 ...]
       icpc [options] file1 [file2 ...]
```

icc

```
[bvr@hmem00 ~]$ icc -help

 Optimization
 ------------
  -O0
  -O1
  -O2
  -O3
  -Ofast enable -xHOST -O3 -ipo -no-prec-div -static

 Code Generation
 ---------------
 -x<code> SSE4.1 AVX

[bvr@hmem00 ~]$ icc -xAVX flops.c
[bvr@hmem00 ~]$ ./a.out

Fatal Error: This program was not built to run in your system.
Please verify that both the operating system and the processor support
Intel(R) AVX.
```

icc

```
[bvr@hmem00 ~]$ icc -help

 Linking/Linker
 ------------
  -L<dir>
  -shared
  -static

   [bvr@hmem00 ~]$ icc -fast flops.c -o flops.fast
[bvr@hmem00 ~]$ icc -O0 flops.c -o flops.O0
[bvr@hmem00 ~]$ ls -l flops.fast flops.O0
-rwxrwxr-x 1 bvr bvr 638944  flops.fast
-rwxrwxr-x 1 bvr bvr  12725  flops.O0

   ldd  LD_LIBRARY_PATH
   file
```

```
[bvr@hmem00 ~]$ file flops.fast
flops.fast: ELF 64-bit LSB executable, x86-64, version 1 (SYSV),
statically linked, for GNU/Linux 2.6.18, not stripped
```

ex. flops.c

Practice yourself :

```
cp ~bvr/flops.c ~
```

Compile with gcc then icc then pgcc

Memory considerations

```
cp ~bvr/par512mb.f ~

ifort -O0 par512mb.f

size a.out
  text    data     bss     dec     hex filename
569154   20304 536978152  537567610  200aa17a   a.out
bss = block started by symbol (uninitialized global data)
  Top

 PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
5011 bvr       20   0  525m 512m  516 R 100.0  0.8   0:20.46 a.out
```

Memory considerations

```
[bvr@hmem00 ~]$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) 4647764
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 515034
max locked memory       (kbytes, -l) unlimited
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 10240
cpu time               (seconds, -t) unlimited
max user processes              (-u) 1024
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Memory considerations

```
[bvr@hmem00 ~]$ ulimit -v 600000
[bvr@hmem00 ~]$ time ./a.out

real  0m5.751s
user  0m4.014s
sys   0m1.652s
    [bvr@hmem00 ~]$ ulimit -v 500000
    [bvr@hmem00 ~]$ time ./a.out
Killed

real  0m0.001s
```

Ex2. Hello world !

```
[bvr@hmem00 ~]$ wget
ftp://ftp.belnet.be/mirror/ftp.gnu.org/gnu/hello/hello-2.8.tar.gz
```

Untar it.
See INSTALL : ./configure; make; make install

With pgcc and icc
Use --prefix

Change the default message into the src
-> advantage of the Makefile

Librairies

```
cp -rp ~bvr/Matmul ~
```

ACML (AMD Opteron) versus
MKL (Intel Math Kernel Lib)

BLAS - Basic Linear Algebra Subprograms
LAPACK - A package of higher level linear algebra
routines;
FFT - a set of Fast Fourier Transform routines
RNG - a set of random number generators and statistical
distribution functions.

Librairies

BLAS

http://www.netlib.org/blas/index.html

```
L1 scalar, vector and vector-vector operations
L2 matrix-vector operations
L3 matrix-matrix operations
```

Librairies

LAPACK

http://www.netlib.org/lapack/

provides routines for solving systems of simultaneous **linear equations**,
least-squares solutions of linear systems of equations, **eigenvalue** problems, and singular value problems.
The associated matrix factorizations (LU, Cholesky, QR, SVD, Schur, generalized Schur) are also provided, as are related computations such as reordering of the Schur factorizations and estimating condition numbers.
Dense and banded matrices are handled, but not general sparse matrices.

for real and complex matrices, in both single and double precision.

Librairies

exercice

See matmul.F :

```
#if defined MATMUL
          c = matmul(a, b)
#elif defined BLAS
          call dgemm('no transpose','no transpose',m,n,p,
     &                1.0d0, a, m, b, n, 1.0d0, c, m)
```

With pgf90 & acml (-L/opt/acml/4.4.0/pgi64/lib/ -lacml)

With ifort & mkl (-lmkl)